Time–Frequency Synthesis of Noisy Sounds With Narrow Spectral Components

Damián Marelli, *Member, IEEE*, Mitsuko Aramaki, *Member, IEEE*, Richard Kronland-Martinet, *Senior Member, IEEE*, and Charles Verron, *Member, IEEE*

Abstract—The inverse fast Fourier transform (FFT) method was proposed to alleviate the computational 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 tradeoff between time and frequency resolution. In this paper, we propose an alternative to the inverse FFT method for synthesizing colored noise. The proposed approach uses subband signal processing to generate time–frequency noise with an autocorrelation function such that the noise obtained after converting it to time domain has the desired power spectral density. We show that the inverse FFT method can be interpreted as a particular case of the proposed method, and therefore, the latter offers some extra design flexibility. Exploiting this property, we present experimental results showing that the proposed method can offer a better tradeoff between time and frequency resolution, at the expense of some extra computations.

Index Terms—Additive synthesis, audio systems, colored noise synthesis, frequency-domain synthesis, time–frequency analysis.

I. INTRODUCTION

SOUND synthesis techniques can be broadly classified into two main approaches. The first consists of techniques based on physical models, which aim at simulating the physics of sound sources. The second consists of those based on signal models, which aim at reproducing perceptual effects independently of the physics of the source [1].

 Physical models have been successfully applied for synthesizing environmental sounds, e.g., wind and fire noises [2], [3]

or rolling sounds [4]. However, synthesis based purely on physical modeling is computationally expensive in general, which makes its implementation unrealistic for real-time applications. To go around this issue, a combined approach taking into account signal properties as well as physical considerations, was proposed in [5] for modeling percussion sounds. This hybrid approach was successfully applied for synthesizing sounds of solid contacts [6], liquids [7], and environmental sounds like wind and fire [8].

Within the category of signal models, linear ones present the great advantage that the synthesis parameters can be determined from the analysis of natural sounds [9]. A linear signal model called "sinusoids plus noise" represents a sound as a combination of a deterministic contribution (i.e., a sum of time-varying sinusoids, also called partials) and a stochastic contribution (i.e., a "time-varying filter" with a white noise entrance) [10]. This model has been successfully applied for analysis, transformation and synthesis of a wide class of musical and environmental sounds [10], [11]. Extensions of this approach for including the model of transients are also available [12], [13]. An alternative linear model called "colored noise by sum of sinusoids" (CNSS) was proposed in [14], and permits representing noisy sounds by summing short-time sinusoids with random frequencies and phases. Another approach called "bandwidth-enhanced sinusoidal model" or "spectral line broadening" permits synthesizing narrowband noise by modulating the phase of a sinusoidal component [15], [16].

Unfortunately, linear models, unlike nonlinear ones such as frequency modulation [17] or waveshaping [18], often suffer from a prohibitive computational cost, making them inappropriate for real time applications. However, the high potential of applications provided by linear synthesis models made the reduction of the computational complexity a major challenge in the early 1990s. One of the most efficient approaches was proposed by Rodet and Depalle [19] who described the so-called "inverse fast Fourier transform" (IFFT) algorithm. This technique, initially designed for additive synthesis of slowly time-varying sinusoids, consists in approximating the time domain signal in the time–frequency domain. A study carried out in [20] concluded that IFFT is the most efficient additive synthesis method available. By providing a clever choice of the analysis window, this method permits a computational cost gain of approximately ten times compared to the time domain oscillator bank method [21]. A different approach for alleviating the computations in additive synthesis is proposed in [22], by approximating sine functions using polynomials. The complexity of this

method is proportional to the sum of the frequency of the sinusoid components rather than the number of components.

A "sinusoids plus noise" synthesizer can be designed by including the generation of colored noise within the IFFT procedure [23], [21]. This is done by generating time–frequency noise using the envelope of the power spectral density (PSD) of the target signal. In [10] and [24], a line-segment spectral envelope is proposed, providing a flexible representation of the stochastic component, which accurate enough for sound synthesis applications. Additionally, the spectral segments can be evenly spaced on the equivalent rectangular bandwidth (ERB) scale to take into account some perceptual properties of human hearing [25]. In practice, synthesis of environmental sounds requires the generation of deterministic and stochastic signals with a wide range of temporal and spectral behaviors. For instance, sounds such as impacts [6], drops of water [7], or fire cracklings [8] have very sharp transients. On the other hand, noisy sounds such as wind whistling or fire hissing have very narrowband components which cannot be synthesized using sinusoids [8]. The IFFT method requires a synthesis window whose length matches that of the autocorrelation function. Thus, the window needs to be long to reproduce narrowband noisy components, and consequently, cannot be used to generate short transient signals.

An approach for overcoming the limitation of the IFFT method consists in using different window lengths according to the type of sound [26]. However, as we show in Section VI-B, this approach does not permit the synthesis of short transients in narrow noisy components. Another approach consists in using the spectral line broadening idea described above when generating a sinusoid [27], [28]. However, as we show in Section V-C, the IFFT method permits changing the frequency of a sinusoid only once per frame. Therefore, this approach can only be used to generate very narrow noise components.

In this paper, we propose a more flexible technique for designing a time–frequency noise synthesizer, which overcomes the inherent tradeoff of the IFFT method. The proposed technique offers accurate resolutions both in time and frequency, using a single synthesis window, at the expense of some extra computations in comparison with the IFFT method. This technique can be used to generate colored noise with any arbitrary spectral shape, and is particularly suitable for generating narrowband noisy signals having sharp transients. It can be considered as a generalization of the IFFT algorithm, and uses the so-called subband technique [29]–[31] to generate time–frequency noise with an autocorrelation function such that, after the synthesis operation, results in noise with the desired PSD.

The paper is organized as follows. In Section II, we give an overview of the IFFT algorithm and in Section III we introduce the subband technique. In Section IV, we show how to use this technique for generating stationary stochastic sounds. In Section V, we compare the subband method with other existing approaches, and in particular we show how it can be considered as a generalization of the IFFT method. In Section VI, we explain how to use the subband technique for synthesizing nonsta-
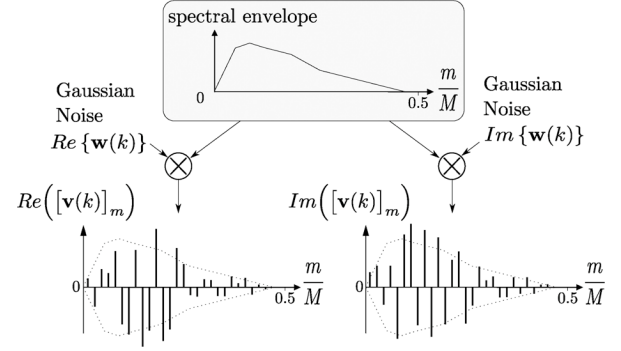

Fig. 1. Building the real and imaginary part of the stochastic STS $\mathbf{v}(k)$ at block k by multiplying a given spectral envelope by two white noise sequences.

tionary stochastic sounds, and we test the proposed scheme synthesizing realistic environmental sounds. Finally, we give concluding remarks in Section VII.

Notation: Throughout the paper, we will use the following notational convention: scalars are denoted using normal (i.e., non-bold) lowercase letters (e.g., x); and vector and matrix using lowercase bold letters (e.g., $\mathbf{x}$) and uppercase bold letters (e.g., $\mathbf{X}$), respectively. The ith entry of a vector $\mathbf{x}$ is denoted by $[\mathbf{x}]_i$ and the i,jth entry of a matrix $\mathbf{X}$ is denoted by $[\mathbf{X}]_{i,j}$. Finally, $\bar{\mathbf{X}}$ denotes the complex conjugate of the matrix $\mathbf{X}$ and $\mathbf{X}^*$ denotes its transpose conjugate, i.e., $\mathbf{X} = \overline{\mathbf{X}^T}$.

II. TIME–FREQUENCY SYNTHESIS OF COLORED NOISE USING THE INVERSE FFT METHOD

Using the IFFT method, a stochastic signal $y(t)$ is generated by the following overlap-add procedure:

$$y(t) = \sum_{k=-\infty}^{\infty} \sigma(t - kD) v_k(t - kD) \qquad (1)$$

where $\sigma(t), t \in \mathbb{Z}$ denotes the *synthesis window* which has tap size M (i.e., $\sigma(t) = 0$ if $t < 0$ or $t \geq M$), D denotes the *synthesis hop size* satisfying $D \leq M$, and the kth block of samples $v_k(t)$, $t = 0, \ldots, M - 1$ is obtained doing the inverse fast Fourier transform of the M-dimensional random vector $\mathbf{v}(k)$, i.e.,

$$v_k(t) = \sum_{m=0}^{M-1} [\mathbf{v}(k)]_m e^{j2\pi \frac{m}{M} t}, \ 0 \leq t < M. \qquad (2)$$

The short-time spectrum (STS) $\mathbf{v}(k)$ at block k is in turn generated using the technique depicted in Fig. 1. More precisely, its mth component $[\mathbf{v}(k)]_m$ is given by

$$[\mathbf{v}(k)]_m = \left| g\left(e^{j \frac{2\pi m}{M}} \right) \right| [\mathbf{w}(k)]_m \qquad (3)$$

where $|g(e^{j(2\pi m)/(M)})|$, $m = 0, \ldots, M - 1$ denotes the spectral envelope and $\mathbf{w}(k)$ is a white complex vector random process (i.e., a sequence of uncorrelated complex random vectors with uncorrelated entries) with Gaussian distribution.

A desired property for stochastic synthesis is that white noise is generated when $|g(e^{j(2\pi m)/(M)})| = 1$ for all $m = 1, \ldots, M$. This property is guaranteed if the z-transform $\sigma(z)$ of the synthesis window $\sigma(t)$ satisfies

$$\sum_{m=0}^{M-1} \left| \sigma \left(e^{j(\omega + \frac{2\pi m}{M})} \right) \right|^2 = 1 \text{ for all } \omega \in [-\pi, \pi]. \quad (4)$$

Remark 1: Since the real and imaginary components of $[\mathbf{w}(k)]_m$ have Gaussian distributions, its magnitude and phase have Rayleigh and uniform distributions, respectively [32], [33]. Consequently, in [10], [25], [24], [27], [28], [23] the generation of the STS is simplified by letting $[\mathbf{v}(k)]_m$ be a complex random variable with amplitude $|g(e^{j(2\pi m)/(M)})|$ and phase given by a random variable uniformly distributed in the interval $[-\pi, \pi]$.

A central issue in the synthesis of colored noise using the IFFT method is its inherent tradeoff between time and frequency resolutions. On the one hand, synthesizing signals whose temporal envelope varies rapidly (transient signals) requires a short synthesis window (i.e., $M < 128$ with a sampling frequency of 44.1 kHz). On the other hand, synthesizing signals with a high frequency resolution (e.g., narrowband signals) requires a long synthesis window (i.e., $M > 1024$ at 44.1 kHz). As mentioned previously, the subband technique can be used to overcome this tradeoff at the expense of some extra computations. We give an overview of this technique next.

III. SUBBAND METHOD

The subband method was first proposed to alleviate the computational complexity of the time domain method in adaptive filtering applications [34], [35], and it found later applications in channel equalization [36], wideband noise control [37], beamforming [38], system identification [39], [31], etc. A recently proposed application is the efficient implementation of linear systems [40]–[42]. The idea is depicted in Fig. 2. The linear system $g(z)$ is approximated by splitting the input signal $x(z)$ into M subbands using the array of filters $\mathbf{h}(z) = [h_1(z), \ldots, h_M(z)]^T$, followed by a downsampling operation of factor D (i.e., one out of D samples is kept). In this way, the subband signal $\boldsymbol{\xi}(z) = [\xi_1(z), \ldots, \xi_M(z)]^T$ is generated, which is called the subband representation of the signal $x(z)$. The subband model is an $M \times M$ transfer matrix $\mathbf{S}(z)$ whose output is denoted by $\boldsymbol{\psi}(z) = [\psi_1(z), \ldots, \psi_M(z)]^T$. The output signal $\hat{y}(z)$ is generated by upsampling $\boldsymbol{\psi}(z)$ by a factor of D (i.e., $D-1$ zero valued samples are added between every two samples), then filtering each component using the array of filters $\mathbf{f}^*(z) = \overline{[f_1(z), \ldots, f_M(z)]}$, and finally adding together all the resulting signals.

We will assume that the filters in the arrays $\mathbf{h}(z)$ and $\mathbf{f}(z)$ are finite-impulse response (FIR), having tap sizes l_h and l_f, respectively. Also, to simplify the notation, and without lost of generality, we will assume that they are causal. The entries of the subband model $\mathbf{S}(z)$ are FIR filters whose supports are defined by two matrices $\mathbf{P}, \mathbf{Q} \in \mathbb{Z}^{M \times M}$ as follows: for each $i, j = 1, \ldots, M$, $[\mathbf{S}(k)]_{i,j} = 0$ for all $k \notin \{[\mathbf{P}]_{i,j}, \ldots, [\mathbf{Q}]_{i,j}\}$). The total number of parameters of the subband model $\mathbf{S}(z)$ is denoted by $l_S = \sum_{i,j=1}^{M} [\mathbf{Q}]_{i,j} - [\mathbf{P}]_{i,j} + 1$. We assume
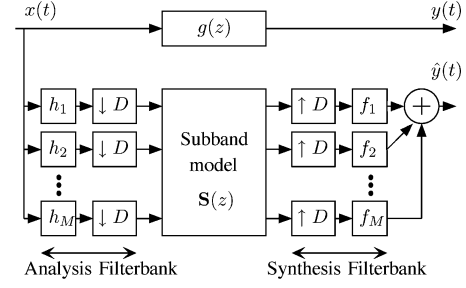

Fig. 2. Subband technique aiming at approximating a linear system $g(z)$ by the subband model $\mathbf{S}(z)$.

that the filterbanks are of Gabor type (i.e., there exists a prototype filter $h_1(z)$ such that, $h_m(z) = h(e^{j2\pi(m-1)/(M)}z)$ for all $m = 2, \ldots, M$ and all $z \in \mathbb{Z}$, and a similar condition holds for $f_m(z)$, $m = 1, \ldots, M$) for which a numerically efficient algorithm exists [43].

The subband technique presents the advantage of significantly reducing the computational cost of a linear system implementation. In particular, using the algorithm in [43], and assuming that M is a power of two, so that an M-point FFT can be implemented with $2M \log_2 M$ (real) multiplications using the Radix-2 algorithm [44, Ch. 6.1], the implementation of either the analysis or the synthesis filterbank requires $(l_x + 2M \log_2 M)/D$ real multiplications per time domain sample, where l_x stands for either l_h or l_f. Also, the computation of $\mathbf{S}(z)$ requires $4l_S/D$ (real) multiplications per sample.

IV. TIME–FREQUENCY SYNTHESIS OF STATIONARY COLORED NOISE USING THE SUBBAND METHOD

In this section, we explain how the subband method can be used for synthesizing stationary colored noise. The technique described here will be used in Section VI to propose a technique for synthesizing nonstationary stochastic sounds.

Suppose we want to generate colored noise $y(t)$ with a prescribed spectral envelope $|g(e^{j\omega})|$. We can do so using the subband method described above by letting $x(t)$ be white noise, $g(z)$ be a spectral factor of $|g(e^{j\omega})|^2$ and designing the subband configuration to minimize $\mathcal{E}\{|y(t) - \hat{y}(t)|^2\}$ ($\mathcal{E}\{\cdot\}$ denotes expected value). To this end, we need to use the so-called polyphase representation [45].

A. Polyphase Representation

The polyphase representation of a scalar signal $x(t)$ is the D-dimensional vector signal $\mathbf{x}(k)$ defined by

$$[\mathbf{x}(k)]_d = x(kD + 1 - d), \text{ for all } d = 1, \ldots, D$$

Also, the polyphase representation of a linear system with impulse response $g(t)$ is the $D \times D$ impulse response matrix $\mathbf{G}(k)$ defined by

$$[\mathbf{G}(k)]_{d,e} = g(kD + e - d), \text{ for all } d, e = 1, \ldots, D \quad (5)$$

The polyphase representation of an analysis filterbank with filters $h_m(z)$, $m = 1, \ldots, M$ and downsampling factor D, is the $M \times D$ transfer matrix $\mathbf{H}(z)$ defined by

$$[\mathbf{H}(k)]_{m,d} = h_m(kD + d - 1) \quad (6)$$

for all $m = 1, \ldots, M$ and $d = 1, \ldots, D$. Also, the polyphase representation of a synthesis filterbank with filters $f_m(z)$, $m = 1, \ldots, M$ and upsampling factor D is given by $\mathbf{F}^*(z)$, where $\mathbf{F}(z)$ is defined in a way similar to (6).

If $\mathbf{h}(z)$ is of Gabor type, and the prototype filter $h(z)$ is causal with tap size l_h, then its polyphase representation is given by [43]

$$\mathbf{H}(z) = \mathcal{W}_M \mathbf{A}_{M,l_h} \mathbf{\Lambda}_h \mathbf{B}_{l_h, D}(z) \qquad (7)$$

where $\mathcal{W}_M \in \mathbb{C}^{M \times M}$ is the discrete Fourier transform (DFT) matrix, i.e., $[\mathcal{W}_M]_{k,l} = M^{-1/2} e^{-j(2\pi)/(M)kl}$ and

$$\mathbf{A}_{M,l_h} = \left[\underbrace{\mathbf{I}_M, \mathbf{I}_M \ldots, \mathbf{I}_M}_{q \text{ times}} \right]_{:,1:l_h} \qquad (8)$$

$$\mathbf{B}_{l_h, D}^T(z) = [\mathbf{I}_D, \ldots, z^{-p+1} \mathbf{I}_D]_{:,1:l_h} \qquad (9)$$

$$\mathbf{\Lambda}_h = \text{diag}\{h(0), \ldots, h(l_h - 1)\} \qquad (10)$$

with $p = \lceil (l_h)/(D) \rceil$, $q = \lceil (l_h)/(M) \rceil$ ($\lceil x \rceil$ denotes the nearest integers greater than or equal to x), $\mathbf{I}_M$ being the $M \times M$ identity matrix, $[\mathbf{X}]_{:,1:l_h}$ denoting the matrix formed with the first l_h columns of $\mathbf{X}$, and $\text{diag}\{x_1, \ldots, x_l\}$ denoting the diagonal matrix with elements $\{x_1, \ldots, x_l\}$ in its main diagonal.

B. Subband Model Design

Using the definitions above, the scheme in Fig. 2 can be expressed in polyphase form as follows:

$$\mathbf{y}(z) = \mathbf{G}(z)\mathbf{x}(z) \qquad (11)$$

$$\hat{\mathbf{y}}(z) = \mathbf{F}^*(z)\mathbf{S}(z)\mathbf{H}(z)\mathbf{x}(z) \qquad (12)$$

If $\mathbf{H}(z)$ and $\mathbf{F}(z)$ have left inverses (i.e., $\mathbf{H}^\dagger(z)\mathbf{H}(z) = \mathbf{I}$ and $\mathbf{F}^\dagger(z)\mathbf{F}(z) = \mathbf{I}$, where the superscript $\dagger$ denotes the (Moore–Penrose) matrix pseudoinverse [46]), then $\hat{\mathbf{y}}(z) = \mathbf{y}(z)$ is guaranteed if the subband model $\mathbf{S}(z)$ is chosen as follows:

$$\mathbf{S}(z) = \mathbf{F}^{*\dagger}(z)\mathbf{G}(z)\mathbf{H}^\dagger(z). \qquad (13)$$

The number of computations required to process a subband signal using $\mathbf{S}(z)$ depends on the number l_S of nonzero entries. A drawback of (13) is that the obtained $\mathbf{S}(z)$ has a large number of nonzero entries. In [47], this issue is addressed by using sparse approximation techniques to solve

$$\mathbf{S} = \arg\min_{\mathbf{S}} \|\mathbf{G} - \mathbf{F}^*\mathbf{S}\mathbf{H}\|_{\mathbf{W}} \qquad (14)$$

where $\|\mathbf{X}\| = \langle \mathbf{X}, \mathbf{X} \rangle_{\mathbf{W}}^{1/2}$

$$\langle \mathbf{X}, \mathbf{Y} \rangle_{\mathbf{W}} = \frac{1}{2\pi D} \int_{-\pi}^{\pi} \text{Tr}\{\mathbf{X}(e^{j\omega})\mathbf{W}(e^{j\omega})\mathbf{Y}^*(e^{j\omega})\} d\omega \qquad (15)$$

and $\mathbf{W}(z)$ is the polyphase representation of a user-supplied spectral weighting function $w(z)$. We summarize this result below, for a more detailed presentation see [47].

For each $m, n = 1, \ldots, M$ and $\tau \in \mathbb{Z}$, define the $D \times D$ transfer matrix

$$\mathbf{V}_{m,n,\tau}(z) = \mathbf{F}^*(z)\mathbf{U}_{m,n,\tau}(z)\mathbf{H}(z) \qquad (16)$$

where the $M \times M$ impulse response $\mathbf{U}_{m,n,\tau}(t)$ is defined by

$$[\mathbf{U}_{m,n,\tau}(t)]_{i,j} = \begin{cases} 1, & i = m, \ j = n, \ t = \tau \\ 0, & \text{otherwise} \end{cases}.$$

Then, $\mathbf{S}(z)$ is computed using the following iterative procedure: let $\hat{\mathbf{G}}^{(0)} = 0$. Then, at iteration k we compute

$$\left(m^{(k)}, n^{(k)}, \tau^{(k)}\right)$$
$$= \arg\max_{(m,n,\tau)} \frac{\left|\left\langle \mathbf{G} - \hat{\mathbf{G}}^{(k-1)}, \mathbf{V}_{m,n,\tau} \right\rangle_{\mathbf{W}}\right|}{\|\mathbf{V}_{m,n,\tau}\|_{\mathbf{W}}} \qquad (17)$$

$$\hat{\mathbf{G}}^{(k)} = \mathbf{F}^*\hat{\mathbf{S}}^{(k)}\mathbf{H} \qquad (18)$$

$$\hat{\mathbf{S}}^{(k)} = \arg\min_{\text{supp}\{\mathbf{S}\} = \mathcal{S}^{(k)}} \|\mathbf{G} - \mathbf{F}^*\mathbf{S}\mathbf{H}\|_{\mathbf{W}} \qquad (19)$$

where $\mathcal{S}^{(k)}$ is the set that includes the indexes $\{(m_l, n_l, \tau_l) : l = 1, \ldots, k\}$, and $\text{supp}\{\mathbf{S}\} = \mathcal{S}^{(k)}$ means that the minimization with respect to $\mathbf{S}$ is done over the matrices having zeros outside $\mathcal{S}^{(k)}$.

C. Choice of Filterbanks

We need to provide choices for the filterbank prototypes $h_1(z)$ and $f_1(z)$. In the context of subband adaptive filtering, it was pointed out in [29] that a diagonal subband model leads to the most efficient subband configuration. In view of this, we choose $h_1(z)$ and $f_1(z)$ so that the nonzero entries of the subband concentrate on the main diagonal as much as possible. To this end, we point out the following fact which follows from ([31], Theorem 1).

Lemma 1: If the frequency response of the analysis filters $h_m(z)$, $m = 1, \ldots, M$ and the synthesis filters $f_m(z)$, $m = 1, \ldots, M$ satisfy the following.

(C1) For each $m = 1, \ldots, M$, the supports of $h_m(e^{j\omega})$ and $f_m(e^{j\omega})$ are contained in the same interval σ_m of measure $2\pi/D$.

(C2) The union $\cup_{m=1}^{M} \sigma_m$ of all M supports cover the whole interval $[-\pi, \pi]$.

Then, the approximation error can be made arbitrarily small using a diagonal subband model of sufficiently large tap size.

In view of Lemma 1, $h_1(z)$ and $f_1(z)$ need to minimize their stop-band energy. Hence, we design $h_1(z)$ as follows:

$$h_1(z) = \arg\min_{h_1(z):h_1(1)=1} \int_{\pi/D}^{2\pi - \pi/D} |h_1(e^{j\omega})|^2 \, d\omega$$

and we choose $f_1(z) = h_1(z)$.

V. COMPARISON BETWEEN SUBBAND AND OTHER AVAILABLE METHODS

A. Inverse FFT Method as a Particular Case of the Subband Method

In this section, we shall show that the IFFT method for synthesizing stochastic sounds (described in Section II) can be seen as a particular case of the subband method (described in Section IV-B).

The vector signal $\mathbf{v}(z)$ in (3) can be written as

$$\mathbf{v}(z) = \boldsymbol{\Delta}_g \mathbf{w}(z) \qquad (20)$$

where the $M \times M$ diagonal matrix $\boldsymbol{\Delta}_g$ is defined by

$$\boldsymbol{\Delta}_g = \mathrm{diag}\left\{ |g(1)|, \left| g\left(e^{j2\pi \frac{1}{M}}\right) \right|, \ldots, \left| g\left(e^{j2\pi \frac{M-1}{M}}\right) \right| \right\}$$

Also, using (9) and (10), (1) and (2) can be jointly written in the polyphase domain as

$$\mathbf{y}(z) = \mathbf{B}_{M,D}^*(z)\boldsymbol{\Lambda}_\sigma \mathcal{W}_M^{-1} \mathbf{v}(z) \qquad (21)$$

where $\mathbf{y}(z)$ denotes the polyphase representation of the signal $y(z)$, and the $M \times M$ diagonal matrix $\boldsymbol{\Lambda}_\sigma$ is defined by

$$\boldsymbol{\Lambda}_\sigma = \mathrm{diag}\{\sigma(0), \sigma(1), \ldots, \sigma(M-1)\}.$$

We want to show that (21)-(20) is a particular case of (12). From (8), we have that $\mathbf{A}_{M,M} = \mathbf{I}_M$. Also, $\overline{\sigma(t)} = \sigma(t)$ implies that $\boldsymbol{\Lambda}_\sigma^* = \boldsymbol{\Lambda}_\sigma$. Then, from (21)-(20) and (7) we have that

$$
\begin{aligned}
\mathbf{y}(z) &= \mathbf{B}_{M,D}^*(z)\boldsymbol{\Lambda}_\sigma \mathcal{W}_M^{-1}\boldsymbol{\Delta}_g \mathbf{w}(z) \\
&= \mathbf{B}_{M,D}^*(z)\boldsymbol{\Lambda}_\sigma^* \mathbf{A}_{M,M}^* \mathcal{W}_M^* \boldsymbol{\Delta}_g \mathbf{w}(z) \\
&= \boldsymbol{\Sigma}^*(z)\boldsymbol{\Delta}_g \mathbf{w}(z)
\end{aligned}
\qquad (22)
$$

where $\boldsymbol{\Sigma}^*(z)$ denotes the polyphase matrix of a Gabor filterbank with prototype $\sigma(z)$.

Suppose now that $x(t)$ in (12) is a white random process. Then, the correlation matrix $\mathbf{R}_{\boldsymbol{\xi}}(k) = \mathcal{E}\{\boldsymbol{\xi}(0)\overline{\boldsymbol{\xi}(k)^T}\}$ of the subband signal $\boldsymbol{\xi}(k)$ is given by

$$
\begin{aligned}
\mathbf{R}_{\boldsymbol{\xi}}(k) &= \mathcal{Z}^{-1}\{\mathbf{H}(z)\boldsymbol{\Phi}_{\mathbf{x}}(z)\mathbf{H}^*(z)\}(k) \\
&= \mathcal{Z}^{-1}\{\mathbf{H}(z)\mathbf{H}^*(z)\}(k) \\
&= \mathcal{Z}^{-1}\{\mathbf{h}(z)\overline{\mathbf{h}^T(z)}\}(kD)
\end{aligned}
\qquad (23)
$$

where $\boldsymbol{\Phi}_{\mathbf{x}}(z) = \mathbf{I}$ denotes the PSD of $\mathbf{x}(t)$. The (m,n)th entry of $\mathbf{h}(z)\overline{\mathbf{h}^T(z)}$ is given by $h_m(z)\overline{h_n(z)}$. Therefore, if the filters $h_m(z)$, $m = 1, \ldots, M$ have narrow pass-bands, then $\mathbf{R}_{\boldsymbol{\xi}}(k)$ is close to the identity matrix up to a constant gain. Intuitively speaking, this means that $\boldsymbol{\xi}(k)$ is an "almost white" vector random process.

Hence, from (22) and (23), we conclude that the IFFT method can be interpreted as the subband method where:

(C1) the synthesis filterbank prototype is chosen as $f(z) = \sigma(z)$, and therefore the support of its impulse response is constrained to the interval $[0, M-1]$;

(C2) the subband model is designed using the diagonal matrix $\mathbf{S}(z) = \boldsymbol{\Delta}_g$, instead of the design proposed in Section IV-B;

(C3) the input of $\mathbf{S}(z)$ is the white random process $\mathbf{w}(t)$ instead of the "almost white" random process $\boldsymbol{\xi}(t)$ with correlation matrix $\mathbf{R}_{\boldsymbol{\xi}}(k)$ given by (23).

It is therefore clear that the subband method is more flexible than the IFFT method, in the sense that it permits dropping the constraints (C1), (C2), and (C3). In Section VI, we show how this extra flexibility can be exploited to obtain a better tradeoff between time and frequency resolution at the expenses of some extra computations.

B. Numerical Example

In this section, we introduce an example to illustrate how the subband method can be used to achieve simultaneously good time and frequency resolutions, at the expenses of some extra computations, when compared to the IFFT method. We generate colored noise with a prescribed spectral shaping function $g(z)$. To this end we use the output of a digital waveguide model, which permits simulating a wave propagation in a bounded medium [48]. It is formed by a loop system containing a pure delay $\Psi(z) = e^{-zd}$ representing the average propagation time needed by the wave to go back and forth between the extremities of the medium, and a linear filter $\Upsilon(z)$ representing the dissipation and the dispersion phenomena. We have chosen $d = 4.5$ ms which corresponds to a tube of 75 cm long, and $\Upsilon(t)$ to be a linear up-chirp function with frequency variation rate 2×10^6 Hz/s., whose amplitude decays exponentially with time-constant 1.3 ms. This simulates a highly dissipative and dispersive medium. The corresponding impulse and frequency responses are shown in Fig. 3. We compared the performances between the IFFT and the subband methods in terms of the spectral approximation error, which is measured by

$$
e = \frac{1}{K}\sum_{k=0}^{K-1}\left| 10\log_{10}\phi_y\left(\frac{2\pi f_c k}{K}\right) - 10\log_{10}\phi_{\hat{y}}\left(\frac{2\pi f_c k}{K}\right)\right|^2
\qquad (24)
$$

where $f_c = 6$ kHz, $K = 8192$, $\phi_y(\omega) = |g(e^{j\omega})|^2$ denotes the target theoretical PSD and $\phi_{\hat{y}}(\omega)$ is the PSD of 10 seconds of synthesized noise, estimated using Welch's averaged modified periodogram spectral estimation method [49].

For the IFFT method we use different values of the synthesis window size M and a synthesis hop size of $D = M/4 = 32$. We design the synthesis window $\sigma(t)$ using a digital prolate spheroidal sequence of order 0 and bandwidth $7\pi/M$ [50] (i.e., an FIR bandpass filter maximizing its energy inside the support $[-(7\pi)/(M), (7\pi)/(M)]$, while meeting a fixed energy constraint). This guarantees that, after multiplying $\sigma(t)$ by a proper scaling factor, condition (4) is satisfied within a tolerance of 0.4 dB.

For the subband (SB) method we choose $M = 128$. To achieve a good time resolution we choose a tap size of $l_f = 128$ for the synthesis prototype $f_1(z)$. As described in Section IV-C, $f_1(z)$ needs to have a good attenuation outside its passband $[-(\pi/D), (\pi/D)]$. We choose a downsampling factor of $D = 32$, which results in an attenuation outside $[-(\pi/D), (\pi/D)]$ being 50 dB smaller than the peak response (a better attenuation can be achieved by choosing a smaller value for D). As mentioned in Section IV-C, we choose $h_1(z) = f_1(z)$. The subband model $\mathbf{S}(z)$ is designed using the iterative algorithm
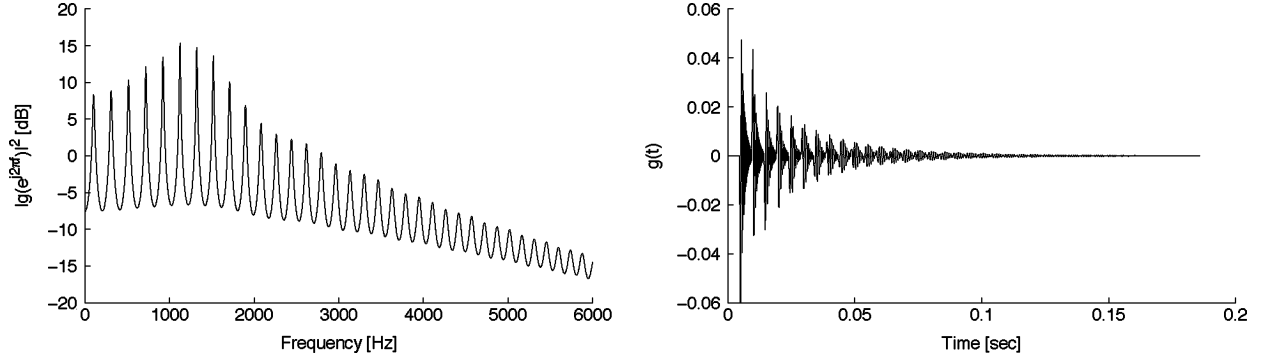
Fig. 3.   Target PSD $|g(e^{j2\pi f})|^2$ (left) and impulse response $g(t)$ of the target spectral envelope $g(z)$ (right).
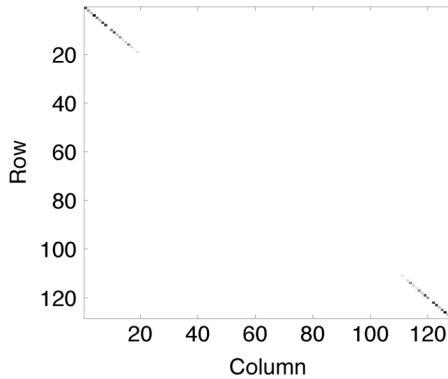

Fig. 4. Energy distribution of the entries of $\mathbf{S}(z)$. The energy of the (m,n)-entry is measured by $10 \times \log_{10}(\sum_{k=-\infty}^{\infty} |[\mathbf{S}(k)]_{m,n}|^2)$.

Fig. 5. PSDs of the signals obtained using the time-domain (TD), inverse FFT (IFFT) and subband (SB) methods.

(17)–(19), where in view of (24), the spectral weight $w(z)$ is chosen as a low-pass filter with cutoff frequency 6 kHz. The iterations are stopped when the approximation error becomes smaller than 0.5, resulting in $\mathbf{S}(z)$ having 438 coefficients. The energy distribution of the entries of $\mathbf{S}(z)$ is shown in Fig. 4.

Following the explanation in Section III, the computation of the analysis filterbank requires 60 multiplications per sample, and the same cost applies for the computation of the synthesis filterbank. Also, since $g(z)$ is real-valued, only half the entries of $\mathbf{S}(z)$ need to be computed. Hence, the computation of $\mathbf{S}(z)$ requires 27 multiplications per sample, and the overall complexity of the subband method is of 147 multiplications per sample.

In Fig. 5, we show the PSD of the signals obtained using the SB method, as well as the IFFT method with $M = 128$ and $M = 8192$. We also include in the comparison the time-domain (TD) method obtained by directly convolving the input signal with the 8192-tap impulse response shown in Fig. 3(right). In Table I we compare the performances of these methods in term of spectral approximation error, synthesis window length, and complexity. We see that the SB method offers an approximation error (i.e., a frequency resolution) similar to that of the 8192-point IFFT method with a synthesis window length (i.e., a time resolution) similar to that of the 128-point IFFT method, at the expense of some extra computations. Also, while the TD achieves the best approximation error and a time resolution of only one sample, it requires a very high number of computations when compared with the IFFT and the SB method.
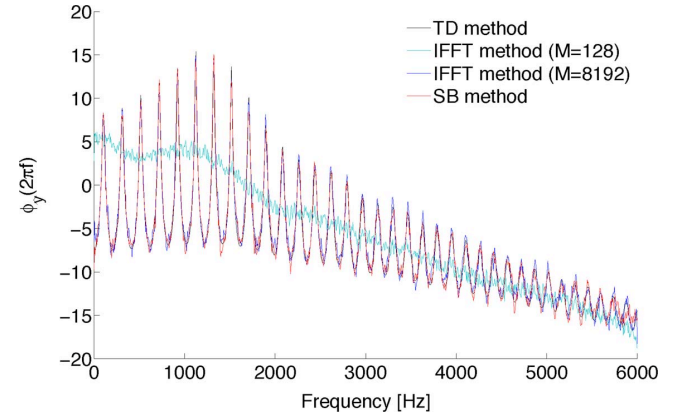
TABLE I
PERFORMANCE COMPARISON OF THE TIME-DOMAIN (TD), INVERSE FFT (IFFT), AND SUBBAND (SB) METHODS

	Approximation error	Synthesis window length [samples]	Complexity mult/sample
IFFT ($M = 128$)	23.25	128	60
IFFT ($M = 8192$)	0.5548	8192	108
SB	0.4706	128	147
TD	0.3752	1	8192

The sound examples are available at http://www.lma.cnrs-mrs.fr/~kronland/TFSynth/sounds.html.

C. About the Use of Spectral Line Broadening

As mentioned in Section I, a narrow noise component can be generated using the spectral line broadening (SLB) technique. This technique consists in modulating the phase of a sinusoid with a random signal having the spectral shape of the desired noise component. A limitation of this technique, when used in combination with the IFFT method, is that the frequency of the sinusoid can only be changed once every D samples. Hence, only very narrow noise components can be generated.

To illustrate this point we first generate narrowband noise whose power spectrum has the shape of a raised cosine function centered at 1723 Hz and having a bandwidth of 86 Hz. For the subband method we design the subband model $\mathbf{S}(z)$ using the iterative algorithm (17)–(19), where the iterations are stopped once the attenuation outside the corresponding passband is 50
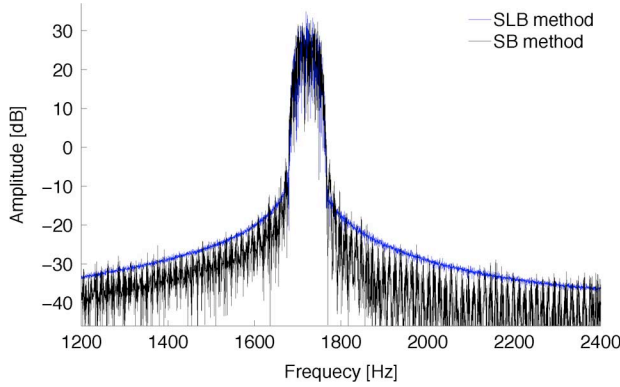
Fig. 6. Spectra of narrowband ($\mathrm{bandwidth} = 86$ Hz) noise generated using the SLB and the SB methods.
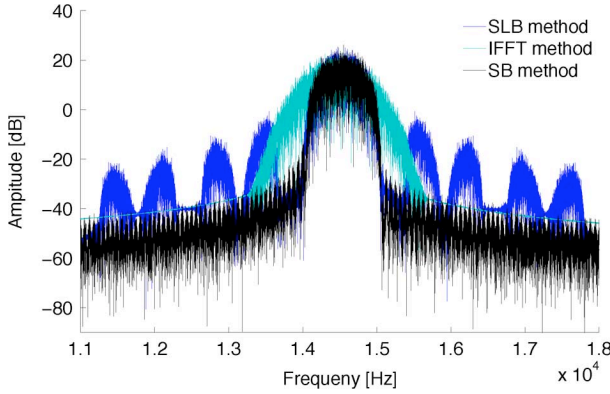

Fig. 7. Spectra of wideband ($\mathrm{bandwidth} = 519$ Hz) noise generated using the SLB and the SB methods.

dB smaller than its peak response. The resulting subband model has 103 nonzero (complex) parameters, and therefore its implementation requires about 13 multiplications per sample. For the SLB method we use a spectral-shaping filter of 412 taps, leading to a complexity equivalent to that of the SB method. We see in Fig. 6 that the spectra of the signals generated using both methods are approximately equivalent. However, this equivalence is broken when generating noise with a larger bandwidth. This can be seen in Fig. 7 where we generated a raised cosine-shaped noise band centered at 14 529 Hz and having bandwidth 519 Hz. The SLB method is not able generate this band, due to its large bandwidth.

Suppose we want to go around this limitation, without using the SB method, and while keeping a good time resolution (i.e., a short synthesis window). In principle, a possibility could be to use a hybrid approach in which narrow bands are generated using the SLB method while wide bands are generated using the 128-point IFFT method. However, the flexibility of this approach is limited. This is shown in Fig. 7 where we see that the generated noisy band is not only too wide for being synthesized by the SLB method, but also too narrow for being synthesized by the 128-point IFFT method.
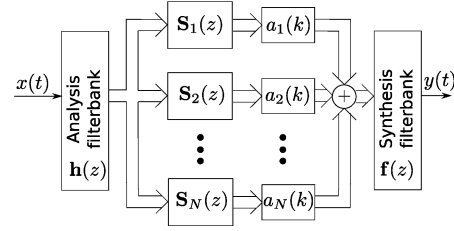

Fig. 8. Scheme for subband-based stochastic sound synthesis.

VI. SCHEME FOR SYNTHESIZING NON-STATIONARY STOCHASTIC SOUNDS USING SUBBANDS

In a practical scenario, a target stochastic signal $y(t)$ is characterized by a time-varying power spectrum, or equivalently, as the output of a time-varying filter (spectral shaping function) with white noise at its input. A drawback of the subband synthesis technique described in Section IV is that the design of the subband model $\mathbf{S}(z)$ corresponding to a given spectral shaping function $g(z)$ is very expensive. In this section, we use the result from Section IV to propose a scheme for synthesizing nonstationary stochastic sounds which avoids the aforementioned drawback. The design is presented in Section VI-A. In Section VI-B, we study its time and frequency resolutions, and compare them with those of the IFFT method. Finally, we evaluate the performance of the proposed design for synthesizing environmental sounds in Section VI-C.

A. Proposed Design

The idea consists of designing a set of subband models to approximate a set of bandpass filters with contiguous passbands covering the frequency range from 0 Hz to 20 kHz. We use a sampling frequency of $f_s = 44.1$ kHz and a subband configuration of $M = 128$ subbands. To achieve a time resolution of 128 samples, we choose the prototypes $h_1(z)$ and $f_1(z)$ as described in Section V-B. To achieve a good frequency resolution we use $N = 134$ frequency bands. The band allocation is done by dividing the frequency range from 0 Hz to 2 kHz into 46 bands uniformly distributed in the linear scale, and dividing the range from 2 kHz to 20 kHz in 88 bands uniformly distributed in the ERB scale. For each $n = 1, \ldots, N$, we design a subband model $\mathbf{S}_n(z)$ which approximates a bandpass filter $g_n(z)$ having its passband on the nth band (notice that we consider complex filters $g_n(z)$, $n = 1, \ldots, N$ as having their passbands in the positive frequencies). Then, a time-varying spectrum is obtained by multiplying the output of each subband model $\mathbf{S}_n(z)$ by a time-varying gain $a_n(k)$. The resulting scheme is depicted in Fig. 8.

As in (4), we need to guarantee that white noise is synthesized when $a_n(k) = 1$, for all $k \in \mathbb{Z}$ and all $n = 1, \ldots, N$. To achieve this, we choose $g_n(z)$, $n = 1, \ldots, 46$ as the following root raised cosine filters:

$$g_n\left(e^{j2\pi(f-n+1)f_0/f_s}\right) = \begin{cases} \sqrt{\frac{1}{2}(1+\cos(\pi f))}, & |f| < \frac{1}{2} \\ 0, & \text{otherwise} \end{cases}$$

(25)

with $f_0 = 43$ Hz, and we choose the filters $g_n(z)$, $n = 47, \ldots, N$ (i.e., those having their passbands above
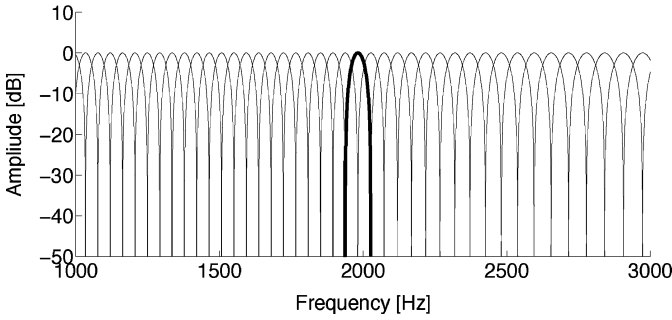
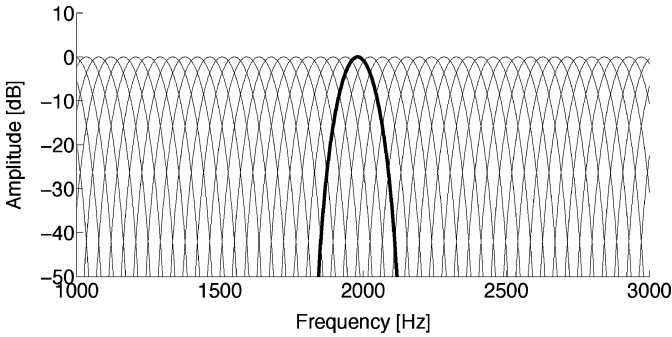Fig. 9. Frequency response of the bandpass filters defined in (25).

Fig. 10. Frequency response of the bandpass filters obtained when using the IFFT method.

Fig. 11. Number of coefficients l_n of the subband model $\mathbf{S}_n(z)$ as a function of n.

Fig. 12. Spectrogram of a glass impact sound synthesized using the subband scheme of Fig. 8.

2 kHz) as in (25) but replacing the linear scale by the ERB scale. The frequency responses of some of the resulting filters are shown in Fig. 9. For comparison purposes, we show in Fig. 10 the bandpass filters corresponding to each frequency band of the IFFT method with $M = 1024$ designed as described in Section V-B. Notice that in Fig. 9, the frequency response of all but the nth filter is zero at the center f_n of the nth frequency band. This property guarantees that the sequence $a_n(k)$ controls the amplitude at f_n, and is not satisfied by the filters in Fig. 10.

As in Section V-B, the total computational cost of the analysis and synthesis filterbanks is of 120 multiplications per sample. The complete synthesis scheme consists of $N = 134$ subband models $\mathbf{S}_n(z)$, $n = 1, \ldots, N$, each of which approximates a passband filter. For each n, the subband models $\mathbf{S}_n(z)$ is designed using the iterative algorithm (17)–(19), where the iterations are stopped once the attenuation outside the corresponding passband is 50 dB smaller than its peak response. The resulting number of coefficients l_n of the nth subband model is shown as a function of n in Fig. 11. The total number of subband coefficients is $l_S = \sum_{n=1}^{N} l_n = 10572$. Hence, as explained in Section IV-B, the computation of the N subband models $\mathbf{S}_n(z)$, $n = 1, \ldots, N$ require 1322 multiplications per sample. Finally, the N real-valued gains $a_n(k)$ need to be applied, once every $D = 32$ samples, at the output of each subband model $\mathbf{S}_n(z)$. The application of these gains requires about 12 multiplications per sample, since the output of most subband models span only one subband. Hence, the overall computational cost of the proposed subband synthesis scheme is 1454 multiplications per sample. This complexity is significantly larger than those of the
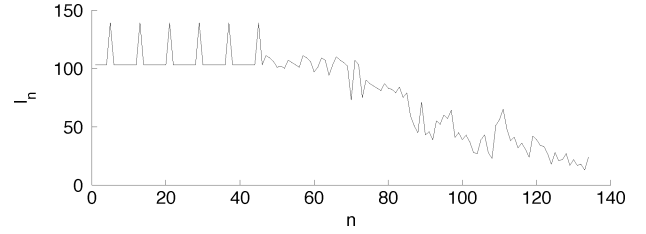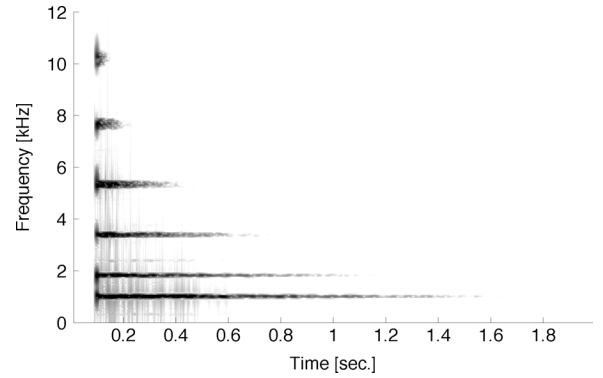
IFFT method shown in Table I. However, notice that the proposed scheme not only achieves simultaneously good time and frequency resolutions, but also, as shown in Fig. 9, it permits an accurate control of the amplitude of each frequency band. The complexity of this scheme can be alleviated by either reducing the number of frequency bands or relaxing the design of the bandpass filters $g_n(z)$, $n = 1, \ldots, N$ (e.g., to resemble the shape of a digital prolate spheroidal sequence as the bands generated by the IFFT method). To give an idea of the computational savings obtained by using subband processing, a time-domain implementation, where the input signal is directly convolved with the filters $g_n(z)$ (having the same stopband attenuation than $\mathbf{S}_n(z)$), would require about 10^6 multiplications per sample.

B. Evaluation of Time and Frequency Resolutions

In this section, we study the time and frequency resolutions of the subband scheme designed in Section VI-A. To this end we synthesize a blurry sound effect on a glass impact which presents both, a short transient at the instant of the impact, and narrow spectral components during the decaying sound. The impact is applied at 0.1 s, and the blurry sound model consists of narrow (noisy) frequency bands at 1051, 1849, 3388, 5339, 7606, and 10163 Hz, whose amplitudes decay exponentially with time constants 4.948, 6.397, 10.78, 21.26, 47.49, and 110.4, respectively. The spectrogram of the sound synthesized using the proposed subband scheme is shown in Fig. 12. The time resolution of a synthesis scheme is given by the length of the synthesis window. This is because this length determines the time span on which a modification of the synthesis coefficients will have its effect. In the case of the subband synthesis
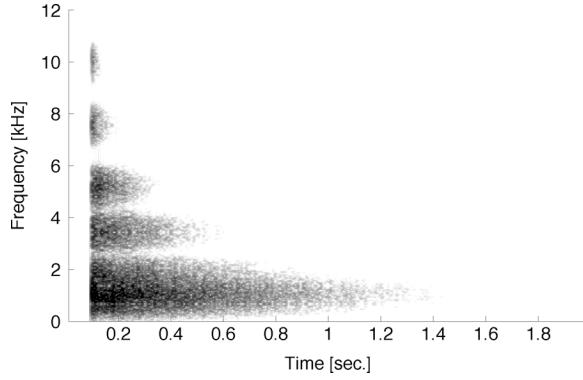
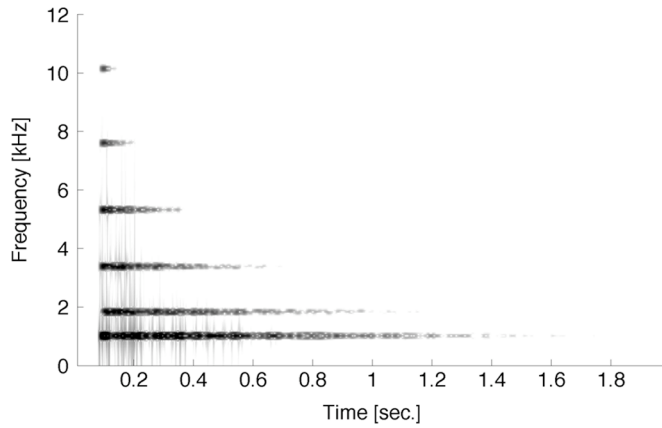Fig. 13. Spectrogram of a glass impact sound synthesized using the IFFT method with $M = 128$.

Fig. 15. Time-domain glass impact sound synthesized using the subband scheme of Fig. 8.

Fig. 14. Spectrogram of a glass impact sound synthesized using the IFFT method with $M = 1024$.

Fig. 16. Time-domain glass impact sound synthesized using the IFFT method with $M = 1024$.

scheme, this is given by the tap size l_f of the synthesis prototype $f_1(z)$, i.e., 128 samples. The same time resolution is obtained using the IFFT method with $M = 128$ points. However, as shown in Fig. 13, this leads to a very poor frequency resolution. A frequency resolution similar to that of the subband synthesis scheme can be obtained using the IFFT method with $M = 1024$ points. This is shown in Fig. 14. However, this leads to a very poor time resolution, due to the long synthesis window.

To illustrate this point in more detail, we compare in Fig. 15 the impact transient response of the subband method with a reference impact transient waveform. This reference waveform is obtained by directly applying the desired subband gains $a_n(k)$ in the time domain, i.e., after the synthesis filterbank, hence avoiding the delay introduced by the synthesis operation. A similar comparison for the IFFT method is given in Fig. 16, showing its slower transient response.

In order to quantify the quality of the transient response we use

$$\varepsilon = \frac{\sum_{t=1}^{T}(y_r(t) - y(t))^2}{\sum_{t=1}^{T} y_r(t)^2}$$

where $y_r(t)$ and $y(t)$ denote the reference and synthesized signals, respectively. Using this measure on the transient responses
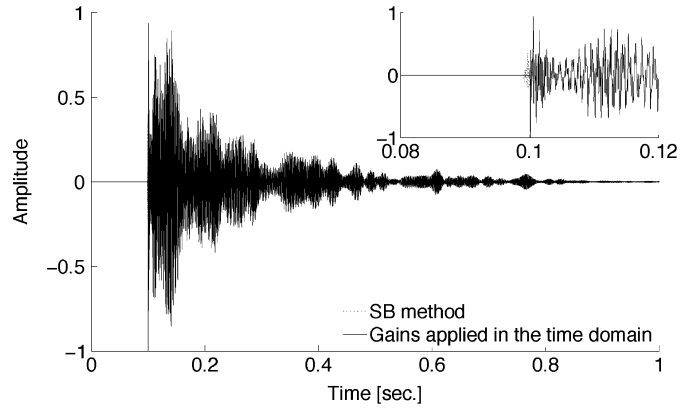
shown in Figs. 15 and 16 we obtain $\varepsilon = -27.15$ dB for the subband method and $\varepsilon = -10.32$ dB for the IFFT method.

The reader can verify that the relevant aspects of the blurry effect on the glass impact sound (i.e., short transient and narrow components) were simultaneously reproduced by the subband method but not by the IFFT method, by listening at the synthesized sounds which can be found at http://www.lma.cnrs-mrs.fr/~kronland/TFSynth/sounds.html.

C. Environmental Sound Examples

In this section, we evaluate the performance of the proposed subband scheme when synthesizing environmental sounds. In the first example, we synthesize fire sound, on which three main contributions are identified. The first is the combustion, which is a narrow component at very low frequency. The second is the hissing, which consists of very narrow components at relatively high frequency (around 5 and 7.5 kHz), and the third is formed by the cracklings, which present very short transients. The spectrogram of the sound synthesized using the proposed subband scheme is shown in Fig. 17. A 128-point IFFT synthesis achieves a similar time resolution which is good enough for reproducing the crackling sounds. However, as shown in Fig. 18, the frequency resolution of this method is not enough for reproducing neither the combustion nor the hissing sound. Again, as shown in Fig. 19, these sounds are accurately synthesized using
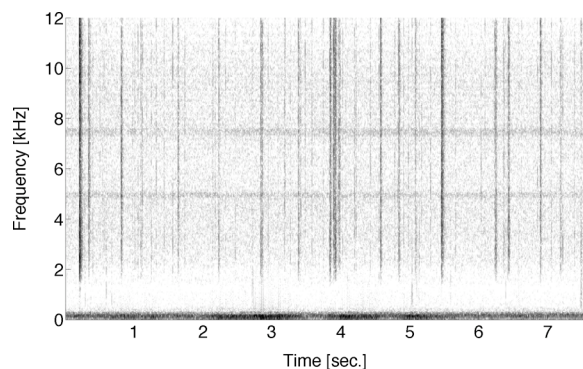
Fig. 17. Spectrogram of fire sound synthesized using the subband scheme of Fig. 8.
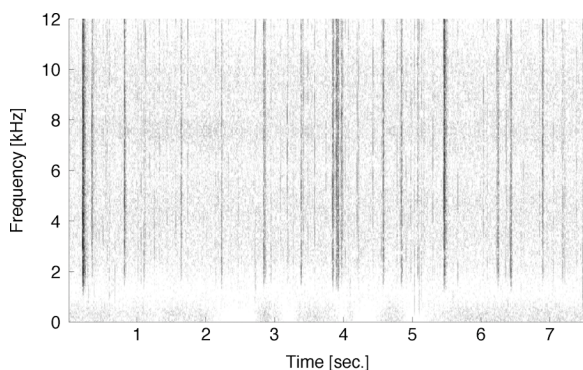

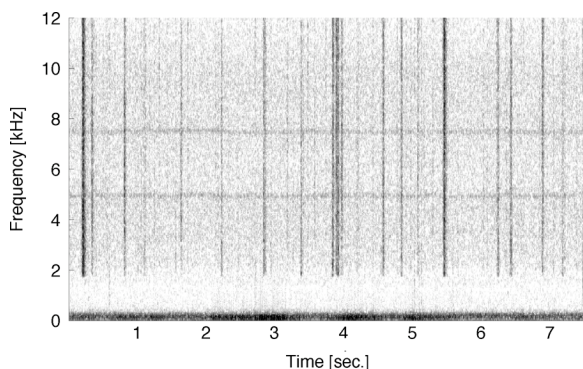Fig. 18. Spectrogram of fire sound synthesized using the IFFT method with $M = 128$.

Fig. 19. Spectrogram of fire sound synthesized using the IFFT method with $M = 1024$.

a 1024-point IFFT method, but as shown in Fig. 20, this method suffers from a poor time resolution which is not enough for synthesizing crackling sounds.

Again, the reader can verify that the three components of the fire sound (i.e., combustion, crackling and hissing) were simultaneously reproduced by the subband method and not by the IFFT method, by listening at the sound examples at http://www.lma.cnrs-mrs.fr/~kronland/TFSynth/sounds.html. We have also made available in that web page other environmental sound examples including wave, wind, whoosh, stones, etc.
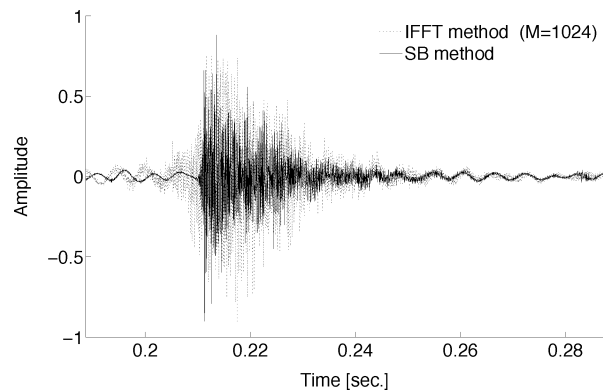

Fig. 20. Time-domain crackling sound synthesized using the IFFT method with $M = 1024$ and the subband scheme of Fig. 8.

VII. CONCLUSION

Colored noise is an important aspect in the synthesis of musical and environmental sounds. Frequency-based synthesis methods, such as the IFFT method, substantially reduce the complexity of the additive model but suffer from an inherent tradeoff between time and frequency resolution. This severe constraint limits the use of this approach for synthesizing natural sounds where both transient and colored noises may coexist. In the aim of designing an environmental sound synthesizer based on a unique algorithm, we have proposed the subband (time–frequency domain) method as an alternative to the IFFT method. This method takes into account correlations in the whole time–frequency plane and can then be seen as a generalization of the IFFT method, offering extra design flexibility. To illustrate the potential of the subband method, we present synthesis examples clearly showing how the tradeoff between time and frequency resolution has been improved compared to the IFFT case, for which the time window size induces the frequency resolution. Thus, contrarily to the IFFT, we have shown that the subband method allows for synthesizing narrow band colored noises by using short windows compatible with the synthesis of audio transient signals.

REFERENCES

[1] C. Roads, *The Computer Music Tutorial*, 5th ed. Cambridge, MA: MIT Press, 2000.
[2] Y. Dobashi, T. Yamamoto, and T. Nishita, "Real-time rendering of aerodynamic sound using sound textures based on computational fluid dynamics," *ACM Trans. Graphics (Proc. SIGGRAPH)*, vol. 22, no. 3, pp. 732–740, 2003.
[3] Y. Dobashi, T. Yamamoto, and T. Nishita, "Synthesizing sound from turbulent field using sound textures for interactive fluid simulation," *EUROGRAPHICS*, vol. 23, no. 3, pp. 539–546, 2004.
[4] C. Stoelinga and A. Chaigne, "Time-domain modeling and simulation of rolling objects," *Acustica United With Acta Acustica*, vol. 93, no. 2, pp. 290–304, 2007.
[5] P. R. Cook, "Physically informed sonic modeling (phism): Synthesis of percussive sounds," *Comput. Music J.*, vol. 21, no. 3, pp. 38–49, 1997.
[6] K. van den Doel, P. G. Kry, and D. K. Pai, "Foleyautomatic: Physically-based sound effects for interactive simulation and animation," in *Proc. 28th Annu. Conf. Comput. Graphics Interactive Tech.*, 2001, pp. 537–544.
[7] K. van den Doel, "Physically-based models for liquid sounds," in *Proc. ICAD 04-10th Meeting Int. Conf. Auditory Display*, 2004.
[8] A. Farnell, *Designing Sound, Procedural Audio for Games and Film.*. Dorset, U.K.: Applied Scientific Press, 2008.

[9] 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.

[10] X. Serra and J. O. Smith, "Spectral modeling synthesis: A sound analysis/synthesis system based on a deterministic plus stochastic decomposition," *Comput. Music J.*, vol. 14, no. 4, pp. 12–24, 1990.

[11] A. Misra, P. R. Cook, and G. Wang, "A new paradigm for sound design," in *Proc. Int. Conf. Digital Audio Effects (DAFx06)*, 2006.

[12] T. S. Verma and T. H. Y. Meng, "Extending spectral modeling synthesis with transient modeling synthesis," *Comput. Music J.*, vol. 24, no. 2, pp. 47–59, 2000.

[13] F. Nsabimana and U. Zolzer, "Audio signal decomposition for pitch and time scaling," in *Proc. 3rd Int. Symp. Commun., Control, Signal Process.*, 2008.

[14] P. Hanna and M. Desainte-Catherine, "A statistical and spectral model for representing noisy sounds with short-time sinusoids," *EURASIP J. Appl. Signal Process.*, vol. 5, no. 12, pp. 1794–1806, 2005.

[15] K. Fitz, L. Haken, and P. Christensen, "Transient preservation under transformation in an additive sound model," in *Proc. Int. Comput. Music Conf.*, 2000.

[16] K. Fitz and L. Haken, "Bandwidth enhanced sinusoidal modeling in lemur," in *Proc. Int. Comput. Music Conf.*, 1995.

[17] J. Chowning, "The synthesis of complex audio spectra by means of frequency modulation," *JAES*, vol. 21, no. 7, pp. 526–534, 1973.

[18] M. L. Brun, "Digital waveshaping synthesis," *JAES*, vol. 27, no. 4, pp. 250–266, 1979.

[19] X. Rodet and P. Depalle, "Spectral envelopes and inverse FFT synthesis," in *Proc. 93rd AES Conv.*, 1992.

[20] N. Meine and H. Purnhagen, "Fast sinusoid synthesis for MPEG-4 HILN parametric audio decoding," in *Proc. Int. Conf. Digital Audio Effects (DAFx02)*, Hamburg, Germany, 2002, pp. 239–244.

[21] X. Rodet and D. Schwarz, "Spectral envelopes and additive + residual analysis/synthesis," in *Analysis, Synthesis, and Perception of Musical Sounds: Sound of Music*. New York: Springer, 2007.

[22] M. Robine, R. Strandh, and S. Marchand, "Fast additive sound synthesis using polynomials," in *Proc. Int. Conf. Digital Audio Effects (DAFx06)*, 2006, pp. 181–186.

[23] X. Amatriain, J. Bonada, A. Loscos, and X. Serra, "Spectral processing," in *DAFX: Digital Audio Effects*. New York: Wiley, 2002.

[24] X. Serra, "Musical sound modeling with sinusoids plus noise," in *Musical Signal Processing.*. Lisse, The Netherlands: Swets & Zeitlinger, 1997, pp. 91–122.

[25] M. Goodwin, "Residual modeling in music analysis-synthesis," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 1996, pp. 1005–1008.

[26] C. Verron, M. Aramaki, R. Kronland-Martinet, and G. Pallone, "Spatialized additive synthesis of environmental sounds," in *Proc. 125th AES Conv.*, 2008.

[27] A. Freed, "Real-time inverse transform additive synthesis for additive and pitch synchronous noise and sound spatialization," in *Proc. 104th AES Conv.*, 1998.

[28] A. Freed, "Spectral line broadening with transform domain additive synthesis," in *Proc. Int. Comput. Music Conf.*, 1999.

[29] A. Gilloire and M. Vetterlli, "Adaptive filtering in subbands with critical sampling: Analysis, experiments, and application to acoustic echo cancellation," *IEEE Trans. Signal Process.*, vol. 40, no. 8, pp. 1862–1875, 1992.

[30] Y. Lu and J. Morris, "Gabor expansion for adaptive echo cancellation," *IEEE Signal Process. Mag.*, vol. 16, no. 2, pp. 68–80, Mar. 1999.

[31] D. Marelli and M. Fu, "Performance analysis for subband identification," *IEEE Trans. Signal Process.*, vol. 51, no. 12, pp. 3128–3142, Dec. 2003.

[32] J.-J. Sung, G.-S. Kang, and S. Kim, "A transient noise model for frequency-dependent noise sources," *IEEE Trans. Comput.-Aided Design AD Integr. Circuits Syst.*, vol. 22, no. 8, pp. 1097–1104, Aug. 2003.

[33] W. Hartmann, Signal, Sound and Sensation. 2004, Amer. Inst. of Phys.

[34] W. Kellermann, "Kompensation akustischer echos in frequenzteilbaendern," *Frequenz*, 1985.

[35] Y. Itoh, Y. Maruyama, N. Furuya, and T. Areseki, "Acoustic echo canceller for teleconference," in *Int. Conf. Commun.*, 1985.

[36] D. Marelli and M. Fu, "A subband approach to channel estimation and equalization for DMT and OFDM systems," *IEEE Trans. Commun.*, vol. 53, no. 11, pp. 1850–1858, Nov. 2005.

[37] J. P. Seon, H. Y. Jeong, C. P. Young, and H. Y. Dae, "A delayless subband active noise control system for wideband noise control," *IEEE Trans. Speech Audio Process.*, vol. 9, no. 8, pp. 892–899, Nov. 2001.

[38] L. Wei, S. Weiss, and L. Hanzo, "Subband adaptive generalized sidelobe canceller for broadband beamforming," in *Proc. IEEE Signal Process. Workshop Statist. Signal Process.*, 2001, pp. 591–594.

[39] D. Marelli and M. Fu, "Asymptotic properties of subband identification," *IEEE Trans. Signal Process.*, vol. 52, no. 1, pp. 142–154, Jan. 2004.

[40] P. Chanda and S. Park, "Immersive rendering of coded audio streams using reduced rank models of subband-domain head-related transfer functions," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2006, pp. 345–348.

[41] D. Marelli, "A functional analysis approach to subband system identification and approximation," *IEEE Trans. Signal Processing*, vol. 55, no. 2, pp. 493–506, Feb. 2007.

[42] D. Marelli, M. Fu, P. Balazs, and P. Majdak, "Approximation of LTI systems in the time–frequency domain for virtual acoustics," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2008.

[43] S. Weiss and R. Stewart, "Fast implementation of oversampled modulated filter banks," *Electron. Lett.*, vol. 36, no. 17, pp. 1502–1503, Aug. 2000.

[44] J. G. Proakis and D. G. Manolakis, *Digital Signal Processing: Principles, Algorithms, and Applications*, 3rd ed. Englewood Cliffs, NJ: Prentice-Hall, 1996.

[45] P. Vaidyanathan, *Multirate Systems and Filterbanks*. Englewood Cliffs, NJ: Prentice-Hall, 1993.

[46] A. Ben-Israel and T. N. E. Greville, *Generalized Inverses*, ser. CMS Books in Mathematics/Ouvrages de Mathématiques de la SMC, 15., 2nd ed. New York: Springer-Verlag, 2003, theory and applications.

[47] D. Marelli and M. Fu, "A recursive method for the approximation of LTI systems using subband processing," *IEEE Trans. Signal Process.*, vol. 58, no. 3, pp. 1025–1034, Mar. 2010.

[48] J. O. Smith, "Physical modeling using digital waveguides," *Comput. Music J.*, vol. 16, no. 4, pp. 74–91, 1992.

[49] P. Welch, "The use of fast Fourier transform for the estimation of power spectra: A method based on time averaging over short, modified periodograms," *IEEE Trans. Audio Electroacoust.*, vol. AU-15, no. 2, pp. 70–73, Jun. 1967.

[50] D. Slepian, "Prolate spheroidal wave functions, Fourier analysis, and uncertainty. V-The discrete case," *Bell Syst. Tech. J.*, vol. 57, pp. 1371–1430, 1978.

Damián Marelli (M'09) received the B.S. degree in electronics engineering from the Universidad Nacional de Rosario, Santa Fe, Argentina, in 1995 and the Ph.D. degree in electrical engineering and the B.S. (Hons.) degree in mathematics from the University of Newcastle, Callaghan, NSW, Australia, in 2003.

In 2003, he held a research associate position at the School of Electrical Engineering and Computer Scienc, University of Newcastle. In 2004 and 2005, he held a postdoctoral research fellowship at the Laboratoire d'Analyse Topologie et Probabilités, CNRS/Université de Provence, France. Since 2006, he has been a Research Academic at the ARC Centre for Complex Dynamic Systems and Control, University of Newcastle. He held an Intra-European Marie Curie Fellowship at the Faculty of Mathematics, University of Vienna, Vienna, Austria, from 2007 to 2008. His main research interests include multirate signal processing, time–frequency analysis, system identification, and statistical signal processing.
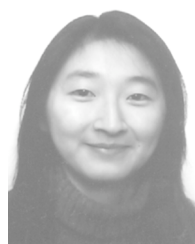
Mitsuko Aramaki (M'09) received the M.S. degree in mechanics (speciality in acoustic and dynamics of vibrations) from the University of Aix-Marseille II, Marseille, France, and the Ph.D. degree for her work at the Laboratoire de Mécanique et d'Acoustique, Marseille, France, in 2003 on analysis and synthesis of impact sounds using physical and perceptual approaches.

She is currently a Researcher at the Mediterranean Institute for Cognitive Neuroscience, Marseille, France, where she works on a pluridisciplinary project combining sound modeling, perceptual and cognitive aspects of timbre, and neuroscience methods in the context of virtual reality.

Richard Kronland-Martinet (SM'09) received the Ph.D. degree in acoustics in 1983, and the Doctorat d'Etat es Sciences degree in 1989 for his work on analysis and synthesis of sounds using time–frequency and time-scale (wavelets) representations, both from the University of Aix-Marseille II, Marseille, France.

He is currently a Director of Research at the National Center for Scientific Research (CNRS), Laboratoire de Mécanique et d'Acoustique, Marseille, where he is the Head of the group Modeling, Synthesis, and Control of Sound and Musical Signals. His primary research interests are in analysis and synthesis of sounds with a particular emphasis on high-level control of synthesis processes. He recently addressed applications linked to musical interpretation and semantic description of sounds using a pluridisciplinary approach associating signal processing, physics, perception, and cognition.

Charles Verron (M'10) received the M.S. degree in acoustics, signal processing, and computer sciences applied to music from University Paris 6, Paris, France, in 2004. He is currently pursuing the Ph.D. degree at Orange Labs, Lannion, France, and at the Laboratoire de Mécanique et d'Acoustique (LMA), Marsellie, France.

He was a Research Assistant in 2006 in the Computer and Audio Research Laboratory, Sydney University, Sydney, Australia, where he contributed to 3-D audio projects. His main research interests include sound synthesis and sound spatialization, for music and virtual reality applications.