Nonlinear Distortion Synthesis Using the Split-Sideband Method, with Applications to Adaptive Signal Processing*

VICTOR LAZZARINI, JOSEPH TIMONEY, *AES Member,* **AND THOMAS LYSAGHT**
(Victor.Lazzarini@nuim.ie) (JTimoney@cs.nuim.ie) (TLysaght@cs.nuim.ie)

Sound and Digital Music Technology Research Group, National University of Ireland, Maynooth, Ireland

A new nonlinear distortion synthesis technique, split-sideband synthesis (SpSB), is discussed. The method is closely related to the well-known principles of waveshaping, single-sideband modulation, and frequency modulation. In the technique described, however, it is possible to separate the different sideband grounds (upper, lower, even, and odd) as independent outputs. This allows for a greater flexibility of sound design for the synthesist, including the production of novel spectral transitions. The method and its implementation are outlined, drawing on the relationship between it and its correlates. The use of multiple SpSB processors is then explored, followed by a discussion of the application of the technique to adaptive effects.

0 INTRODUCTION

Beginning with the introduction of the technique of audio frequency modulation (FM) by Chowning [1], nonlinear distortion synthesis has had a rich history of development. Many similar techniques were developed alongside it, including phase modulation (PM) [2], waveshaping [3], and phase distortion [4]. Most of these techniques are correlated and work on the principle of distorting a simple waveshape in some way to generate complex spectra. In fact, most of these methods can be regarded as subsets of summation formula synthesis, described first by Windham and Steiglitz [5] and then, more comprehensively, by Moore [6], [7]. A main advantage of many of these techniques is the relative economy in computation that they represent, as they are capable of generating a variety of tones with only a fraction of the cost of additive synthesis.

Since the early 1990s relatively little new work has been carried out on the signal processing issues related to these methods. Recently new interest has been reinjected into the area with the development of adaptive FM (AdFM) synthesis [8], [9] and other similar applications of the technique [10]. The present work is in fact an extension of part of that research, as a refinement of some of the techniques developed for that work. As will be demonstrated, it is readily available for adaptive applications.

In the present paper a nonlinear distortion technique is discussed, which is related to several of the synthesis methods mentioned, but is novel in its formulation. By naming it split-sideband (SpSB) synthesis we indicate its major feature, that of independent sideband control. The SpSB method employs common distortion synthesis parameters such as modulation/carrier frequencies and the modulation (or distortion) index. It generates four independent outputs, with the resulting complex spectra being separated into different sideband groups: lower odd, lower even, upper odd, and upper even (Fig. 1). These signals can then be mixed down in a variety of combinations and at different levels to produce different spectra. They can also be further processed or spatialized.

This paper is organized as follows. We will first describe the synthesis method from the perspective of obtaining the four different output signals. In sequence we will place the technique in relation to other correlated distortion methods. In particular we will demonstrate how SpSB is a form of both dynamic waveshaping and FM/PM synthesis. We will also relate the method to other relevant previous research. A reference implementation will be presented, followed by some examples. The paper concludes with a discussion of complex (that is, multielement) and adaptive SpSB.

1 SPLIT-SIDEBAND METHOD

SpSB synthesis is based on two main principles: 1) production of independent odd and even sidebands, and 2) separation of lower and upper sideband groups. The first is realized by a combination of amplitude (ring) modulation and function mapping (similar to waveshaping) and the second is enabled by the use of complex analytic signals.

*Manuscript received 2007 December; revised 2008 July 25.

1.1 Producing Even and Odd Sidebands

If we consider the following function mapping (using $\omega = 2\pi f$, with f being the frequency of the input sine wave in hertz),

$$y(t) = \cos[I \sin(\omega t + \phi)]. \tag{1}$$

then, using the Jacobi–Anger identity [11],

$$\begin{aligned} e^{ik\sin(\omega t)} = e^{ik\cos(\omega t - \pi/2)} &= \sum_{n=-\infty}^{\infty} i^n J_n(k) \cos\left[n\left(\omega t - \frac{\pi}{2}\right)\right] \\ &= J_0(k) + 2\sum_{n=1}^{\infty} J_{2n}(k)\cos(2n\omega t) \\ &\quad + 2i\sum_{n=1}^{\infty} J_{2n-1}(k)\sin[(2n-1)\omega t] \end{aligned} \tag{2}$$

we obtain the following expansion of Eq. (1):

$$y(t) = \Re\{e^{jI\sin(\omega t+\phi)}\} = J_0(I) + 2\sum_{n=1}^{\infty} J_{2n}(I)\cos(2n\omega t + 2n\phi). \tag{3}$$

We can see that this is a relatively easy method to produce only even frequency components. Each component is scaled by $J_n(I)$, a Bessel function of the first kind of order n. (Such scaling functions will be further discussed in relation to waveshaping.) Although this expansion is theoretically infinite, the effective signal bandwidth will be dependent on the index I so that for values of n much larger than I, the amplitude of the nth component is very small.

Likewise, if instead we use

$$y(t) = \sin[I \sin(\omega t) + \phi] \tag{4}$$

we will produce the following spectrum:

$$\begin{aligned} y(t) &= \Im\{e^{jI\sin(\omega t+\phi)}\} \\ &= 2\sum_{n=1}^{\infty} J_{2n-1}(I)\sin[(2n-1)\omega t + (2n-1)\phi]. \end{aligned} \tag{5}$$

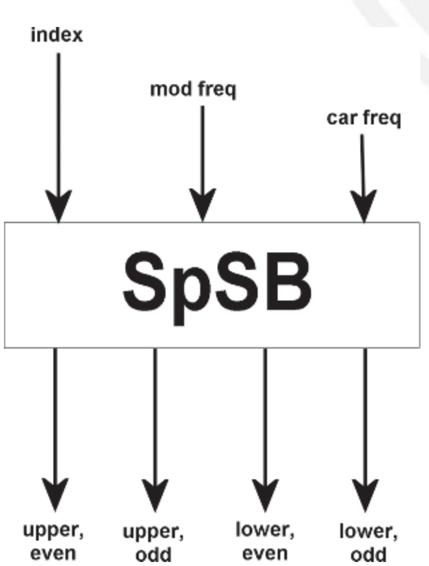

Fig. 1. SpSB synthesis inputs and outputs.

This will give us the missing odd components of Eqs. (1) and (3). Similar bandwidth considerations apply here. Both spectra will be harmonic, with the fundamental frequency in both cases being assigned to the radian frequency ω (missing in the first spectrum, but implied by relationships in the higher harmonics).

In order to produce spectra that can be cenetred at any frequency, we can then ring-modulate both signals with a sine-wave carrier at ω_c,

$$y_{\text{even}}(t) = \sin(\omega_c t)\cos[I \sin(\omega_m t)] \tag{6}$$

and

$$y_{\text{odd}}(t) = \sin(\omega_c t)\sin[I \sin(\omega_m t)]. \tag{7}$$

This allows us to obtain two sideband groups separately: the even sidebands [Eq. (6)], $\omega_c \pm 2n\omega_m$, and the odd sidebands [Eq.(7)], $\omega_c \pm (2n-1)\omega_m$. This pair of equations produces spectra that are characterized by their ω_c/ω_m ratio, placed symmetrically around ω_c. In fact, they generate, in a different way, the same output as in classic FM (or more accurately phase modulation) synthesis, except that here we have separate control over even and odd sidebands. The links to FM synthesis will be further discussed shortly.

1.2 Separating the Upper and Lower Sidebands

In order to separate the sidebands above and below the carrier frequency, we will modify the ring modulation of Eqs. (6) and (7) slightly. Instead of multiplying real signals, which exhibit Hermitian spectra (with positive and negative frequencies mirrored at 0 Hz), we will instead use analytic signals. These will contain either positive or negative frequencies only. When their spectra are convolved with complex sinusoids (by time-domain multiplication), we will be able to produce only upper or lower sidebands. In order to produce the required analytic signals we will need to generate quadrature signals, which is quite simple for sinusoids. In that case our complex sine-wave carrier $x_{\text{car}}(t)$ will be defined as

$$x_{\text{car}}(t) = \sin(\omega_c t) - j\cos(\omega_c t). \tag{8}$$

This can be implemented simply with a pair of oscillators producing sine and cosine phase sinusoids. However, it is not possible to produce a synthetic quadrature signal for the cosine or sine–mapped signals of Eqs. (1) and (4) using similar principles because of phase offset peculiarities of their expansions [see Eq. (2)]. To solve this we will instead apply a Hilbert transform [12] to produce the correct phase delays needed. We can define the Hilbert transform $H\{x\}$ of an arbitrary signal as

$$H\{s(t)\} = s(t) * \frac{1}{\pi t} = \frac{1}{\pi}\int_{-\infty}^{\infty} \frac{s(\tau)}{t-\tau}\, d\tau. \tag{9}$$

This results in a constant phase shift of $\pi/2$ across the positive spectrum, thus generating the required quadrature signal. Consequently we can produce signals exhibiting only positive frequencies,

$$z_{\mathrm{pos}}(t) = s(t) + \mathrm{j}H\{s(t)\} \tag{10}$$

or, conversely, we can choose to preserve only the negative spectrum,

$$z_{\mathrm{neg}}(t) = s(t) - \mathrm{j}H\{s(t)\}. \tag{11}$$

A Hilbert transform filter (Fig. 2)[13], which implements it, can then be used to take in an arbitrary real signal $s(t)$ and decompose it into a complex form, as in

$$\begin{aligned} s(t) = z_{\mathrm{pos}}(t) + z_{\mathrm{neg}}(t) = \Re(z_{\mathrm{pos}}(t)\} + \Im\{z_{\mathrm{pos}}(t)\} + \Re(z_{\mathrm{neg}}(t)\} \\ - \Im(z_{\mathrm{neg}}(t)\}. \end{aligned} \tag{12}$$

With that in hand we can then realize single-sideband modulation [14] instead of the ring modulation shown in Eqs. (6) and (7). Using the signals defined in Eqs. (8) and (12) the upper and lower sidebands are then produced by the following mix of heterodyne signals,

$$\begin{aligned} s_{\mathrm{upper,even}}(t) &= \cos[I \sin(\omega_{\mathrm{m}}t)] \sin(\omega_{\mathrm{c}}t) \\ &\quad + H\{\cos[I \sin(\omega_{\mathrm{m}}t)]\} \cos(\omega_{\mathrm{c}}t) \\ &= J_0(I) \sin(\omega_{\mathrm{c}}t) + 2 \sum_{n=1}^{\infty} J_{2n}(I) \sin(\omega_{\mathrm{c}} + 2n\omega_{\mathrm{m}}t) \end{aligned} \tag{13}$$

$$\begin{aligned} s_{\mathrm{upper,odd}}(t) &= \sin[I \sin(\omega_{\mathrm{m}}t)] \sin(\omega_{\mathrm{c}}t) \\ &\quad + H\{\sin[I \sin(\omega_{\mathrm{m}}t)]\} \cos(\omega_{\mathrm{c}}t) \\ &= 2 \sum_{n=1}^{\infty} J_{2n-1}(I) \cos[\omega_{\mathrm{c}} + (2n-1)\omega_{\mathrm{m}}t] \end{aligned} \tag{14}$$

$$\begin{aligned} s_{\mathrm{lower,even}}(t) &= \cos[I \sin(\omega_{\mathrm{m}}t)] \sin(\omega_{\mathrm{c}}t) \\ &\quad - H\{\cos[I \sin(\omega_{\mathrm{m}}t)]\} \cos(\omega_{\mathrm{c}}t) \\ &= J_0(I) \sin(\omega_{\mathrm{c}}t) + 2 \sum_{n=1}^{\infty} J_{2n}(I) \sin(\omega_{\mathrm{c}} - 2n\omega_{\mathrm{m}}t) \end{aligned} \tag{15}$$

$$\begin{aligned} s_{\mathrm{lower,odd}}(t) &= \sin[I \sin(\omega_{\mathrm{m}}t)] \sin(\omega_{\mathrm{c}}t) \\ &\quad - H\{\sin[I \sin(\omega_{\mathrm{m}}t)]\} \cos(\omega_{\mathrm{c}}t) \\ &= 2 \sum_{n=1}^{\infty} J_{2n-1}(I) \cos[\omega_{\mathrm{c}} - (2n-1)\omega_{\mathrm{m}}t]. \end{aligned} \tag{16}$$

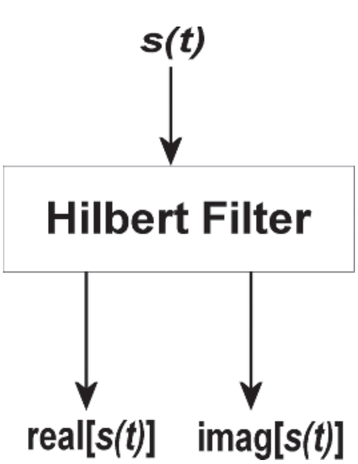

Fig. 2. Hilbert transform filter.

A scaling constant of 0.5 is normally added to each output so that the total signal amplitude is the same as when the four sideband groups are generated together.

2 RELATIONSHIP TO OTHER TECHNIQUES

SpSB synthesis sits in a crossroad of several distortion techniques. It is in essence an amplitude modulation method using single-sideband principles, which also includes nonlinear waveshaping of sinusoidal inputs. In addition it is a reformulation of FM (PM) synthesis, producing an exact spectral match of an FM signal if the four signals are mixed together. This section discusses the relationship of SpSB to these three techniques.

2.1 Single-Sideband Modulation Principles

The technique of SSB modulation is simply an extension of the ring-modulation principle, and its formulation is a matter of applying trigonometric identities to obtain the correct formulae. The upper sideband is the real part of the product of a complex sinusoid and a complex signal, both of which contain only positive frequencies. Using two sinusoids as an example, we have

$$\begin{aligned} s_{\mathrm{ssb,upper}}(t) &= \Re\{\mathrm{e}^{\mathrm{j}\omega_{\mathrm{c}}t} \times \mathrm{e}^{\mathrm{j}\omega_{\mathrm{m}}t}\} \\ &= [\cos(\omega_{\mathrm{c}}t) + \mathrm{j}\sin(\omega_{\mathrm{c}}t)][\cos(\omega_{\mathrm{m}}t) + \mathrm{j}\sin(\omega_{\mathrm{m}}t)] \\ &\quad + [\cos(\omega_{\mathrm{c}}t) - \mathrm{j}\sin(\omega_{\mathrm{c}}t)][\cos(\omega_{\mathrm{m}}t) - \mathrm{j}\sin(\omega_{\mathrm{m}}t)] \\ &= \cos(\omega_{\mathrm{c}}t)\cos(\omega_{\mathrm{m}}t) - \sin(\omega_{\mathrm{c}}t)\sin(\omega_{\mathrm{m}}t) \\ &= \cos[(\omega_{\mathrm{c}} + \omega_{\mathrm{m}})t]. \end{aligned} \tag{17}$$

Conversely, the negative sideband (difference frequency) can be obtained by multiplying the same complex sinusoid by a signal with only negative frequencies and taking its real part,

$$\begin{aligned} s_{\mathrm{ssb,lower}}(t) &= \Re\{\mathrm{e}^{\mathrm{j}\omega_{\mathrm{c}}t} \times \mathrm{e}^{-\mathrm{j}\omega_{\mathrm{m}}t}\} \\ &= [\cos(\omega_{\mathrm{c}}t) + \mathrm{j}\sin(\omega_{\mathrm{c}}t)][\cos(\omega_{\mathrm{m}}t) - \mathrm{j}\sin(\omega_{\mathrm{m}}t)] \\ &\quad + [\cos(\omega_{\mathrm{c}}t) - \mathrm{j}\sin(\omega_{\mathrm{c}}t)][\cos(\omega_{\mathrm{m}}t) + \mathrm{j}\sin(\omega_{\mathrm{m}}t)] \\ &= \cos(\omega_{\mathrm{c}}t)\cos(\omega_{\mathrm{m}}t) + \sin(\omega_{\mathrm{c}}t)\sin(\omega_{\mathrm{m}}t) \\ &= \cos[(\omega_{\mathrm{c}} + \omega_{\mathrm{m}})t]. \end{aligned} \tag{18}$$

For nonsinusoidal modulators, SSB is normally implemented in similar fashion to SpSB, by using a Hilbert transform filter.

2.2 SpSB and Waveshaping

As mentioned before, SpSB synthesis is based on the principle of nonlinear waveshaping, where a signal is mapped according to a certain transfer function. In this case, if we want to identify the polynomial that makes up the transfer functions used (cosines and sines), we begin by using Taylor series expansions,

$$\cos(x) = 1 = \frac{x^2}{2!} + \frac{x^4}{4!} - \frac{x^6}{6!} + \cdots = \sum_{n=0}^{\infty} (-1)^n \frac{x^{2n}}{2n!} \tag{19}$$

$$\sin(x) = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} + \cdots = \sum_{n=0}^{\infty} (-1)^n \frac{x^{2n-1}}{(2n+1)!}. \tag{20}$$

From this perspective it is possible to use the theory of waveshaping to arrive at the same spectra defined by the expansion of Eqs. (1) and (4). Based on Eqs. (19) and (20), the expansion of the sum of these equations is

$$\cos[I\sin(\omega t)] + \sin[I\sin(\omega t)] = \sum_{n=0}^{\infty}(-1)^n\frac{[I\sin(\omega t)]^{2n}}{2n!} + (-1)^n\frac{[I\sin(\omega t)]^{2n+1}}{(2n+1)!}. \tag{21}$$

This polynomial expansion touches two well-known facts of waveshaping: 1) even polynomials only produce even components (a similar point can be made for odd polynomials); 2) alternating − and + signs for even and odd terms produces better balanced spectral evolutions as the value of I changes. We can also show how the scaling functions $J_n(I)$ in Eqs. (3) and (5) are related to the polynomial expansion in Eq. (19),

$$\begin{aligned} &J_0(I) + 2\sum_{n=1}^{\infty}J_{2n}(I)\cos(2n\omega t) + J_{2n-1}(I)\sin[(2n-1)\omega t] \\ &= \sum_{k=0}^{\infty}(-1)^k\frac{(I/2)^{2k}}{(k!)^2} + 2\sum_{n=1}^{\infty}\sum_{k=1}^{\infty}(-1)^k\frac{(I/2)^{n+2k}}{k!(n+k)!}\cos(n\omega t) \\ &= 1 + I\sin(\omega t) - \frac{I^2\sin^2(\omega t)}{2} - \frac{I^3\sin^3(\omega t)}{3!} + \cdots \end{aligned} \tag{22}$$

which uses the definition of $J_n(I)$ for integral values of n [15],

$$J_n(I) = \sum_{k=0}^{\infty}(-1)^k\frac{(I/2)^{n+2k}}{k!(n+k)!}. \tag{23}$$

In fact, this is yet another way of deriving Bessel functions of the first kind. It is certainly possible to demonstrate it by using Taylor expansion in Eq. (21) up to order 4 and collecting all the terms associated with the different cosine wave frequencies,

$$\begin{aligned} &1 + I\sin(\omega t) - \frac{I^2\sin^2(\omega t)}{2} - \frac{I^3\sin^3(\omega t)}{3!} + \frac{I^4\sin^4(\omega t)}{4!} \\ &= 1 - \frac{I^2}{4} + \frac{I^4}{64} + 2\left(\frac{I}{2} - \frac{I^3}{16}\right)\sin(\omega t) + 2\left(\frac{I^2}{8} - \frac{I^4}{64}\right)\cos(2\omega t). \end{aligned} \tag{24}$$

The larger the value of I, the more polynomial terms will be needed to define the spectral expansion of these formulas adequately. A five-term expansion will feature three components only and will only be precise enough for values of I below 1. This is equivalent to the relationship between I and the effective signal bandwidth of SpSB synthesis. This fact also makes SpSB a dynamic form of waveshaping as the effective order of the polynomial transfer function will vary with I, adding extra components to the output spectrum.

2.3 SpSB as a Form of FM Synthesis

If we take the non-single-sideband form of SpSB demonstrated in Eqs. (6) and (7), it is easy to demonstrate that this pair of equations is another form of the usual FM (more exactly PM) synthesis. By using a trigonometric identity, and the expansion shown in Eqs (3) and (5), we have

$$\begin{aligned} y(t) &= \sin(\omega_c t)\cos[I\sin(\omega_m t)] + \sin(\omega_c t)]\sin[I\sin(\omega_m t)] \\ &= 0.5\{\sin[\omega_c t + I\sin(\omega_m t)] + \sin[\omega_c t = I\sin(\omega_m t)] \\ &\quad - \cos[\omega_c t + I\sin(\omega_m t)] + \cos[\omega_c t - I\sin(\omega_m t)]\} \\ &= \sum_{n=-\infty}^{\infty}J_{2n}(I)\sin(\omega_c t + 2n\omega_m t) \\ &\quad + \sum_{n=-\infty}^{\infty}J_{2n-1}(I)\cos[\omega_c t + (2n-1)\omega_m t]. \end{aligned} \tag{25}$$

This is almost the original FM formulation by Chowning, except that the even and odd sidebands have different phase offsets. If we employed a cosine wave as a carrier to generate the odd sidebands, we would have had an exact match to sine-wave carrier and modulator FM. In its full form SpSB is therefore effectively single-sideband FM synthesis.

2.4 Related Work

The majority of the research on distortion techniques took place in the 1970s and 1980s. Particularly relevant to the current research was the development of variations on the FM technique to provide a more flexible sideband control. The two major works in this area involved some sort of ring modulation of an FM signal, using exponentiated signals. Moorer[7] proposed the following variant, which allowed for some clever modification of sideband amplitudes:

$$\begin{aligned} s(t) &= e^{I\cos(\omega_m t)}\sin[\omega_c t + I\sin(\omega_m t)] \\ &= \sum_{n=0}^{\infty}\frac{I^n}{n!}\sin(\omega_c t + n\omega_m t). \end{aligned} \tag{26}$$

It is especially interesting how this formula is able to produce single-sideband spectra. Notice that while based on the original FM principles, we have a completely different set of sideband amplitudes, which are no longer based on Bessel functions. This method was further developed by Palamin et al. [16], who proposed a more flexible way of developing asymmetric spectra,

$$\begin{aligned} s(t) &= e^{[1/2(r-1/r)\cos(\omega_m t)]}\sin\left[\omega_c t + \frac{I}{2}\left(r + \frac{1}{r}\right)\sin(\omega_m t)\right] \\ &= \sum_{n=-\infty}^{\infty}r^n J_n(I)\sin(\omega_c t + n\omega_m t). \end{aligned} \tag{27}$$

In their approach we have a single parameter r, which allows for modification of the sideband amplitudes, $r < 1$ reinforcing lower order partials and $r > 1$ resulting in the reverse effect. Both methods, while giving some new means of changing sideband amplitudes, do not, however, allow for independent separation and control of sideband groups. The technique proposed here provides such independent control, which can create a full range of results, from classic FM-like to various combinations of gapped and single-sideband spectra.

3 IMPLEMENTATION

SpSB can be implemented in a variety of software synthesis systems. The only required component that is not always found in all systems is the Hilbert transform filter. This, however, can be implemented easily as a pair of sixth-order all-pass filters, producing the following analytic output:

$$g(n) = y(n) + \mathrm{j}\omega(n-1). \tag{28}$$

The two all-pass filters will maintain close to a 90-degree phase separation over most of the useful frequency range. These filters, realized in cascading first-order sections, have the following equations [17]:

$$\omega_m(n) = C_m[x_m(n) - \omega_m(n-1)] + x_m(n-1), \quad m = 1, 2, \ldots, 6 \tag{29}$$

with C_m = {0.3609, 52.7412, 11.1573, 44.7581, 179.622, 708.4578}, and

$$y_m(n) = C_m[x_m(n) - y_m(n-1)] + x_m(n-1), \quad m = 1, 2, \ldots, 6 \tag{30}$$

with C_m = {1.2524, 5.5761, 22.3422, 89.6271, 365.7914, 2771.1114}.

A reference implementation in Csound [18] is presented in Fig. 3. It uses the Hilbert opcode, which implements the pair of filters given in the precedings. In versions of Csound 5 prior to 5.08, this opcode had its outputs reversed, with the imaginary (sine) first, followed by the real (cosine) output. The code presented here is written for the fixed form of the opcode (real, imaginary).

```
/* SpSB opcode
a1,a2,a3,a4 SpSB kamp, kfc, kfm, kndx, \
ifn /
a1,a2,a3,a4—upper/even, upper/odd, \
lower/even, lower/odd outputs /
kamp—amplitude
kfc—carrier frequency
kfm—modulator frequency
kndx—index of modulation
```

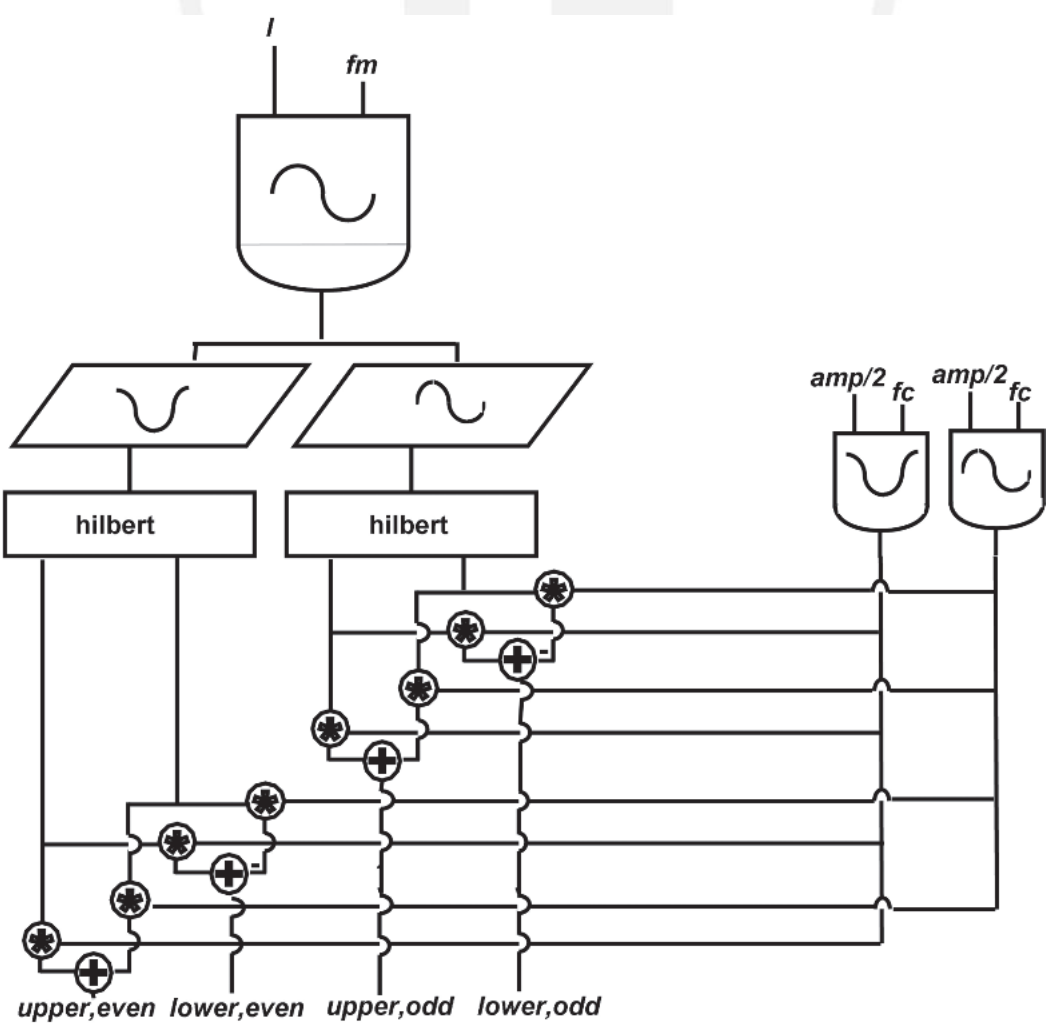

Fig. 3. SpSB synthesis.

```
ifn—sinewave function table number
*/

opcode SpSB, aaaa,kkkki
ka,kc,km,kndx,ifn xin

a1 oscili kndx/(2*$M_PI),km,ifn ; sine \
modulator /
a2 tablei a1,ifn,1,0.25,1       ; co- \
sine function /
a3 tablei a1,ifn,1,0,1          ; sine \
function /

; complex modulators
aae, abe hilbert a2 ; cos (sin()) : even \
sidebands, analytic /
aao, abo hilbert a3 ; sin(sin()) : odd \
sidebands, analytic /
; complex sine carrier: 0.5 (sin(ifc) - \
jcos(ifc)) /
ac oscili ka/2,kc,ifn
ad oscili ka/2,kc,ifn,0.25

; even and odd sidebands, lower/upper \
sides /
aeu = aae*ac + abe*ad
aou = aao*ac + abo*ad
ael = aae*ac – abe*ad
aol = aao*ac – abo*ad
 xout aeu,aou,ael,aol
endop
```

4 EXAMPLES

The spectral plots of the four different SpSB outputs are shown in Figs. 4–7. These were produced with $f_c = 5000$, $f_m = 200$, and $I = 5$. Fig. 4 shows the resulting upper even sidebands ($f_c + 2nf_m$)—5000, 5400, 5800, . . .—whereas in Fig. 5 we see the complementary upper odd sidebands—5200, 5600, 6000, . . . The lower sidebands, all frequencies below the carrier, are shown in the next two figures. Even sidebands—5000, 4600, 4200, 3800, 3400, and 3000—are seen in Fig. 6. Their complement—4800, 4400, 4000, 3600, and 3200—are shown in Fig. 7.

As with FM synthesis (and waveshaping), by varying the index of modulation, more significant components will be produced, generating dynamic spectra. Different ratios of carrier to modulator frequency (known as the *c*/*m* ratio) will determine important characteristics of the output spectra, such as their level of harmonicity. In addition, different mixes of the four outputs can also be used to provide a variety of timbral possibilities. For instance, with $c/m = 1$ and using mostly upper even sidebands, plus a small amount of upper odd sidebands, we will be able to generate a clarinetlike timbre, where odd harmonics prevail ($f_c, f_c + 2f_m, f_c + 4f_m, \ldots$). By mixing more of the other outputs, this can be seamlessly transformed into a tone with completely different timbral characteristics. A large variety of such dynamic spectra is possible, rendering SpSB a prime candidate for spectral matching applications [19].

5 COMPLEX SpSB

The SpSB synthesis technique has been presented here in its simplest form. It is possible to develop it further with a number of variations. By analogy to complex FM [20], for instance, we can use two SpSB processors in a stacked arrangement, whereby one of the outputs from the first processor is fed into the modulation input of the second,

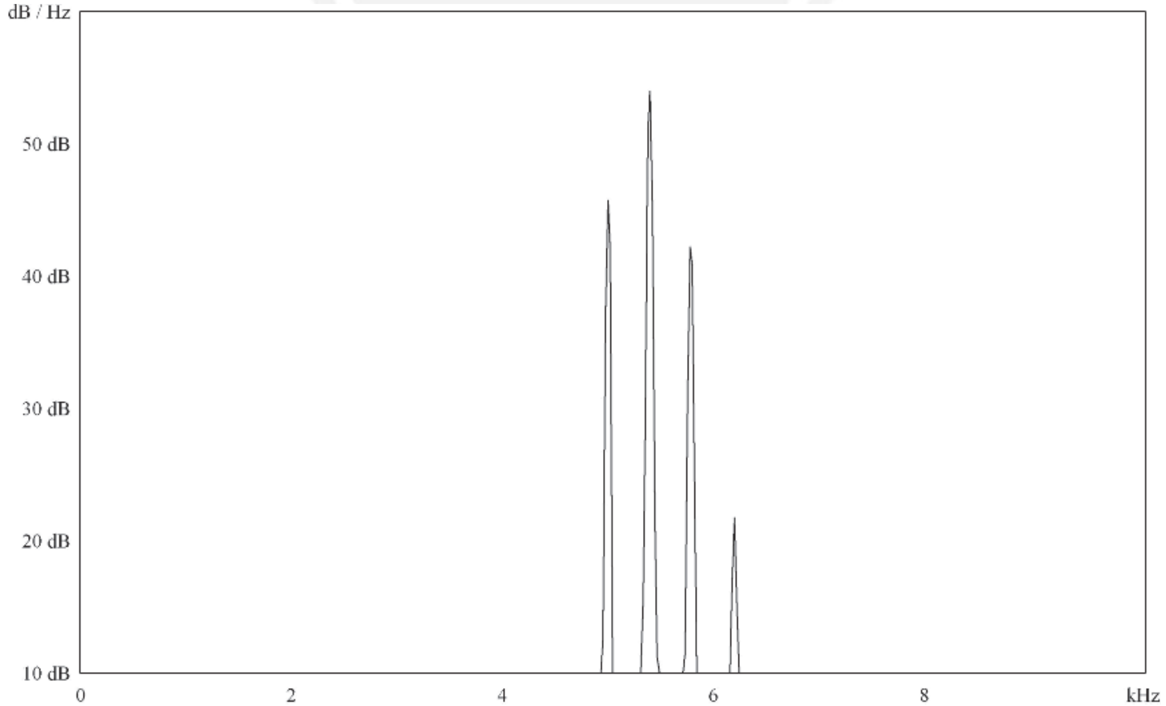

Fig. 4. Upper even SpSB sidebands.

generating different combinations of sidebands. One very interesting result is obtained by the following structure: SpSB1 upper even sideband output feeding into SpSB2 modulator frequency input (Fig. 8). In this case the lowest sideband will be the carrier frequency of the second processor.

With this design we will obtain a spectrum with a number of formant regions (Fig.9), The spacing of components around these regions will be controlled by f_m and the first formant region center frequency by a certain choice of f_{c1},

$$f_{c1} = \frac{f_a - f_{c2} - 2f_m}{2} . \tag{31}$$

The other formant regions will be centered at integer multiples of the first formant frequency. The f_a frequency is

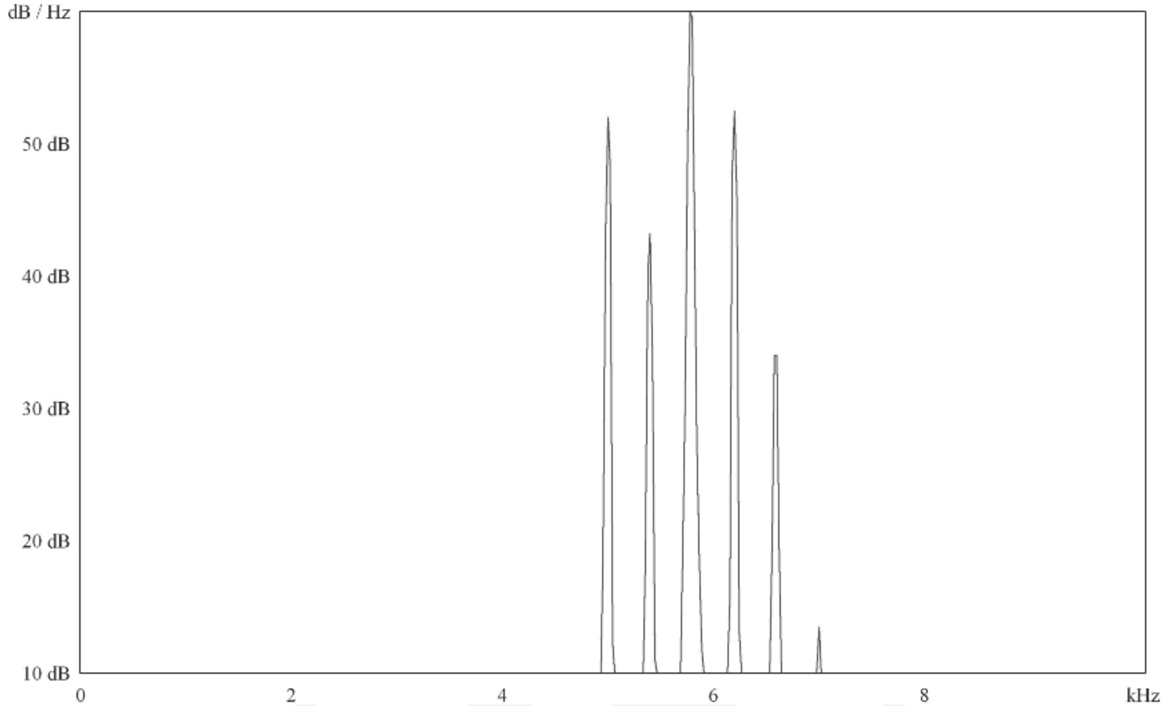

Fig. 5. Upper odd SpSB sidebands.

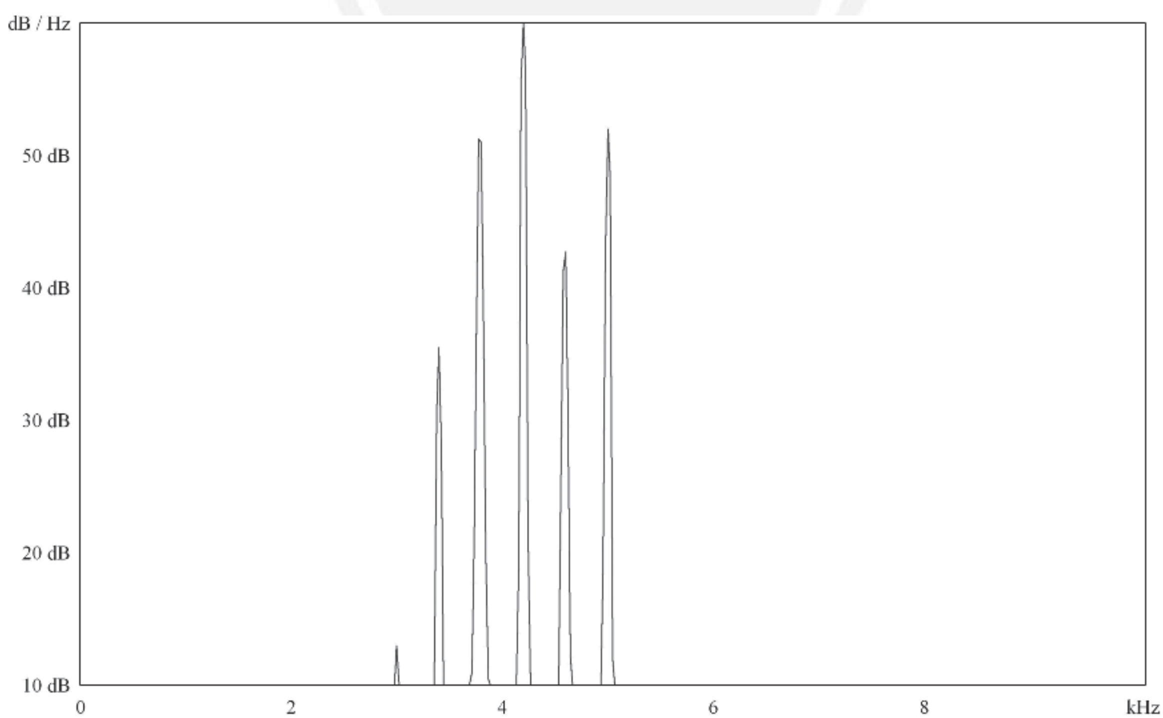

Fig. 6. Lower even SpSB sidebands.

made an integer multiple of the fundamental f_0, the closest to a target formant frequency f_f,

$$f_a = \mathrm{int}\left(\frac{f_f}{f_0} + 0.5\right) f_0. \qquad (32)$$

The lowest component will always be f_{c2}, but the perceived fundamental frequency will depend on the ratio $f_{c2}/2f_m$.

6 ADAPTIVE SpSB SYNTHESIS

Adaptive effects have been described as 1) generally, as based on controllable time-varying parameters, and 2) more specifically [21], as using extracted features of input audio to drive such controls (self-adaptive). As all the SpSB parameters can be time-varying, the technique is ready for adaptive processing applications according to definition 1). Moreover, the SpSB design is easily modified to allow for self-adaptive applications, in similar fashion to aforementioned AdFM. All that is necessary is to substitute the sinusoidal carrier by an input signal, with a pitch tracker so that the correct carrier modulator ratio can be imposed. A reference implementation for an adaptive SpSB (ASpSB) processor is as follows:

```
/*ASpSB opcode
a1,a2,a3,a4 ASpSB asig, krat, kndx, ifn

a1,a2,a3,a4—upper/even, upper/odd, \
lower/even, lower/odd outputs /
asig—input
krat—c:m ratio
kndx—index of modulation
ifn—sinewave function table number
*/
opcode ASpSB,aaaa,akki

asig,krat,kndx,ifn xin

kfc,kac ptrack asig, 512 ; pitch tracking
a1 oscili kndx/ (2*$M_PI), kfc/krat,ifn \
; sine modulator /
```

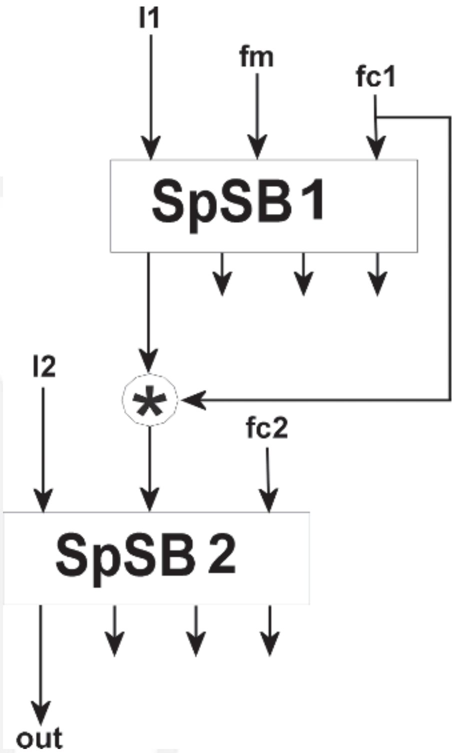

Fig. 8. Complex modulation SpSB.

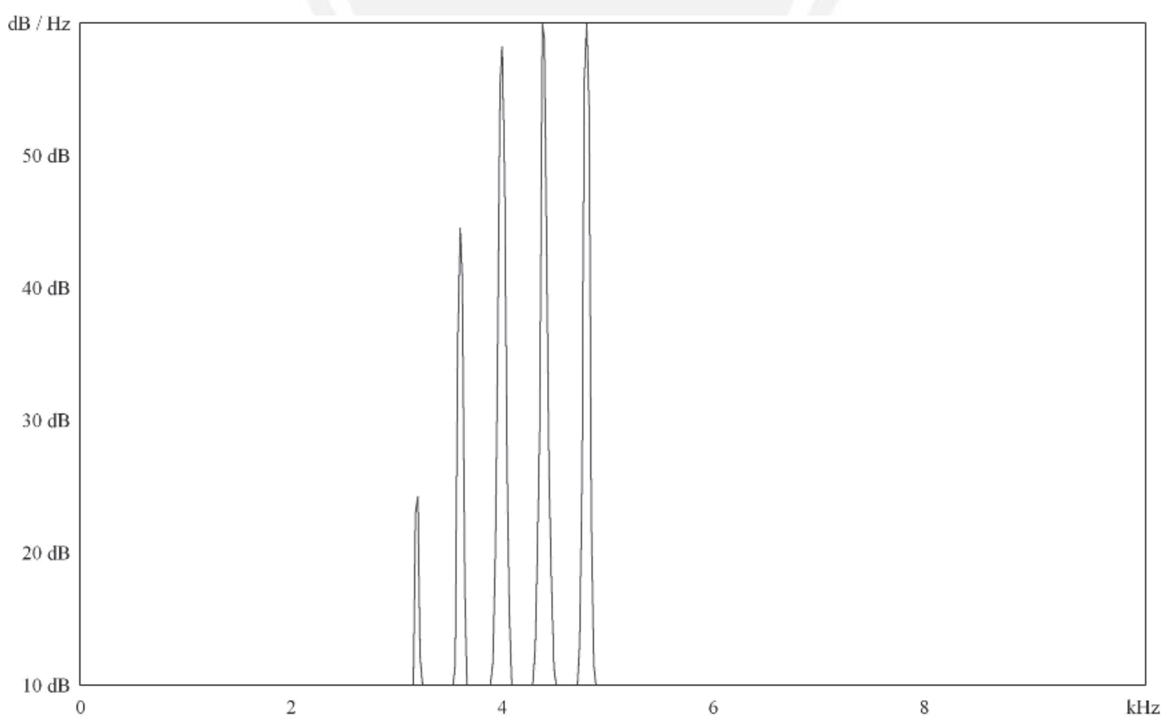

Fig. 7. Lower odd SpSB sidebands.

```
a2 tablei a1,ifn,1,0.25,1 ; cosine func- \
tion /
a3 tablei a1,ifn,1,0,1   ; sine func- \
tion /

; complex modulators
abe, aae hilbert a2 ; cos(sin())  : even \
sidebands, analytic /
abo, aao hilbert a3 ; sin(sin()) : odd \
sidebands, analytic /

; complex carrier
ad,ac hilbert asig
; even and odd sidebands, lower/upper \
sides /
aeu = aae*ac – abe*ad
aou = aao*ac – abo*ad
ael = aae*ac + abe*ad
aol = aao*ac + abo*ad
 xout aeu,aou,ael,aol
endop
```

The resulting spectra of ASpSB instruments are very similar to those of AdFM, as this technique is actually a refinement of the heterodyne adaptive FM design. This, in turn, has been demonstrated to be effectively a multicarrier process. Here, as it is possible to choose which one of the sideband groups we will use, we have a wider variety of effects possible.

Fig. 10 and 11 demonstrate an interesting example using a trumpet tone as input, where ASpSB is used to add some nearby components to each harmonic, creating a "growling" sound. For this example we have set a *c*/*m* ratio of 1:0.1, an index of modulation of 2, and we used only the upper sideband outputs (both even and odd). Two or three extra components are added to the upper side of each harmonic, generating the beating effects that characterize the growl.

Another interesting example is the use of the lower sidebands to produce subharmonics, generating a change of timbre and pitch. This effect is shown in the spectrogram on Fig. 12, which shows an oboe G4 tone being slowly morphed into a lower pitch sound by increasing the index of modulation, with *c*/*m* set to 1.5. This example uses only the lower sideband outputs.

The full range of heterodyne-based AdFM effects is possible here, with the extra possibilities offered by separate sideband outputs. Other interesting results might be obtained by placing the different sideband groups in separate spatial positions with multichannel audio. In addition, further processing might be added individually to the different outputs.

7 CONCLUSION

In this paper a method based on a nonlinear distortion process, SpSB synthesis, has been discussed. The technique is capable of generating a variety of timbres, producing independent even/odd and lower/upper sideband groups. As with other similar methods, it employs the typical parameters of modulator and carrier frequencies as well as the modulation index. In addition to the use of a single SpSB processor, it is also possible to create other arrangements of multiple such units. Considering that each unit provides four different outputs, by combining them we can create a variety of instrument designs. Finally the method is easily modified to provide support for adaptive techniques. In a similar vein to AdFM, we can use arbitrary input signals to produce a wide range of effects for a variety of musical applications. A number of sound examples and Csound code are available in the SpSB resource page [22].

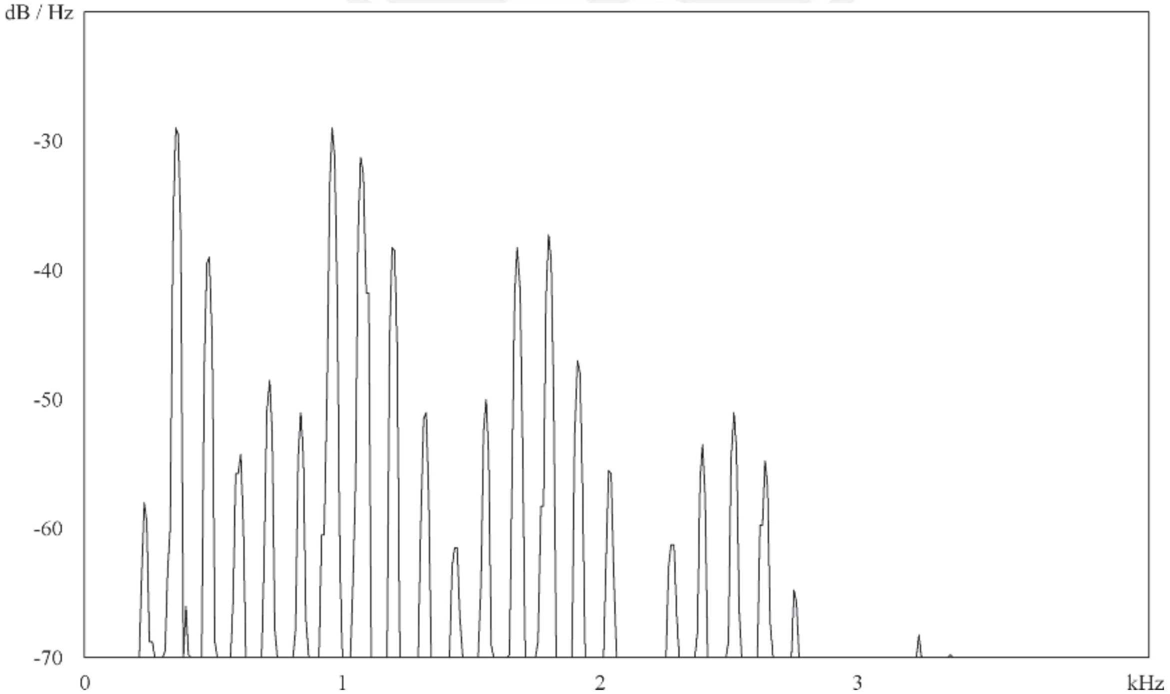

Fig. 9. Complex modulated SpSB output. Other formant regions will be centered at integer multiples of first formant.

8 REFERENCES

[1] J. Chowning, "The Synthesis of Complex Audio Spectra by Means of Frequency Modulation," *J. Audio Eng. Soc.*, vol. 21, pp. 526–534 (1973 Sept.).

[2] S. Cavaliere, G. Evangelista, and A. Piccialli, "Synthesis by Phase Modulation and Its Implementation in Hardware," *Computer Music J.*, vol. 12, no. 1, pp. 29–42 (1988).

[3] M. Le Brun, "Digital Waveshaping Synthesis," *J.*

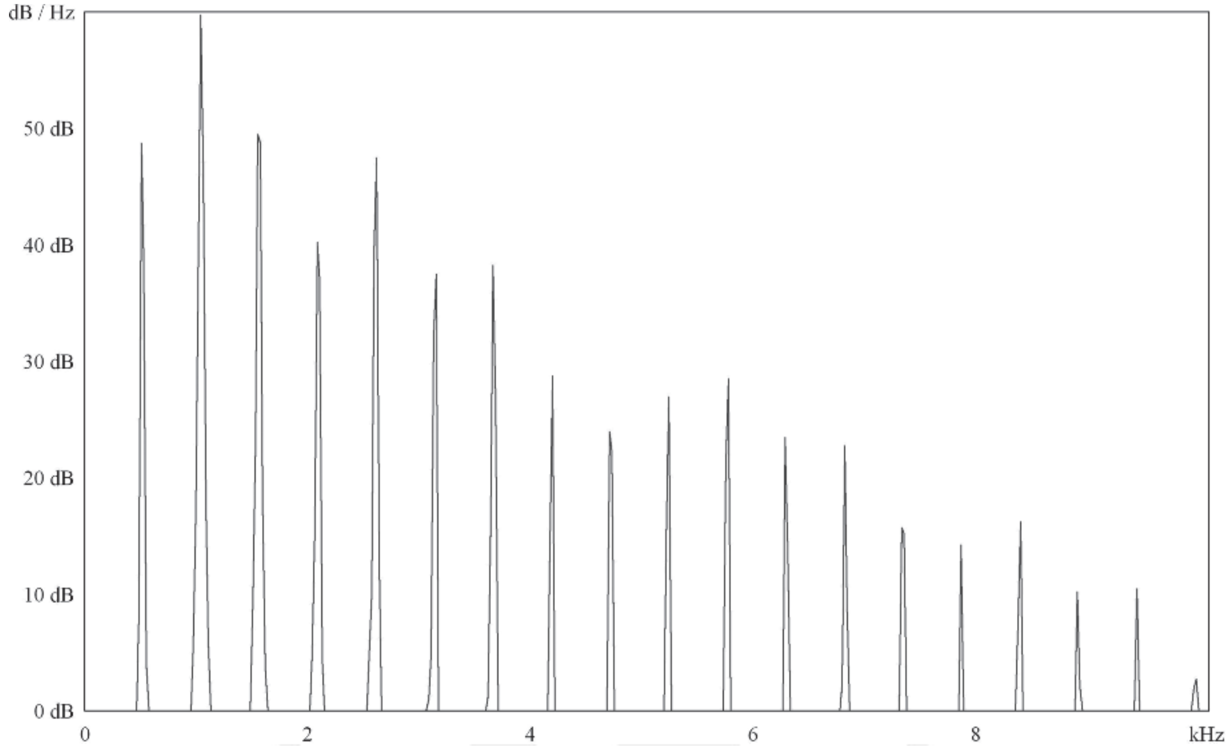

Fig. 10. Trumpet C4 tone, steady-state spectrum.

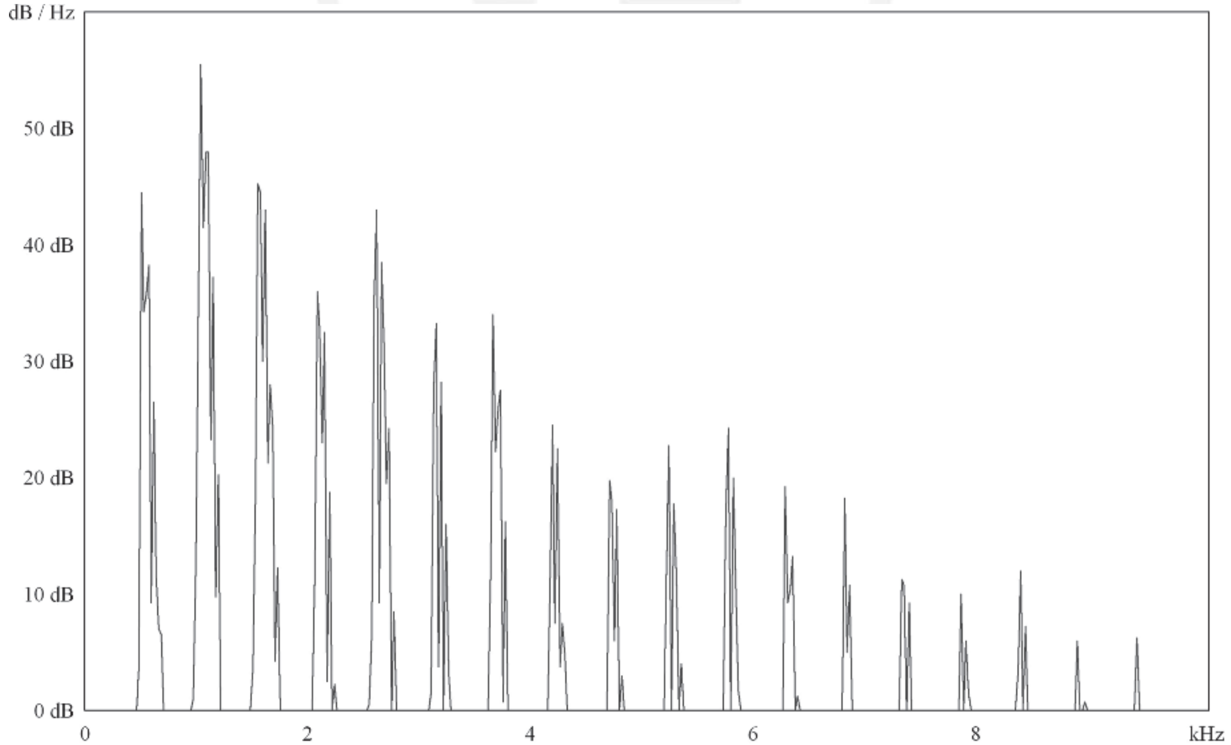

Fig. 11. Trumpet C4 tone, steady-state spectrum, processed through ASpSB, using upper sideband outputs only, with c/m = 1:0.1 and I = 2, generating a growl sound.

Audio Eng. Soc., vol. 27, pp. 250–266 (1979 Apr.).

[4] M. Ishibashi, U.S. Patent 4,658,691 (1987).

[5] G. Windham and K. Steiglitz, "Input Generators for Digital Sound Synthesis," *J. Acoust. Soc. Am.,* vol. 47, pp. 665–666 (1970).

[6] J. A. Moorer, "The Synthesis of Complex Audio Spectra by Means of Discrete Summation Formulas," *J. Audio Eng. Soc.,* vol. 24, pp. 717–727 (1976 Nov.).

[7] J. A. Moorer, "Signal Processing Aspects of Computer Music: A Survey," *Proc. IEEE,* vol. 65, pp. 1108–1141 (1977 Aug.).

[8] V. Lazzarini, J. Timoney, and T. Lysaght, "Adaptive FM Synthesis," in *Proc. 10th Int. Conf. on Digital Audio Effects* (Bordeaux, France, 2007), pp. 21–26.

[9] V. Lazzarini, J. Timoney, and T. Lysaght, "The Generation of Natural–Synthetic Spectra by Means of Adaptive Frequency Modulation," *Computer Music J.,* vol. 32, no. 2, pp. 9–22 (2008).

[10] G. Poepel and R. Dannenberg, "Audio Signal Driven Sound Synthesis," in *Proc. Int. Computer Music Conf.* (Barcelona, Spain, 2005), pp. 391–394.

[11] G. N. Watson, *A Treatise on the Theory of Bessel Functions,* 2nd ed. (Cambridge University Press, Cambridge, MA, 1944), pp. 22–23.

[12] V. Cizek, "Discrete Hilbert Transform," *IEEE Trans. Audio Electroacoust.,* vol. 18, pp. 340–343, (1970).

[13] P. Regalia, "Special Filter Designs," in *Handbook for Digital Signal Processing,* S. K. Mitra, and J. F. Kaiser, Eds. (Prentice-Hall, Englewood Cliffs, NJ, 1993), pp. 907–980.

[14] S. Wardle, "A Hilbert Transform Frequency Shifter for Audio," in *Proc. Int. Conf. on Digital Audio Effects* (Barcelona, Spain, 1999), pp. 25–29.

[15] F. W. Olver, "Bessel Functions of Integer Order," in *Handbook of Mathematical Functions,* M. Abramowitz and I. A. Stegun, Eds. (Dover, New York, 1965), pp. 355–389.

[16] J. P. Palamin, P. Palamin, and A. Ronveaux, "A Method of Generating and Controlling Musical Asymmetrical Spectra," *J. Audio Eng. Soc.,* vol. 36, pp. 671–685 (1988 Sept.).

[17] B. Hutchins, *The Musical Engineer's Handbook* (Electronotes, Ithaca, NY, 1975).

[18] J. Fitch, "On the Design of Csound 5," in *Proc. 3rd Linux Audio Conf.* (ZKM, Karlsruhe, Germany, 2005), pp. 37–42.

[19] A. Horner, J. Beauchamp, and L. Haken, "Wavetable and FM Matching Synthesis of Musical Instrument Tones," in *Proc. Int. Computer Music Conf.* (San Jose, CA, 1992), pp. 18–21.

[20] B. Schottstaedt, "The Simulation of Natural Instrument Tones Using Frequency Modulation with a Complex Modulating Wave," *Computer Music J.,* vol. 1, no. 4, pp. 46–50 (1977).

[21] V. Verfaille, U. Zolzer, and D. Arfib, "Adaptive Digital Audio Effects (a-DAFx): A New Class of Sound Transformations," *IEEE Trans. Speech Audio Process.,* vol. 14, pp. 1817–1831 (2006).

[22] http://music.nuim.ie/vlazzarini/spsb/.

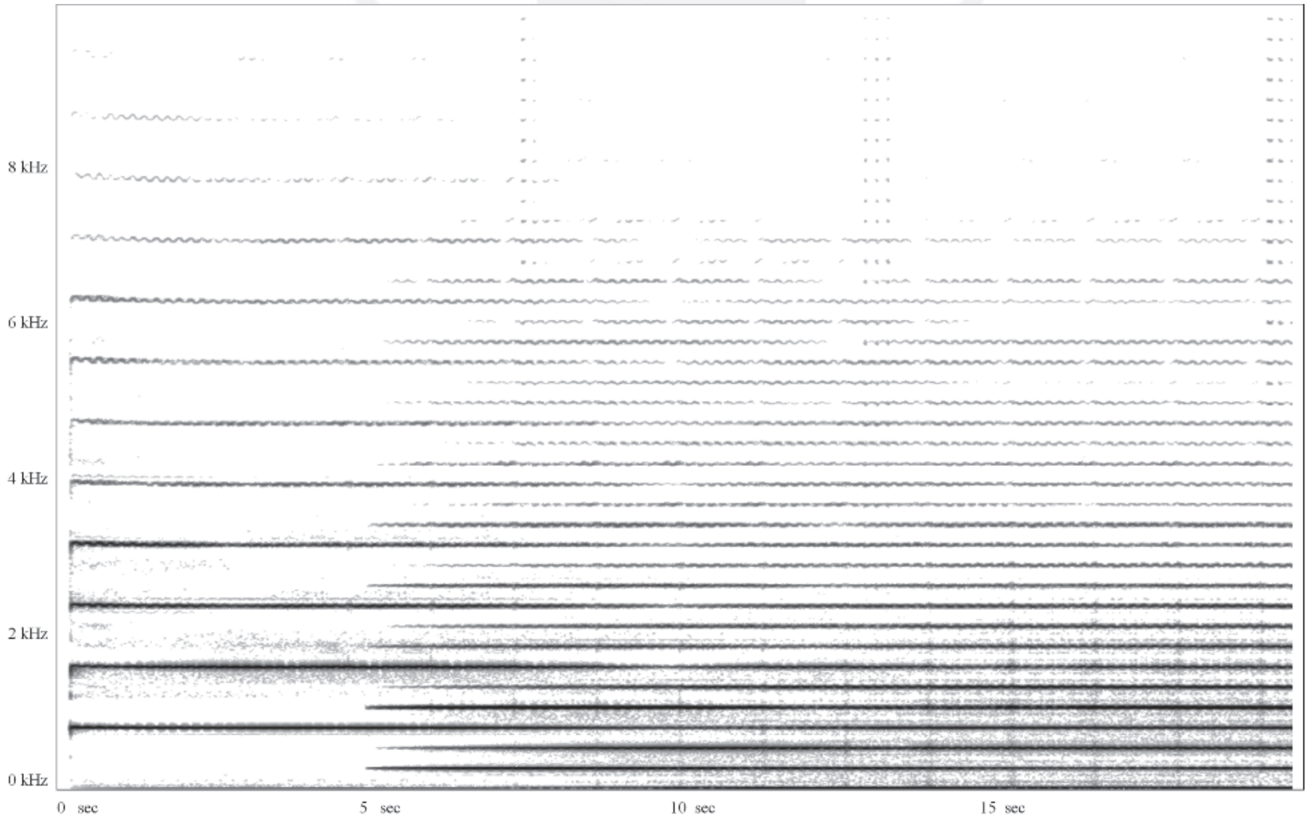

Fig. 12. Oboe G4 tone with slow introduction of subharmonics through ASpSB with $c/m = 3{:}2$. I is initially set at 0 and then increased from 0 to 5.

THE AUTHORS

V. Lazzarini

J. Timoney

T. Lysaght

Victor Lazzarini is a senior lecturer in the Music Department and director of the Music Technology Laboratory at the National University of Ireland, Maynooth. A graduate of the Universidade Estadual de Campinas (UNICAMP) in Brazil, he completed his doctorate at the University of Nottingham, UK, in 1996.

Working in the area of computer music, both as a researcher and as a composer, his work is directed toward sound synthesis, musical signal processing, and computer music languages.

●

Joseph Timoney studied electronic engineering at Trinity College, Dublin, Ireland. Following this, on obtaining a research studentship from Teltec Ireland, he pursued postgraduate studies in TCD. In 1994 he completed an M.Sc. degree by research in the area of telecommunications and in 1999 was awarded a Ph.D. degree for his work in the area of robust adaptive signal processing.

In September 1999 he joined the Department of Computer Science at NUI Maynooth as a lecturer. He is involved in the undergraduate programs in computer science and also in music technology. His research interests are based in the area of audio signal processing, with a focus on musical sound synthesis. In 2003 he spent a three-month research visit at ATR laboratory in Kyoto, Japan. He is a member of the Audio Engineering Society.

●

Thomas Lysaght holds a B.A. degree in music and mathematical physics, a higher diploma in education, and an M.Comp.Sc. degree in computer science, all from the National University of Ireland, Maynooth (NUIM).

He is an accomplished musician in both classical and traditional music and is currently a lecturer in computer science at NUIM. His Ph.D. research focuses on time–frequency analysis and synthesis using modal distribution. His particular interest is in the analysis/synthesis of traditional acoustic instruments.