PHASE BASHING FOR SAMPLE-BASED FORMANT SYNTHESIS

Miller Puckette
Center for Research in Computing and the Arts
CALIT2, University of California, San Diego

Reprinted from *Proceedings*, ICMC 2005, pp. 733-736.

ABSTRACT

Many techniques have been proposed for synthesizing the sung or spoken voice; they generally fall into two classes. First, the purely synthetic techniques (FM, FOF, PAF, ...) make sounds with specified formant frequencies, bandwidths, and amplitudes. Second, the sample-based methods (LPC, Vocoders, PSOLA, ...) reconstruct sounds from pre-recorded speech or singing. Here we present a technique that can synthesize spectra either with specified formants, or in imitation of a recorded sound. Since it is not patent encumbered it can be used to replace the PAF operator for open-source re-implementations of pieces using PAFs.

1. INTRODUCTION

The purpose of this paper is to bring about a unification of synthetic techniques for synthesizing sounds with desired spectra with sample-based ones. The synthetic techniques have the advantage of total specifiability, whereas the sample-based ones can be more natural sounding, and are much easier to wield.

Synthetic methods date at least to the invention of subtractive synthesis. In recent years, formant-generating systems have become more popular, starting with VOSIM [**?**] and continuing through FM formant generation [**?**], the FOF [**?**], and the PAF [**?**]. Sample-based ones [**?**, **?**, **?**] have usually used pitch information to make phase coherent cross-fades between segments of recorded speech and singing.

Eckel *et al.* extended the FOF generator to FOG, in which sampled waveforms may replace the FOF's sinusoids. This has been further generalized to include PSOLA [**?**]. In this paper we take a similar approach, extending an earlier effort to synthesize periodic sounds by concatenating wave packets[**?**]. The work reported here is distinguished by careful attention to phase effects during cross-fades to produce desirable spectra even when the sampled signal deviates from periodicity. First we treat the case of purely sinusoidal "grains", and then we discuss how to prepare recorded samples so that they can be used in the same way.

2. SYNTHESIS ALGORITHM

Figure 1 shows a block diagram for a packet-based formant synthesizer. Phases are generated by a phase accumulator (at the top) which repeats at the rate $\omega/2$ where ω is the desired fundamental frequency of the output. The table on the left hand side holds a packet, which is considered as a sampled period of a (real-valued) waveform with period 1. The phases of the partials of the stored waveform are all assumed to be aligned at $n = 0$:

$$p(t) = \sum_{k=0}^{N-1} a_k e^{2\pi k t i}$$

where the coefficients a_k are real-valued. The parameter S is a shift factor; if $S = 1$ each period of the phase generator swipes through the table twice, so that the wavetable is scanned as a phase increment corresponding to a fundamental frequency of ω. The notation $p(t)$ reflects our assumption that the wavetable (and others below) are sampled with sufficiently high resolution and/or interpolated so that their values may be regarded as depending on a continuous parameter t.

The right-hand side table lookup is a windowing function, necessary in case the value of S is not a half-integer. The windowing function is assumed to be zero outside $[-0.5, 0.5]$. In these units, the Hann window for example is $w(t) = 0.5 + 0.5 \cdot \cos(2\pi t)$.

The Hann window provides perfect reconstruction of the waveform $p(t)$ if we make two overlapping copies of the diagram of Figure 1, one-half cycle out of phase, provided further that the parameter S is an integer so that the two copies make an in-phase cross-fade of the waveform $p(n)$. This can be done as diagrammed in Figure 2.

2.1. Making synthetic formants

We now analyze the spectrum of the periodic signal generated in Figure 2 in the special case where the table $p(t)$ (still regarded as holding a continuous function of the parameter t) contains a sinusoid, either the fundamental or a harmonic:

$$p(t) = e^{2\pi h t i}$$

where the integer h is the number of the harmonic. Since the output is a linear function of the waveform, we can then infer the spectrum for an arbitrary choice of $p(t)$.

For simplicity of analysis we will suppose the fundamental is of the form $\omega = 2\pi/N$ so that the resulting

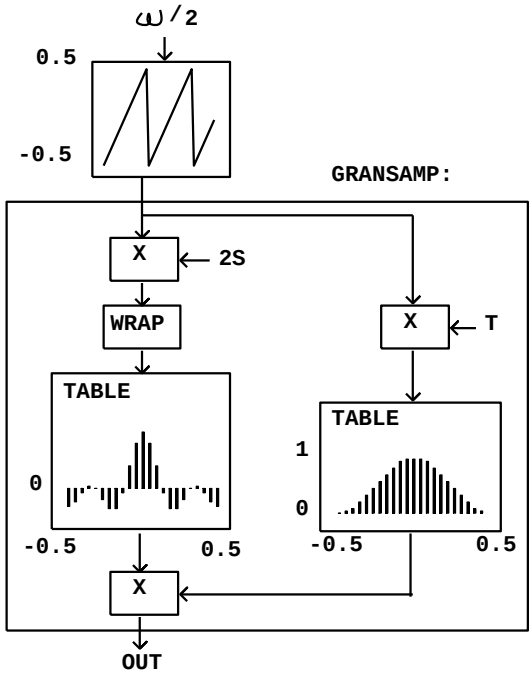

Figure 1. GRANSAMP: Synthesis by windowed wave packets.

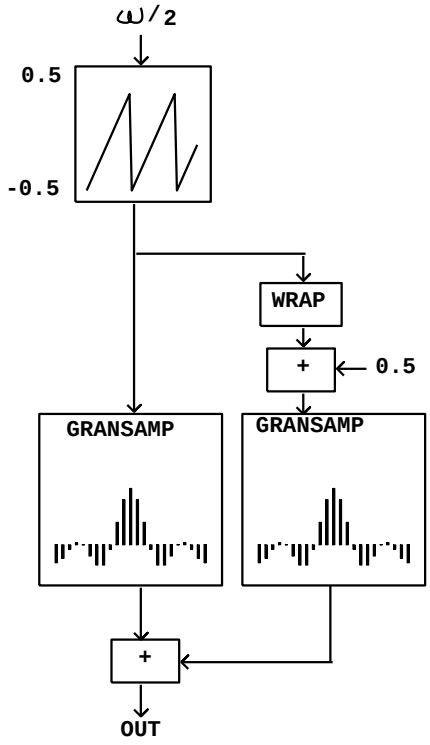

Figure 2. Combining two out-of-phase copies of "GRANSAMP" from Figure 1.

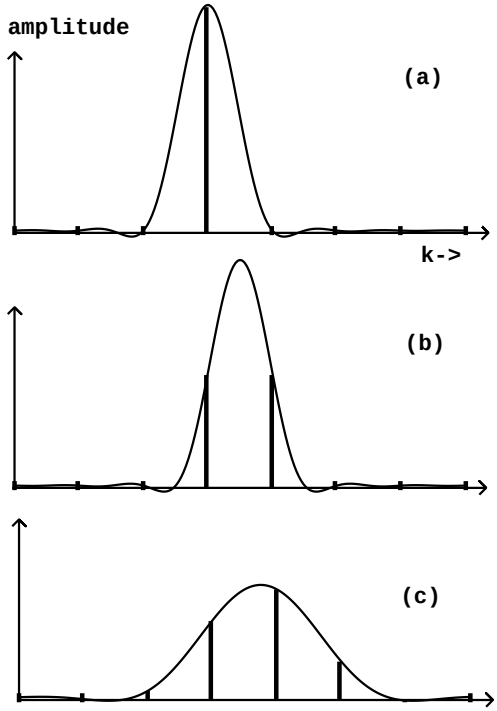

Figure 3. Spectrum of the output of Figure 2: a. $hS = 3$, $T = 1$; b. $hS = 3.5$, $T = 1$; c. $hS = 3.75$, $T = 2$.

period N is an integer; this only amounts to a possible sample rate adjustment which does not affect the end result. For $-N \leq n < N$ the output of the diagram of Figure 1 (with period $2N$) is:

$$x[n] = w(nT/2N)p(nS/N)$$

This may be expressed as a Fourier series:

$$x[n] = \sum_{k=0}^{N-1} b_k e^{2\pi kti/(2N)}$$

where the partial amplitudes b_k are real-valued (phase zero at $n = 0$) and equal to:

$$b_k = \frac{1}{T} W(\frac{k - 2hS}{T})$$

$$W(k) = \frac{1}{4}\mathrm{sinx}(k-1) + \frac{1}{2}\mathrm{sinx}(k) + \frac{1}{4}\mathrm{sinx}(k+1)$$

$$\mathrm{sinx}(k) = \sin(2\pi k)/(2\pi k) \text{ (or 1 when } k = 0)$$

Here $W(k)$ is the (suitably normalized) Fourier transform of the Hann window function. Overlap-adding as in Figure 2 extracts the even-numbered partials:

$$y[n] = x[n] + x[n + N]$$

which gives:

$$y[n] = \sum_{k=0}^{N-1} c_k e^{2\pi kti/N}$$

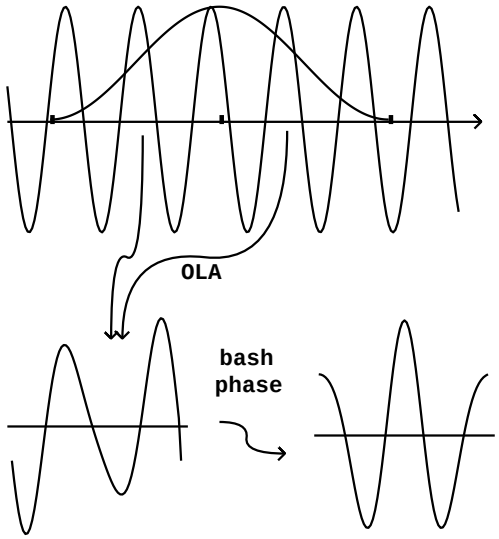

Figure 4. Analyzing an incoming sinusoid to make a wave packet. First the sound is Hann windowed and wrapped around two-for-one to give a waveform. Then the waveform's components are individually bashed into phase.

$$c_k = \frac{2}{T} W(\frac{2(k - hS)}{T})$$

Figure 3 shows the result for three (hS, T) pairs. The spectrum has one formant. Changing the parameter S has the effect of rescaling the spectral envelope along the frequency axis. The parameter T controls bandwidth.

When the minimum bandwidth is selected by setting $T = 1$, as in parts (a) and (b) of the figure, the result is a mixture of consecutive partials; if the center frequency lies directly on a partial, only the one corresponding partial is present in the result. This is the same result as that of setting the bandwidth to zero in the PAF generator.

3. USING RECORDED SOUNDS

The packet function $p(t)$ may be derived from a recorded sound, in such a way that a recorded sinusoid gives a single formant as described in the previous section, but so that a recorded vocal sound, for instance, can give its spectral envelope to the resynthesized result. The technique still has the same possibility of shifting and bandwidth control as in the purely synthetic case.

We make the initial assumption that the signal to be analyzed is a sum of (not necessarily harmonic) sinusoids. Assuming their frequencies are widely enough separated we can treat them individually. We therefore assume that the signal to analyze is a pure sinusoid of frequency $f\omega$ where ω is the fundamental frequency of an analysis period N:

$$\omega = 2\pi/N.$$

Under this assumption, we analyze a sinusoid:

$$r[n] = e^{2\pi hni/N}$$

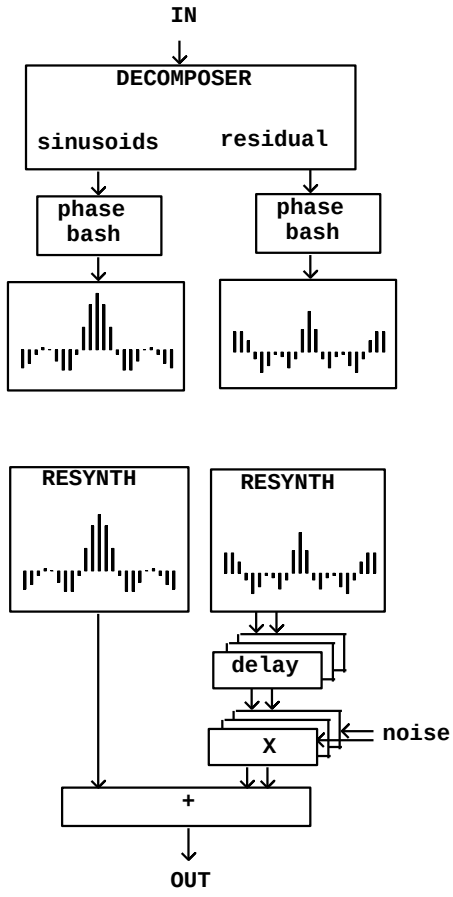

Figure 5. Handling sinusoidal and noisy components separately: a. analysis; b. resynthesis.

by windowing it over a period $2N$ and overlap-adding it to a waveform of period N. Figure 4 shows the result for a sinusoid of frequency 2.4ω. The analysis period is $N = 2\pi/\omega$, so 2.4 periods of the analyzed sinusoid fit into the analysis period. The first step is to window a segment of the incoming sound over twice the analysis period. Next, the windowed signal is overlapped with itself to make one cycle to analyze (the OLA step). Finally, the phases of the signal components are set to zero (the phase bashing step.)

Exactly as in the synthesis step, the combination of a Hann window and a two-way overlap ensures that any sinusoidal component in the analyzed signal is represented in the resulting waveform in the most compact possible way, as a sum of two neighboring harmonics. If the analyzed sinusoid happens to lie on a harmonic of the analysis period, the resulting wavetable is again a pure sinusoid.

Since all the partials of the output have a fixed phase, different wavetables may be cross-faded coherently. For example, it is straightforward to analyze successive frames in a recorded vocal sample and play them back, successively cross-fading the frames to mimic the time-varying spectral envelope of the original sound.

The phase-bashing technique can be combined with sinusoidal/stochastic decomposition [**?**, **?**]. Figure 5, part

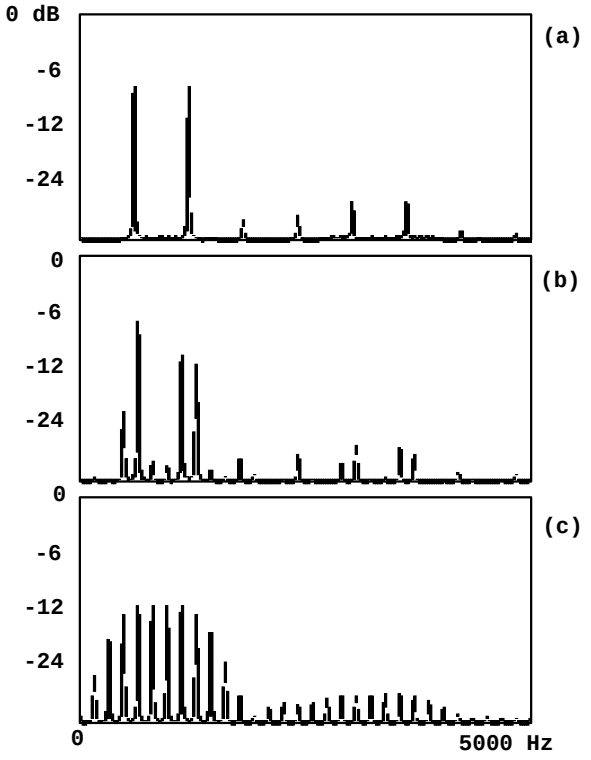

Figure 6. A real analysis/synthesis example: a. the original sound; b. a reconstructed sound with $T = 1$; c. with increased bandwidth.

(a), shows the decomposition step, in which an incoming sound is separated (in real time or not) into sinusoidal and noisy parts. Each is separately divided into a succession of analysis frames and converted into phase-bashed wavetables.

For the reconstruction step (Figure 5 part b), the sinusoidal and noisy tables are each used to reconstruct signals as in Figure 2. The noisy part is then de-pitched by modulating it with a band-limited noise signal. For best results several slightly time-shifted copies are modulated separately [**?**].

Figure 6 shows a real situation in which a sung vowel (/a/ as in "la") is analyzed and resynthesized. Part (a) shows an analyzed spectrum of the original voice, whose frequency is about 600 Hz. Two 'formants' have center frequencies about $1.5f_0$ and $5.5f_0$.

In part (b), the voice is resynthesized at a low pitch (about 170 Hz.) to show a sharp reconstruction of the original spectrum. The result has a formant for each harmonic of the original sample.

Part (c) shows the result of adding bandwidth by increasing the T parameter in order to erase the visible quantization of the spectrum of part (b) around harmonics of the original fundamental.

Unfortunately (I am grateful to a reviewer who pointed this out!) the /a/ vowel should normally contain two formants at about 800 and 1000 Hz; since the fundamental frequency of the recorded sample is so absurdly high, the correct formants are not possible to infer from the spectrum. But I'm running out of space and must stop here.

4. REFERENCES

[1] S. Templaars. The vosim signal spectrum. *Interface*, 6:81–96, 1977.

[2] John Chowning. Frequency modulation synthesis of the singing voice. In Max V. Mathews and John R. Pierce, editors, *Current Directions in Computer Music Research*, pages 57–64. MIT Press, Cambridge, 1989.

[3] Xavier Rodet. The chant project: from the synthesis of the singing voice to synthesis in general. *Computer Music Journal*, 8(3):15–31, 1984.

[4] Miller S. Puckette. Formant-based audio synthesis using nonlinear distortion. *Journal of the Audio Engineering Society*, 43(1):224–227, 1995.

[5] F. Charpentier and M. Stella. Diphone synthesis using an overlap-add technique for speech waveforms concatenation. In *Proceedings of ICASSP*, volume 3, pages 2015–2018. IEEE, 1986.

[6] Geffroy Peeters and Xavier Rodet. Sinola: a new analysis/synthesis method using spectrum peak shape distortion, phase and reassigned spectrum. In *Proceedings of the International Computer Music Conference*, pages 153–156, Ann Arbor, 1999. International Computer Music Association.

[7] Norbert Schnell et al. Synthesizing a choir in real-time using pitch synchronous overlap app (psola). In *Proceedings of the International Computer Music Conference*, pages 102–108., Ann Arbor, 2000. International Computer Music Association.

[8] Michael Clarke and Xavier Rodet. Real-time fof and fog synthesis in msp and its integration with psola. In *Proceedings of the International Computer Music Conference*, Ann Arbor, 2003. International Computer Music Association.

[9] Miller S. Puckette. Synthesizing sounds with specified, time-varying spectra. In *Proceedings of the International Computer Music Conference*, pages 361–364, Ann Arbor, 2001. International Computer Music Association.

[10] Xavier Serra. *A System for Sound Analysis/Transformation/Synthesis based on a Deterministic plus Stochastic Decomposition*. PhD thesis, Stanford University, 1989.

[11] Xavier Serra and Julius O. Smith. Spectral modeling synthesis. In *Proceedings of the International Computer Music Conference*, pages 281–283., Ann Arbor, 1989. International Computer Music Association.