音 楽 情 報 科 学 5－6
（1 9 9 4． 2． 4）

# 時間方向の制御を可能とした MOORE型 PHASE VOCODERの拡張

Eric Lyon

慶應義塾大学　環境情報学部

F.Richard Moore(ELEMENTS OF COMPUTER MUSIC 1990)によって、音響を連続的なFFTフレームで分析/再合成するための実用的なフェイズヴォコーダが提唱された。このプログラムでは、再合成に際して逆FFTかオシレータバンクを使用する。
本論文では、Mooreのプログラムをさらに拡張し、再合成における時間的な条件を連続的に変化させる方法を検討する。

# Time Varying Extensions to the Moore Phase Vocoder

Eric Lyon
Research Associate
Keio University, SFC
5322 Endo Fujisawa Kanagawa
252
eric@cmlab.sfc.keio.ac.jp

F. Richard Moore describes a practical phase vocoder program which analyses a sound in successive FFT frames and resynthesises the analysis data using either inverse FFTs or an oscillator bank. Extensions to Moore's program are introduced enabling time-varying alteration of the conditions of resynthesis.

## An Overview of Pv

We will consider the Moore phase vocoder, pv, from a functional standpoint. A detailed explanation of the mathematics, and a complete implementation in C code is given in Moore(1990). Pv, a program written for the Unix environment, takes a stream of floating point time-domain samples as input, transforms these samples in overlapped windowed blocks into an amplitude/frequency spectral representation, optionally multiplies all frequency components by a constant, and then transforms the (possibly modified) spectrum back into time-domain samples which are written as output. The efficient FFT algorithm is used for converting the time-series to an amplitude/phase spectrum. The phases are then "unwrapped" to produce an instantaneous frequency representation. For conversion back to the time domain, an inverse FFT is used when possible (that is, when the frequencies have not been modified), otherwise, a less efficient oscillator bank method is employed.

The user is presented with the following parameters:

R N M D I P synt

R - Sampling rate of the input signal
N - Length of the FFT (a power of 2)
M - Window size
D - Decimation of the input signal
I - Interpolation of the output signal
P - Multiplication factor for frequency
synt - Oscillator bank synthesis threshold

The traditional use of the phase vocoder is for independent control of pitch and time. Control of pitch is effected through the P parameter. If P = 2.0, the resynthesized sound will be transposed up an octave, but retain its original duration. Temporal modifications are effected by adjusting the ratio of I/D. If I/D = 2.0, the resynthesized sound will be twice as long as the input, but will retain its original pitch evolution. It is possible to combine both modifications in a single procedure. If P = 2.0 and I/D = 2.0, the synthesized sound will be both twice as long and up an octave from the original, quite the opposite of what would expect from similar analog tape manipulation.

## Considerations for Modifications to Pv

In addition to the traditional uses of the phase vocoder, many others may be imagined, when considering that the representation of a musical signal as a time-varying spectrum is extremely general. Each frequency in the spectrum could be multiplied by a different value, rather than all multiplied by the same constant. This creates many unusual harmonic effects. Similarly, each amplitude value could be multiplied by a different value, essentially implementing a filter. Using the oscillator bank resynthesis method, each frequency could be resynthesized using a waveform other than a sinusoid, creating various distortion effects. Finally, dynamically altering the conditions of resynthesis creates many interesting compositional possibilities. We now turn to a discussion of some representative time-varying processors developed using pv as a model.

## The Tofu Processor

The Tofu processor is a time-varying spectral warping program. In addition to the parameters discussed above, the user specifies two monotonic functions. The first function provides a multiplier for each frequency component (and is therefore of size N/2 where N is the FFT size). The second function specifies a time-varying offset for the frequency multiplication function. To give a sense of how this works, consider a warp function which consists of a "bump" in the middle frequency range, and is 1.0 at all other points. As the offset value varies from 0-1 (scaled internally to N/2), the spectral "bump" is moved across the entire frequency range. A few observations are pertinent. Although the phase vocoder is designed for use with harmonic sounds, it has proved useful for inharmonic sounds and sounds with a fair amount of broadband noise. It is important to note that for such sounds (indeed for all sounds) , the phase vocoder will be more "sensitive" in the higher frequency range, where harmonics are intervallically closer together. By similar reasoning, the "bump" described above will change in intervallic size as it moves up and down the spectrum. (Regarding intervallic size, note that the ratios between successive harmonics become increasingly smaller. For example, 1:2 = an octave, 2:3 = a perfect fifth, a considerably smaller interval, perceptually.)

## The Red Clam Processor

This program uses the possibility of resynthesis with arbitrary waveforms. The user provides two waveforms (which must be of the same length) and an interpolation function. The interpolation function determines relative weighting of each resynthesis waveform in a composite waveform which is calculated for oscillator resynthesis of each spectral frame. As a simple example, if the first waveform is a single period sinusoid, and the second waveform is a double period sinusoid, and the interpolation function is a line from 0 - 1, the resynthesized sound will gradually transform from its normal transposition to an octave higher, but without time modification or pitch glissando. Many more complicated uses of this processor are possible. For example, the resynthesizing waveforms could be considerably more complex. With such a paradigm, frequency aliasing is a considerable risk, and the user has the option of specifying the highest frequency to resynthesize. This has the side effect of increasing the speed of the processor as fewer waveforms are synthesized. Another option is to provide a sieve function which turns off specific frequency bins. This can considerably increase the speed of the processor, achieving performance comparable to the inverse FFT in some cases. For the purpose of saving computation time, pv provides a synthesis threshold below which oscillator resynthesis is suppressed. The threshold is a constant, and Moore leaves it as an exercize to make the threshold vary with the maximum reported amplitude (Conservatively, as 60dB below the maximum). This feature has been found to be necessary for any signals with considerable dynamic variation, and is implemented in all processors discussed here which use oscillator resynthesis.

## The Resident Processor

One limitation of the above processors is that only one frame of spectral information is available at a time. The resident processor was inspired by a feature of the UPIC system designed by Iannis Xenakis. The UPIC system provides a graphic spectral representation of a synthetic texture. It is possible to improvise on this texture by moving a mouse to various locations. Resident mimics this paradigm by storing an entire phase vocoder analysis in memory, and then traversing this spectral database according to a user specified function.

### The Resent Processor

Resent, like resident, operates on a spectral database. However, whereas resent resynthesizes particular spectral frames at specific times, without modification to the frames themselves, resent synthesizes spectral frames based on the idea of different harmonics being resynthesized at different speeds (including negative speeds, i.e. time reversal). The user provides a function specifying the resynthesis speed for each harmonic. The user may optionally provide a second function specifying the initial position in the database for each harmonic. Many different results are possible with this processor, most obviously interesting looping effects as blocks of harmonics move in and out of phase. Different phasing effects are produced when each harmonic begins at the same position and moves at a speed close to that of its neighbor.

### The Jones Synthesizer

One other interesting possibility is to omit the analysis portion of pv entirely and algorithmically specify a target spectrum for resynthesis. Jones is an example of this approach. The jones model is of independent linear motion for each harmonic. The user specifies the minimum and maximum durations for segments. The algorithm creates a beginning and end amplitude for each segment (for each harmonic). At each segment, the harmonic is given either a weak amplitude or a strong one, where the weak amplitude is scaled by a user specified parameter. The user also specifies the odds against a harmonic receiving a strong amplitude. Finally, the user may optionally specify a filter function which rescales the amplitudes of the harmonics. The frequencies are set close to the center of each frequency bin, with a small percentage of deviation from frame to frame. As the frequency deviation is not large enough to require oscillator bank resynthesis, jones benefits from FFT synthesis as well as omission of the analysis procedure.

## CONCLUSIONS

The Moore implementation of the phase vocoder offers a useful starting point for exploring a wide range of sound processing possibilities. Some representative processors have been presented. However there is much room for research and experimentation, both in the use of the processors described above, and the development of new processors. Many possibilities exist for designing more intelligent and interesting spectral synthesis algorithms using the jones model alone.

References:

Mark Dolson, "The Phase Vocoder: A Tutorial." Computer Music Journal, 10(4) (1986)

John W. Gordon and John Strawn, "An Introduction to the Phase Vocoder" in "Digital Audio Signal Processing - An Anthology", (Ed.,John Strawn). William Kaufman, pp. 221-265 (1985)

F. Richard Moore, "Elements of Computer Music." Prentice-Hall, pp. 29-263 (1990)

James A. Moorer, "The Use of the Phase Vocoder in Computer Music Applications," Journal of the Audio Engineering Society 24(9), pp.717-727 (1978)

Christopher T. Penrose, "Practical Signal Processing: Filtering,Interpolating and Enriching Digital Signals with the Handy Phase Vocoder Algorithm." Proceedings of the 1992 CCMR(Center for Computer Music Research) Computer Music Conference,Delphi, Greece (1992)

L.R. Rabiner, R.W. Schafer, "Digital Processing of Speech Signals."Prentice Hall, pp. 250-310 (1978)