# ADAPTIVE FM SYNTHESIS

*Victor Lazzarini, Joseph Timoney and Thomas Lysaght*

Sound and Music Technology Research Group
National University of Ireland, Maynooth
Victor.Lazzarini@nuim.ie, {JTimoney,TLysaght}@cs.nuim.ie

## ABSTRACT

This article describes an adaptive synthesis technique based on frequency (phase) modulation of arbitrary input signals. The background and motivation for the development of the technique, as well as related work, are discussed. A detailed description of delay line-based phase modulation of sinusoidal and complex signals is provided. The basic design of an implementation of the technique is presented and commented. A series of examples using four different instrumental sources are discussed. The results show a wide range of possible effects through the use of the technique, from addition of higher components, to changes in the odd-even harmonic balance and the introduction of controlled inharmonicity.

## 1. INTRODUCTION

Adaptive digital audio effects[1] form an important subset of musical signal processing techniques. A key aspect of their usefulness in music composition and performance is that they provide a means to retain significant gestural information contained in the original signal. One of the major criticisms of electronic and computer music is the relative lack of gestural control over the sonic result. With the use of adaptive techniques, it is possible to re-inject much of the liveliness perceived in musical signals of instrumental origin.

Frequency modulation (FM) synthesis[2] is classic synthesis method which has been very useful as an economic means to generate time-varying complex spectra. However, one of its limitation s, as is the case with many of the classic techniques, has always been the difficulty in producing more natural-sounding spectral evolutions, due to the lack of fine gestural control over the sound. The usual methods of controlling FM synthesizers, such as keyboards, modulation-wheels, sliders and breath controllers never provided more than a basic means of dynamically modifying the synthesized signal. The limitation might be related to the lack of resolution of these gestural controllers, but it is also related to the paradigm of control employed.

The traditional approach has been to treat synthesis and control parameters separately, which ultimately will lead to a split between gesture and sound result. An alternative is provided by approaching the problem from the adaptive point of view, whereby the synthesis parameters are derived from the input signal itself. In addition, in our proposed method, the input signal itself is used in the synthesis process. By extracting significant information from the input signal and applying it as a way of controlling the modulation of the same signal, we have arrived at a gesture-aware form of FM synthesis, Adaptive-FM (AdFM).

### 1.1. Related work

The technique described here lies in the area defined by Poepel and Dannenberg[3] as 'audio-signal-driven' sound synthesis. These authors have proposed a number of correlate techniques, including a FM synthesis method using instrumental input as modulation sources and a variation on the technique proposed here, which they named 'self-modulation'.

Their FM synthesis approach differs from AdFM in that it is a case is of single-carrier complex modulation, whereas here we propose a technique similar to multi-carrier FM. As hinted above, the method of self-modulation uses a similar basic signal processing principle. However, it differs substantially from our approach in very significant points (which will generally result in the lack of fine control over the process). These differences will be discussed below.

More traditionally, some classic signal processing algorithm designs also lie in the same general area. Different methods of waveshaping[4][5] of arbitrary input signals, as well as single-side band[6] and ring modulation are very much related to the present work. However, none of these provide as fine control over the resulting synthetic output as ADFM.

## 2. THE TECHNIQUE OF ADFM

The synthesis method provided here is based on two elements: the employment of a variable delay line as a means of phase modulation of a signal and the use of an arbitrary input, to which parameter estimation will be applied.

### 2.1. Delay line-based phase modulation

A well-known side-effect of variable delays is the phase modulation of the delay line input. This is the basis for all classic variable-delay effects such as flanging, chorus, pitch shifting and vibrato. It is thus possible to model simple (sinusoidal) audio-rate phase modulation using a delay-line with a suitable modulating function (Fig.1).

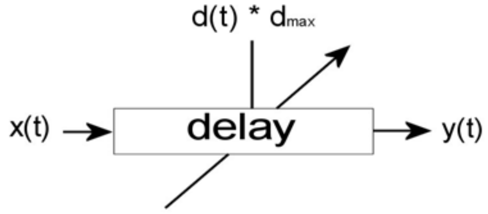


Figure 1. *Delay-line phase modulation*

We now consider the case where the input to the delay line is a sinusoidal signal of frequency $f_c$ :

$$x(t) = \sin(2\pi f_c t) \quad (1)$$

The instantaneous frequency *IF(t)* of the phase-modulated signal is given by the following relationship [7]:

$$IF(t) = -\frac{\partial d(t)}{\partial t} d_{\max} f_c + f_c \quad (2)$$

where *d(t)* is the modulating signal and $d_{max}$ is the maximum delay in seconds. Using a scaled raised cosine as a modulating function

$$d(t) = 0.5\cos(2\pi f_m t) + 0.5 \quad (3)$$

we have (by substituting *d(t)* in Eq.2)

$$IF(t) = \pi f_m \sin(2\pi f_m t) d_{\max} f_c + f_c \quad (4)$$

which characterises sinusoidal frequency (phase) modulation. In such arrangement, the sinusoidal term in Eq(4) is known as the frequency deviation, whose maximum absolute value $DEV_{max}$ is:

$$DEV_{\max} = \frac{\Delta d \pi}{T_m} f_c = \Delta d \times \pi f_m f_c \quad (5)$$

with $\Delta d = d_{max} - d_{min}$..

Now, turning to FM theory, we characterise the index of modulation $I$ as the ratio of the maximum deviation and the modulation frequency:

$$I = \frac{DEV_{\max}}{f_m} = \frac{\Delta d \pi f_m f_c}{f_m} = \Delta d \pi f_c \quad (6)$$

So, the $\Delta d$ that should apply as the amplitude of our sinusoidal modulating signal can now be put in terms of the index of modulation

$$\Delta d = \frac{I}{\pi f_c} \quad (7)$$

and the modulating signal is now:

$$d(t) = \frac{I}{\pi f_c}\left[0.5\cos(2\pi f_m t) + 0.5\right] \quad (8)$$

The resulting spectra according to FM theory is dependent on the values of both $I$ and the $fc{:}fm$ ratio:

$$y(t) = J_0(I)\sin(\omega_c t) + \sum_{k=1}^{I+1} J_k(I)\sin(\omega_c t + k\omega_m t) + J_{-k}(I)\sin(\omega_c t - k\omega_m t) \quad (9)$$

where $\omega_c = 2\pi f_c, \omega_m = 2\pi f_m, J_k(I)$ are Bessel functions of the 1st kind of order $k$ and

$$J_{-k}(I) = (-1)^k J_k(I) \quad (10)$$

Interestingly enough, in the delay-line formulation of FM/PM, the index of modulation for a given variable delay width is proportional to the carrier signal frequency. This situation does not arise in classic FM. Also, when considering the width of variable delay for a given value of $I$, we see that it gets smaller as the frequency rises. In a digital system, for $I$=1, this will be less than 1 sample at the Nyquist frequency.

### 2.2. Using an arbitrary input signal

In Eq.9, we see the ordinary spectrum of simple FM. However, for our present purposes, we will assume the input to be a complex arbitrary signal made up of $N$+1 sinusoidal partials of amplitudes $a_n$ , radian frequencies $\omega_n$ and phase offsets $\phi_n$ ,originating, for instance, from instrumental sources:

$$x(t) = \sum_{n=0}^{N} a_n \sin(\omega_n t + \phi_n) \quad (11)$$

The resulting phase-modulated output is equivalent what is normally called multi-carrier FM synthesis, since the carrier signal is now complex. This output can be described as

$$y(t) = \sum_{n=0}^{N} a_n \sin(\omega_n t + I_n \sin(\omega_m t) + \phi_n) \quad (12)$$

where $\omega_m$ is the modulation frequency and $I_n$ is the index of modulation for each partial. According to Eq.9, this would be equivalent to the following signal:

$$y(t) = \sum_{n=0}^{N} a_n \left[ \begin{array}{c} J_0(I_n)\sin(\omega_n t + \phi_n) + \\ \sum_{k=1}^{I+1} \left\langle \begin{array}{l} J_k(I_n)\sin(\omega_n t + k\omega_m t + \phi_n) + \\ J_{-k}(I_n)\sin(\omega_n t - k\omega_m t + \phi_n) \end{array} \right\rangle \end{array} \right] \quad (13)$$

The different indexes of modulation for each component of the carrier signal can be estimated by the following relationship:

$$I_n = \Delta d \pi f_n = \frac{I}{\pi f_o} \pi f_n = I \frac{f_n}{f_o} \quad (14)$$

Again, we see here that the effect of the relationship between the index of modulation and the carrier frequency is that higher partials will be modulated more intensely than lower ones. Depending on the bandwidth and richness of the input signal, it is quite easy to generate very complex spectra, which might be objectionable in some cases.

This is indeed the case in the related technique of self-modulation, where both the modulating signal and the carrier are complex. However, since here we have full control of the index of modulation and we have a sinusoidal modulator, it is possible to realise more subtle and controlled FM.

Another key aspect of the proposed method is that the *fc:fm* ratio parameter can be also be taken advantage of by estimating the fundamental frequency of the input signal (assumed to be monophonic). In this case, a variety of different spectral combinations can be produced, from inharmonic to harmonic and quasi-harmonic.

### 2.3. Input signal parameter estimation

In order to allow for a full control of *fc:fm* ratio and modulation index, it is necessary to estimate the fundamental frequency of the carrier signal. That will allow the modulator signal frequency and amplitude to be set according to eq.8. This can be achieved with the use of a pitch tracker, which is a standard component of any modern musical signal processing system. For the current implementation, a spectral analysis pitch tracking method was devised, based on an algorithm by M Puckette et al [8][9], which provides fine accuracy of fundamental frequency estimation. In addition to tracking the pitch, it is also useful to obtain the amplitude of the input signal, which can be used in certain applications to scale the index of modulation. This is also provided by our parameter estimation method.

### 2.4. Bandwidth and aliasing issues

Although the spectrum of FM is band-limited, it is capable of producing very high frequencies, according to eqs.9 and 13. With digital signals this can lead to aliasing problems, if the bandwidth of the signal exceeds the Nyquist frequency. The fact that the index of modulation increases with frequency, for a given $\Delta d$, as seen in eq.14, is obviously problematic. However, in practice, since the kind of input signals we will be employing generally exhibit a spectral envelope that decays with frequency, objectionable aliasing problems might be greatly minimised, given that $a_n$, in eq.13, for higher values of $n$ will be close to 0. Of course, if our input contains a lot of energy in the higher end of the spectrum, such as for instance an impulse train, then aliasing will surely occur.

The simplest solution for such problematic signals is to impose a decaying spectral envelope by the use of a filter. This will have the obvious side-effect of modifying the timbre of the input signal. Another, more computationally costly, solution is to oversample the input signal. This would either remove the aliased signals or place them at an inaudible range.

## 3. IMPLEMENTATION

The basic design of AdFM is shown on fig.2. There are three basic components: i) pitch tracker ii) modulating source (a table-lookup oscillator) and iii) variable delay line with interpolated readout. All of these components are found on modern musical signal processing systems, so the technique is highly portable. The implementation discussed here uses the Python[10] language (for control) and Csound 5[11] (as the synthesis engine). It is important to note that this design can be used either for realtime or 'off-line' applications. In addition, plugins can be easily developed from it using csLadspa [12] .

The equivalent Csound 5 code for the design on fig.2 is shown below, with comments:

```
/* AdFM opcode
  asig AdFM  ain,krat,kndx,ifn

  ain – input signal
  krat – fc:fm ratio
  kndx – index of modulation

    ifn – mod signal function table
*/
opcode AdFM,a,akkiii
 setksmps 1
 ipi = 3.1415926
 ioff = 2/44100 /* 2-sample offset*/
 as,krt,knx,ilo,ihi,ifn xin
 /* pitch tracking */
 kcps,kamp ptrack as,512
 /* modulator */
 adt oscili knx/(ipi*kcps),kcps/krt,ifn
 adp   delayr 1
 /* delay line */
 adel  deltap3  adt + ioff
       delayw  as
       xout adel
endop
```

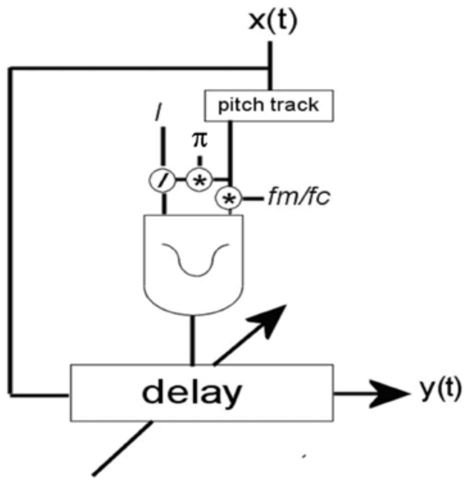


Figure 2. *The Basic AdFM design*

This implementation uses a spectral analysis pitch tracking opcode written by the authors, a linear interpolation oscillator to generate the modulation signal and a cubic interpolation variable delay line. Due to the use of cubic interpolation, the minimum delay is set to 2 samples to avoid errors in the circular buffer readout.

A number of variations can be made to the basic design. For instance, the amplitude of the signal, which is produced together with pitch tracking, can be used to scale the index of modulation. This will generate so called brass-like tones[13], where the brightness of the synthetic output will be linked to the amplitude evolution of the input sound. Alternatively, it can be used to determine the *fc:fm* ratio.

Depending on the characteristics of the input signal, it might be useful to include a lowpass filter before the signal is sent to the AdFM processor. The cutoff frequency of the lowpass filter can also be controlled by the estimated input amplitude. As discussed earlier, this will reduce aliasing as well as overall brightness, which sometimes is a downside of FM synthesis.

The basic design and its variations have been combined in a computer instrument written in Python, using the Csound 5 API, with a GTK-based graphic user interface (using PyGTK). All the synthesis parameters are exposed by the GUI, so the user can adjust the technique to suit his/her needs.

## 4. EXAMPLES AND DISCUSSION

Four different types of carrier signal were chosen as a way of examining the qualities of the AdFM synthetic signal. A flute input with its spectral energy concentrated in the lower harmonics was a prime candidate for experimentation. The clarinet was chosen for its basic quality of having more prominent odd harmonics. Finally, the piano and voice were used as a means of exploring the possibilities of synthesising different types of harmonic and inharmonic spectra by the use of various $f_c$:$f_m$ ratios.

### 4.1. Flute input

The original flute spectrum (steady-state), effectively with *I*=0, is shown on fig.3. As clearly seen in that figure, it features quite prominent lower harmonics. By applying an index of modulation of 0.3, on a 1:1 $f_c$:$f_m$ arrangement, we can start enriching the spectrum with higher harmonics (Fig.4). At these low values of *I*, there is already a considerable addition of components between 5 and 10 KHz. The overall spectral envelope still preserves its original, decaying, shape.

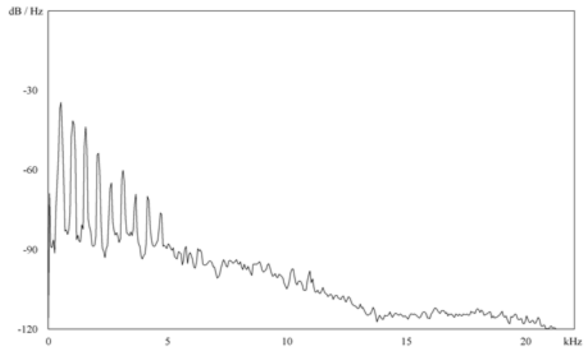


Figure 3. *Steady-state spectrum of a flute playing C4*

With higher values of *I,* we can see a dramatic change in the timbral characteristics of the original flute sound. Fig.4 shows the resulting spectrum, now with *I*=1.5. Here, we can see that components are now spread to the entire frequency range. The original decaying spectral envelope is distorted into a much more gradual shape and the difference between the loudest and the softest harmonic is only of about 20 dB. The resulting sound has been described as 'string-like' and the transition between the flute and AdFM spectra is capable of providing interesting possibilities for musical expression. Also, it is important to note that important gestural characteristics of the original sound, such as pitch fluctuations/vibrato and articulation are preserved in the synthetic output.

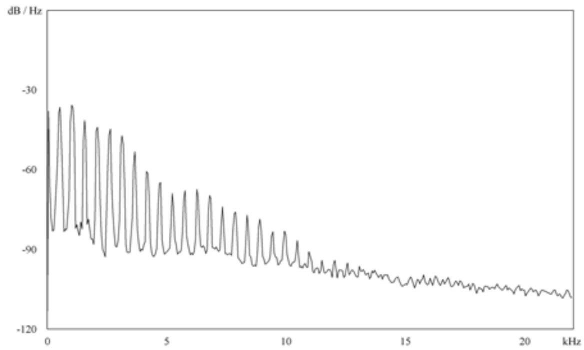


Figure 4. *AdFM spectrum using a flute C4 signal as carrier with fc:fm=1 and I=0.3*

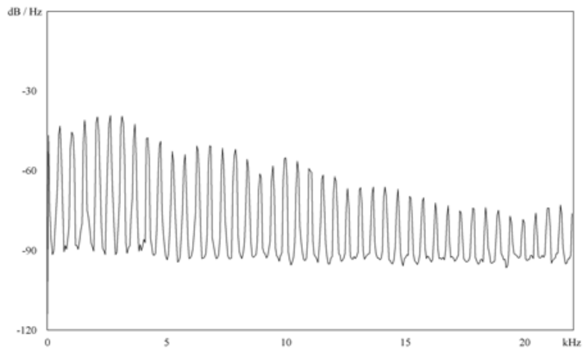


Figure 5. *AdFM spectrum using same input as fig.3, but now with I=1.5*

As *I* gets higher, the spectrum gets even brighter, but the problems with aliasing start to become significant. To prevent this and also to allow for a different spectral envelope, an optional lowpass filtering of the input signal is suggested. In that case, the filter is inserted in the signal path at the delay-line input. A butterworth low-pass filter with a cutoff frequency between 1000 and 5000 Hz has proven useful. It is possible to couple the cutoff frequency with *I*, so that for higher values of that parameter, more filtering is applied.

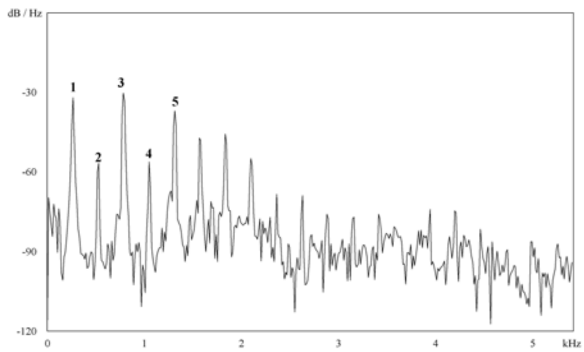


Figure 6. *Detail of steady-state spectrum of clarinet C3. The higher relative strength of lower-order odd harmonics against even ones is clearly seen.*

### 4.2. Clarinet input

Our second experiment used a clarinet signal as a carrier wave for AdFM. The clarinet exhibits a steady-state spectrum where the lower-order even harmonics are significantly less energetic than its odd neighbours (fig.6). Due to this fact, the multi-carrier-like characteristic of AdFM helps generate quite a change in the spectra of that instrument.

As the index of modulation increases, the balance between odd an even harmonics changes substantially. With $I$=1.5, it is possible to see that there is now very little difference between the strengths of odd and even components (fig.7). In addition, higher-order harmonics become more present, and the spectral envelope levels out. This is due to the well-know spread of energy that is characteristic of FM synthesis.

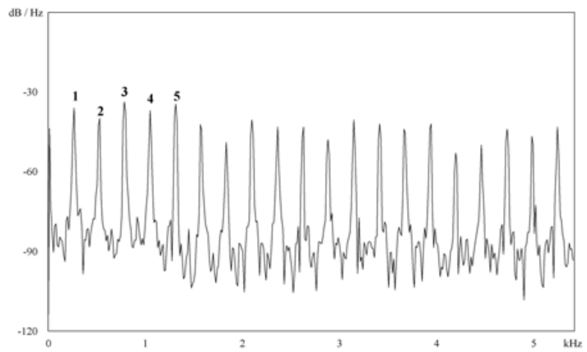


Figure 7**.** *Detail of AdFM spectrum using a clarinet C3 signal as carrier with fc:fm=1 and I=1.5. Odd and even harmonics have now comparable strengths.*

### 4.3. Piano input

In the previous examples, we have kept the ratio between the modulating frequency and carrier fundamental at 1. However, as we know from FM theory, a range of different spectra are possible if we use different ratios. It is possible to create a range of effects that range from changing the fundamental of the sound to transforming a harmonic spectrum into a inharmonic one. We proceeded to take a piano C2 signal as our carrier and then tuned our modulator to 1.41 times that frequency. The original piano spectrum is shown on fig.8, where we can clearly see its harmonics.

The resulting AdFM spectrum, with $I$=0.15 is shown on fig.8. This particular ratio creates a great number of components, whose relationship will imply a very low fundamental, thus generating what is perceived as an inharmonic spectrum. With the 1:1 ratio, the sums and differences between $f_c$ and $f_m$ created components whose frequencies were mostly coincident. Here, a variety of discrete components will be generated, creating the denser spectrum seen in fig.9.. The AdFM sound resulting from this arrangement has been described as 'bell-like'. Transitions between piano and bell sounds can be effected by changing $I$ from 0 to the desired value. The application of a lowpass filter at the delay-line input will also allow for some variety and control over the brightness of the result.

### 4.4. Voice input

A vocal input was used as the fourth different source examined in this work, demonstrating a pitch shift effect. Setting the $f_c : f_m$ ratio to 2, we are able to obtain a sound that is now ½ the pitch of the original. This is due to the introduction of a component at ½ the fundamental frequency corresponding to $f_c - f_m$ in eq.14.

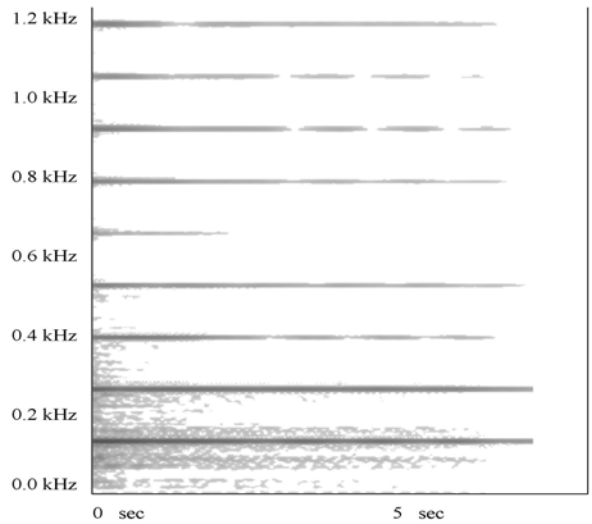


Figure 8. *Spectrogram of a piano C2 tone, showing its first harmonics in the 0-1.2Khz range*

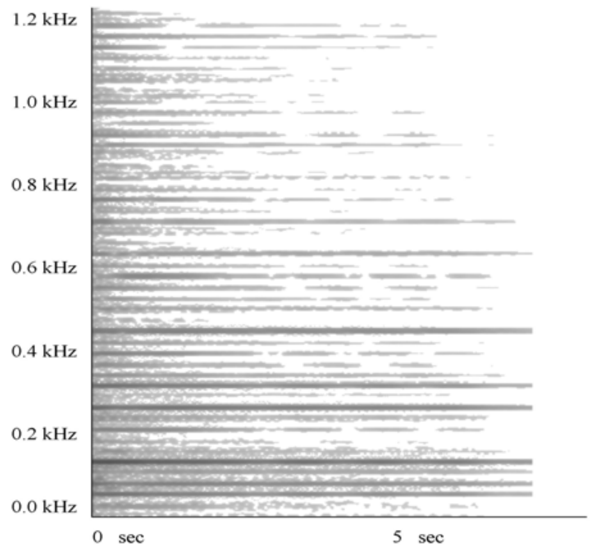


Figure 9. *Spectrogram of an AdFM sound using a piano C2 signal as carrier, with fc:fm=1:1.41 and I=0.5, showing the 0 -1.2Khz range. The resulting inharmonic spectrum, with a large number of components, is clearly seen in comparison with fig. 7*

With the index of modulation at low values (around 0.15), it is possible to preserve some of the spectral shape of the original sound, a crucial step in keeping the intelligibility of the vocal phonemes. Although there is some addition of high frequency components and a flattening of spectral peaks, the AdFM is still perfectly intelligible.

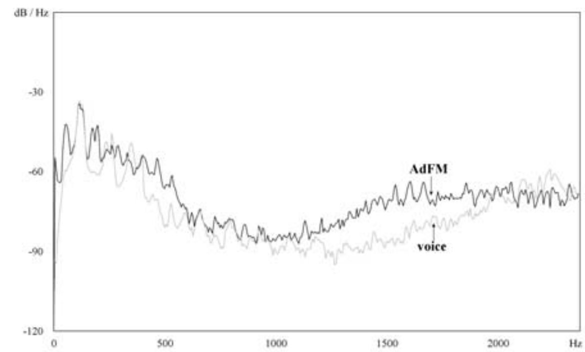


Figure 10. *Comparison of spectral snapshots of a vocal and an AdFM vocal sounds, with I=0.1 and fc:fm=2*

Fig.10 shows a comparison between a vowel steady-state spectrum and its AdFM-processed counterpart. The sub-harmonic peak can be seen at the left of the picture below the original fundamental (a peak at 0Hz is also present, due to the $f_c$ - $2f_m$ component). The recording of the phrase "this is AdFM Synthesis" is shown as a spectrogram on fig.11., both as the original signal (right) and the AdFM output (left), using the same parameters as above. Again, the octave change is clearly seen, as well as the increase in the number of significant components in the signal.

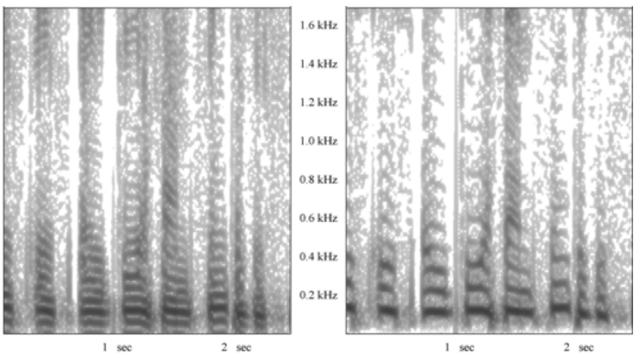


Figure 11. *Detail of spectrogram of a recording of the phrase "this is AdFM synthesis", with AdFM vocal on the left and the original vocal sound on the right*

## 5. CONCLUSION

We have presented here an alternative approach to the classic technique of FM synthesis, based on an adaptive design. This method belongs to a general class of processes that have been called audio signal driven sound synthesis. Since the FM synthesis theory is well known, it was possible to adapt it to provide a good understanding of the output signal. With the present technique it is possible to have a fine control over the synthetic result, which also preserve a substantial amount of the gestural information in the original signal. Four different types of carrier signal were used in this work to demonstrate the wide range of spectra that the technique is capable of. We are confident this is a simple yet effective way of creating hybrid natural-synthetic sounds for musical applications.

## 7. REFERENCES

[1] V. Verfaille and D. Arfib, "Implementation Strategies for Adaptive Digital Effects", *Proc. of DAFx02*, pp. 21-26, 2003.

[2] J. Chowning, "The Synthesis of Complex Audio Spectra by Means of Frequency Modulation," Journal of the Audio Engineering Society, 21, pp. 526-34, 1973.

[3] G. Poepel and R. Dannenberg, "Audio Signal Driven Sound Synthesis", *Proc. of ICMC 05,* pp. 391 – 394, 2005.

[4] M. Le Brun, "Digital Waveshaping Synthesis". Journal of the Audio Engineering Society, 27(4), pp.250-266, 1979.

[5] D. Arfib, "Digital synthesis of complex spectra by means of multiplication of non-linear distorted sine waves" , Journal of the Audio Engineering Society, 27(10), pp.757-779, 1979..

[6] S. Wardle, "A Hilbert-Transformer Frequency Shifter for Audio", *Proc. of DAFx98,* pp.25-29, 1998.

[7] S. Dilsch and U. Zoelzer, "Modulation And Delay Line Based Digital Audio Effects", *Proc. of DAFx99*, 2004

[8] M. Puckette and J. Brown, "Accuracy of frequency estimates from the phase vocoder". *IEEE Transactions on Speech and Audio Processing* 6 (2), pp.116-172, 1998

[9] M. Puckette, T. Appel and D. Ziccarelli. *Real-time Audio Analysis Tools for PD and MSP*. *Proceeding s of the ICMC98*, pp. 109-112, 1998.

[10] G. Van Rossum and F. Drake, *The Python Language Reference Manual*, Network Theory, Bristol, 2003.

[11] J. ffitch, "On the Design of Csound5", *Proceedings of the 3rd Linux Audio Conference*, ZKM, Karlsruhe, Germany, pp. 37-42, 2005.

[12] V. Lazzarini and R. Walsh, "Developing LADSPA plugins with Csound", *Proceedings of the 5th Int. Linux Audio Conf.*, TU Berlin, Germany, 30-36, 2007.

[13] J. C. Risset, *An Introductory Catalogue of Computer Synthesized Sounds*, Bell Laboratories, 1969.