Considerations for a Computational Estimation of the Complex FM Spectrum

Joseph Timoney[2], Victor Lazzarini[2], Tom Lysaght[2], Jari Kleimola[1] and Lorcan MacManus[1]

[1] National University of Ireland, Maynooth, Sound and Digital Music Technology Group, Maynooth, Co. Kildare, Ireland

[2]Aalto University, School of Electrical Engineering, Department of Signal Processing and Acoustics, P.O. Box 13000, FI-00076 AALTO, Espoo, Finland

E-mail: jtimoney@cs.nuim.ie, victor.lazzarini@nuim.ie, tlysaght@cs.nuim.ie, jari.kleimola@aalto.fi, lorcan@gmx.com

Abstract

Complex frequency modulation (FM) is a useful method for sound synthesis as it allows the creation of rich timbres using only a small number of waveforms. However, the mathematical expression for the frequency domain representation of a frequency modulated (FM) wave with a non-sinusoidal modulator is a complex equation with multiple summation terms. Its implementation in software requires nested loops, with the number of loops depending on the number of sinusoidal components required to describe the complex modulator wave. Although the spectral representation can be easily computed from the FFT of the time domain waveform, this is at the expense of the insight that can be gained by direct evaluation of the analytic expression for the frequency domain representation. This paper looks at creating a software implementation of the analytic expression that overcomes the need for many nested loops and can be generalized. It will also suggest how it could be applied in a distributed processing environment.

Keywords
Complex Frequency Modulation, Bessel functions, FM synthesis implementation

1. Introduction

The use of sinusoidal Frequency Modulation (FM) for sound synthesis has been well accepted since the landmark paper of Chowning [1]. This paper showed how the modulation of the frequency (strictly speaking it was the phase) of a sinusoidal signal, termed the carrier, by another sinusoidal signal, termed the modulator, could be used to produce a timbre that was far richer spectrally than either the carrier or modulator themselves. Chowning's ideas were eventually realized in the famous Yamaha DX7 synthesizer, which is closely associated with music of the 1980s [2]. In collaboration with

Chowning, Yamaha developed the DX7 so that the sound creation algorithms on the synthesizer could have more than just a single modulator and carrier connection. The sinusoidal modulators could be arranged in series, in parallel or with feedback before being input to the carrier. Additionally, multiple modulated carriers could be combined together. All these expanded the timbral creation possibilities for the instrument. It also showed, indirectly, that complex frequency modulation (Complex FM) was a more realistic approach for FM synthesis engines if users were to be able to generate intricate sounds, or emulate those of acoustic instruments more accurately. Soon after the DX7, Casio introduced the technique of Phase Distortion as a rival technology [3], although in reality it was a form of complex FM [4]. The popularity of FM synthesizers waned during the 1990s, but FM synthesis has been given a new lease of life in recent times. This came with the release of the FM7 and FM8 VSTs by Native Instruments [5], the development of new FM-based sound synthesis techniques [6], and with work that has shown how an FM-oriented approach can lead to an efficient solution for creating bandlimited oscillators for virtual analogue applications [7].

Most students of computer music will have learnt the expression for simple FM synthesis as it is a staple of many textbooks [8]. However, they may not be as familiar with the expressions for complex FM. This has appeared in the research literature, particularly for 'Double modulator/single carrier FM'. It was used in [9], [10] and [11] for the creation of simulations of real-instrument sounds. More recently, it was found to be very useful for the bandwidth analysis of 'Exponential FM' signals. In this case, the analytical spectrum was found to be far superior to the FFT as it provided a continuous spectral envelope from which a highly accurate bandwidth measurement could be easily made [12]. Furthermore, unlike the FFT spectrum, this representation was not limited by the sampling frequency or subject to corruption by aliasing distortion. Another application area is in the design of bandlimited modulation functions for Periodic Linear Time-varying (PLTV) filters [13]. These PLTV filters transform their input into a Complex FM signal. The analytic spectral representation must be used to calculate the resulting bandwidth of the filter output so that an appropriate sampling frequency can be chosen for the PLTV implementation. Thus, there are occasions when the analytical spectral representation of a complex FM signal is distinctly advantageous over an FFT-based approach.

A major issue in writing a program to evaluate the general expression for the analytical spectrum of the Complex FM signal is that multiple nested loops can be required, depending on the number of sinusoidal modulators. By examining the expression, a more efficient approach was identified that minimizes this requirement. As a result the depth of nested loops executed by the program will not increase regardless of the number of modulation terms. Also, if a distributed computing environment is available it is possible to deconstruct this approaches computations into a form that suitable for implementation on this type of platform. This paper gives the background to the Complex FM expression, then details our implementation, and finishes with some discussion and conclusions.

2. Complex Frequency Modulation

To begin with we will consider a general FM synthesis equation where the carrier is a sinusoid and the modulation is some time-varying function $f(t)$

$$s(t) = \sin(\omega_c t + f(t)) \tag{1}$$

where the carrier frequency in radians is ω_c

The simple case is where the frequency modulation is a cosine and thus by integration the phase modulation can be found

$$f(t) = \int \omega_D \cos(\omega_m t) dt = I \sin(\omega_m t) \tag{2}$$

where the modulating frequency in radians is ω_m and I represents the modulation index, the ratio of the peak frequency deviation, ω_D, to the modulating frequency, that is

$$I = \frac{\omega_D}{\omega_m} \tag{3}$$

Substituting (2) into (1) gives

$$s(t) = \sin(\omega_c t + I \sin(\omega_m t)) \tag{4}$$

The Fourier series for an equation of the form of (4) can be found in terms of Bessel functions of the first kind. This is available in many computer music textbooks, [8] and [13] for example. It is

$$\sin(\omega_c t + I \sin(\omega_m t)) = \sum_{k=-\infty}^{\infty} J_k(I) \sin(\omega_c t + k\omega_m t) \tag{5}$$

Considering the carrier frequency as a central point, (5) states that the spectrum contains an infinite number of components extending to the left and right of this at both positive and negative frequencies located at $\omega_c + k\omega_m$ and $\omega_c - k\omega_m$ respectively. These components are termed sidebands. The magnitude of each component is $J_k(I)$, a Bessel function order is given by the index k and whose argument is the modulation index I. Negative frequency components are reflected back into the positive spectrum [1].

When the modulator $f(t)$ of (1) is not a simple sinusoidal signal the expansion of (5) grows in complexity in relation to the number of harmonic components that can describe the modulation. For example, if a purely sinusoidal expansion, with a maximum of K components, is used to describe $f(t)$ then we can write

$$d(t) = \sin(\omega_c t + f(t) + \theta) = \sin\left(\omega_c t + \sum_{i=1}^{K} I_i \sin(i\omega_m t + \varphi_i) + \theta \right) \tag{6}$$

where θ is an arbitrary phase shift of the carrier, and the function *f*(*t*) is described by a number of harmonically related components of fundamental frequency ω_m, each one with a phase shift of φ_i and magnitude I_i. The spectrum of (6) can be written as

$$\sin\left(\omega_c t + \sum_{i=1}^{K} I_i \sin(i\omega_m t + \varphi_i) + \theta\right) = \sum_{k_k}\cdots\sum_{k_1}\left(\prod_{i=1}^{K} J_{k_i}(I_i)\right)\sin\left(\omega_c t + \left(\sum_{i=1}^{K} k_i(i\omega_m t + \varphi_i)\right) + \theta\right) \qquad (7)$$

This expression has appeared in a number of works from early papers on communications technologies [14] to later articles in the Computer Music Journal ([9] and [15]). It is also important to other fields such as radar signal processing. Comparing (7) and (5) it is clear that the interpretation is much more difficult. General observations on the shape of a spectrum of a Complex FM signal, as given by (7), include:

The energy in the sidebands tends to distribute itself in accordance with the shape of the modulating wave [14]. In general, the effective bandwidth of the signal depends on whether or not the harmonics of the modulating signal add in phase, in a way that will increase or decrease the peak frequency deviation of the resulting signal.

The sidebands produced by the modulation of the carrier by the first modulating oscillator are modulated again as a carrier by the next modulation term. This process repeats for each successive modulation term [16].

3. Implementation of Complex FM

A direct implementation of (7) would require the use of nested loops. One nested loop is for each modulation term. This could easily become very large and time consuming. Furthermore, if the number of time points for analysis is also big this could act as a further drain on processing resources.

It is difficult to see immediately how (7) could be re-interpreted to speed up its implementation. First, in theory the Bessel function indices given by k_i will go from $-\infty$ to ∞. This needs to be truncated to some reasonable limit. An accepted approximation for the bandwidth (*BW*) of an FM signal is to use Carson's rule [17], which states

$$BW = \Delta f + 2 f_{\max} \qquad (8)$$

where Δf is the total width of the band traversed by the instantaneous frequency and $f_{\max}$ is the maximum frequency present in the modulating signal. [17] cautions, however, that this rule will not apply to all cases. In particular, this was not designed with digital signals in mind and it is most likely that most of the time it will not be stringent enough to act as a limit for the prevention of audible aliasing distortion. An alternative approach is to analyze Bessel functions at different orders for various values of argument. It can be observed that, in all cases, as the order increases the amplitude of the Bessel function reaches zero. This is illustrated in Figure 1 for arguments of 0, 5, 10 and 20 respectively.

From the Figure it can be seen that the greater the value of the argument the higher the order before the amplitude becomes zero.

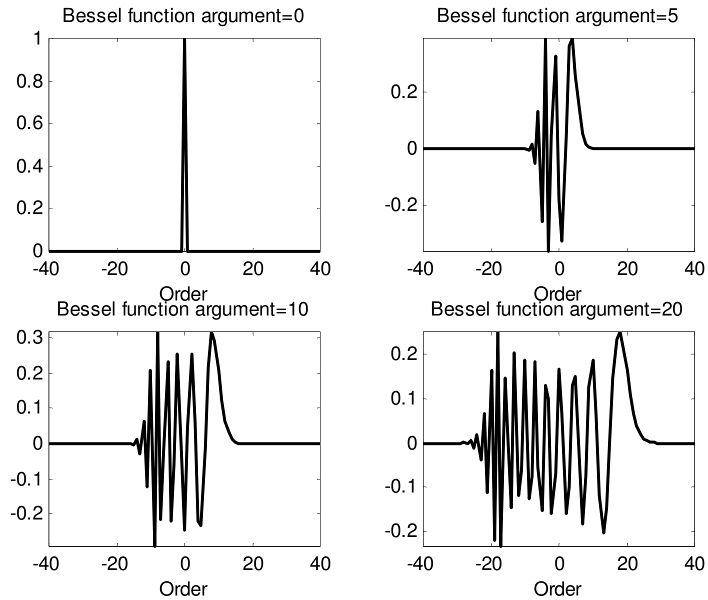


Figure 1 *Relationship between Bessel Function order and its argument for values of 0, 5, 10, and 20.*

To find an upper limit we try to measure a relationship between the value of the argument and the order at which the Bessel function will be a consistently low value.

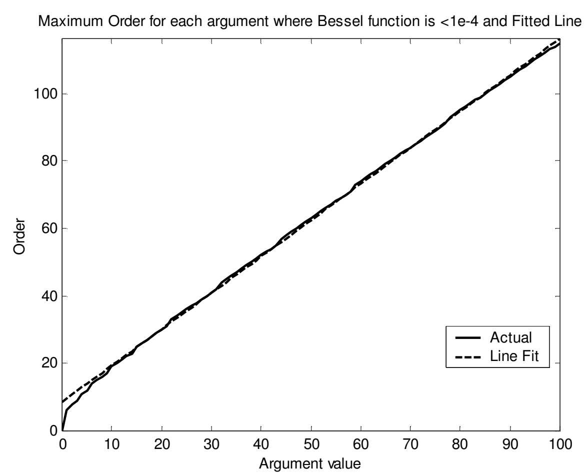


Figure 2 *Bessel function order for different arguments at which the value of the function is consistently below $1x10^{-4}$*

For example in Figure 1 in the bottom right panel it can be seen that for an argument of 20 the Bessel Function will reach a value close to zero at approximately an order of 30. If we measure the point at which the Bessel function consistently has a low value for a range of arguments we can find a relationship. Figure 2 shows a plot of the point where the Bessel Function value is consistently smaller than $1x10^{-4}$ (i.e. -80dB which is sufficient to prevent audible aliasing) for arguments ranging from 0 to 100. Results are only shown for the case where the order is positive, because it can be shown [15] that the positive order amplitude is related to the negative order amplitude by

$$J_{-k}(I) = (-1)^k J_k(I) \qquad (9)$$

In Figure 2 it can be seen that the relation is almost linear. Fitting a line as shown in figure 2 we get the expression

$$k_i = 1.2185 I_i + 5{,}625 \qquad (10)$$

This provides an approximate upper limit on k_i depending on the modulation index I_i. Note that the value of k_i must be an integer so the ceiling of (10) should be computed. Note too that due to the product term of the Bessel functions in (7) the overall spectral magnitude at a particular frequency could possibly be very small (of the order $10^{-4(K)}$). A qualification could also be added to (10) because in the case where the modulation index is less than 0.2 the order is simply 1, that is

$$\text{If } I_i \le 0.2 \text{ then } k_i = 1 \qquad (11)$$

This is known as narrowband FM [17] where, in the single sinusoidal modulator case, the magnitude spectrum is almost the same as that of sinusoidal amplitude modulation (AM).

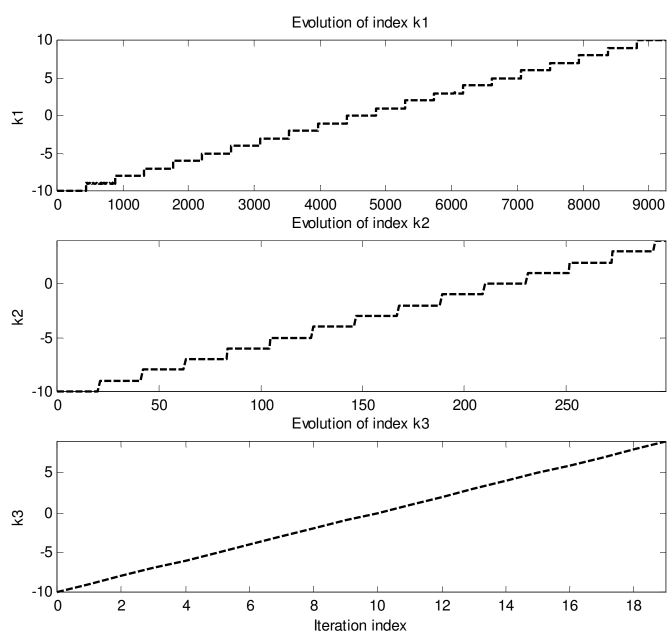


Figure 3 *Sample evolution of Indices for a three modulator signal*

Next, it is worth examining the evolution of the indices k_i in (7). There are as many indices as modulation components. If the evolution of the indices is plotted a pattern can be observed. Figure 3 illustrates this for a signal with three modulators (K=2) whose modulation indices are I_1=1, I_2=0.7, and I_3=0.2. Using (10), we obtain maximum values for k of k_1=6, k_2=5, and k_3=5. In Figure 3 it can be seen that the indices are periodic, with those of k_3 having period 1. The shapes of the evolutions are approximately like either a staircase or sawtooth. This suggests that one could streamline the implementation of (7) if this phenomenon could be exploited.

If we define a waveform period as

$$T = \prod_{i=1}^{K} (2k_i - 1) \tag{12}$$

and a time vector as

$$\tau = \frac{[0, \ldots, T-1]}{T} \tag{13}$$

Defining a vector of maximum indices

$$\mathbf{k}_i = 2[k_1, \ldots, k_K] - 1 \tag{14}$$

where the scaling by 2 and subtraction of 1 is because the indices must be symmetric around 0. Another k vector must also be written

$$\mathbf{k}_f = 2[1, k_1, \ldots, k_K] - 1 \tag{15}$$

and a rising sawtooth function of frequency ω_0 at time t which is written as

$$x(t) = saw(\omega_0, t) \tag{16}$$

Then the evolution of each index k_i can be written as

$$x_i(\tau) = 0.5(\mathbf{k}_i) saw\left(2\pi \prod_{l=1}^{i} \mathbf{k}_f(l), \tau\right) - 0.5\left(saw\left(2\pi \prod_{l=1}^{i} \mathbf{k}_i(l), \tau\right)\right) \tag{17}$$

where i=1,…,K-1, and assuming that the vectors $\mathbf{k}_i$ is indexed from 1 to K, and $\mathbf{k}_f$ is indexed from 1 to K+1.
In the case where i=K, then the expression is simpler

$$x_{K-1}(\tau) = 0.5(\mathbf{k}_{K-1}) saw\left(2\pi \prod_{l=1}^{K} \mathbf{k}_f(l), \tau\right) + 0.5 \tag{18}$$

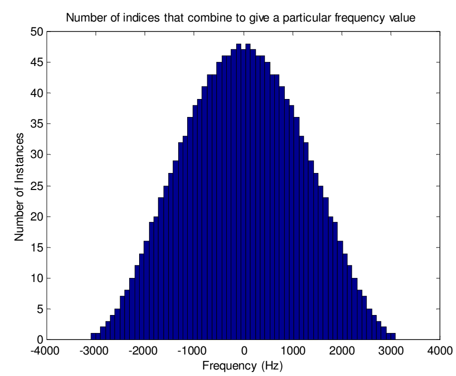
The total number of frequencies at which (7) will be evaluated will be

$$\prod_{i=0}^{K-1} 2k_i - 1 \qquad (19)$$

Note that the expression for (19) is the same as that for (12). However, there will be duplicates of frequency values because of the way indices combine. Assuming that the modulation is harmonic the total number of unique frequencies will be

$$1 + 2\sum_{i=1}^{K} (i+1)k_i \qquad (20)$$

It was found through experiment that the distribution of duplicates will approximately follow a Gaussian distribution. Figure 4 demonstrates this for the example given above. This is a histogram illustrating the distribution of the number of combinations of the index values generated by (17) after multiplication by the harmonic frequencies of the modulator (ω_m, 2 ω_m, 3 ω_m,…, K ω_m). In the plot it is assumed that the carrier frequency is 0Hz, however, in the actual signal this will be translated to the true carrier frequency, ω_c.



s

Figure 4: *Number of instances of duplicates of sideband frequencies after combination of indices in (16) and (17) with harmonic frequency values of the modulation*

The Bessel Function evaluations using the indices returned by (17) and (18) will also follow a periodic pattern. This is shown in Figure 5. As would be expected the periodicity will mirror that of the indices in Figure 3.

There are two possibilities for making the Bessel function evaluations in (7). The first is to do it directly by substituting in the index values k_i. However, because of duplications in the indices many evaluations will be repeated. Alternatively, we could make an evaluation only for the unique set of indices and use this to create a repeating waveshape that matches those in Figure 5. This second option can be implemented using scaled and shifted Pulse Width Modulation functions in a similar manner to a zero-order hold operation [18]. Another consideration is the actual computation of the Bessel functions themselves. Many computing languages have a procedure available in a math library for computing them. However, more efficient alternatives have been proposed, for example see [19], [20], and [21]. In particular, the trigonometric approximations of [21] could be easily implemented. However, they are only available up to a maximum modulation index of 9 [21].

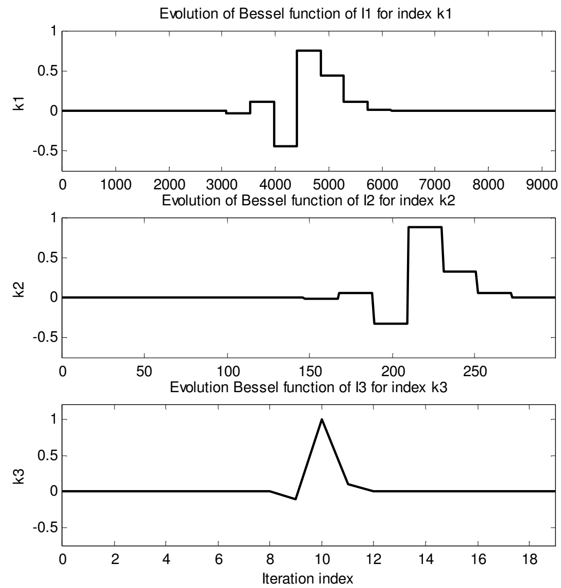


Figure 5: *Periodicity in evaluation of Bessel function evaluations at successive indices for each value of the modulation index*

3.1 Implementation outline

Table 1 shows an outline of the algorithm to compute (7) and its spectrum using equations (12)-(18). The common programming convention that the first index of an array is 0 is adopted. The input is the carrier and modulation frequencies, ω_c and ω_m, an array of modulation indices $\mathbf{I}=[I_0\ I_1\ \ldots.]$ and an array of phases $\boldsymbol{\varphi}=[\varphi_0\ \varphi_1\ \ldots.]$, the carrier phase shift θ and, finally, an array of time values t.

The vectors $\mathbf{k}_i$, $\mathbf{k}_f$, and the scalar T are initialized using (12)-(15). Then, (17)-(18) provides the evolution of the index values. Once these are known the frequencies Ω, phases Γ and Amplitudes A (i.e. the product of the Bessel functions at each index) in (7) can be found. The frequencies at which (7) is evaluated are Ω (see (19)) while the number of unique frequencies is given by (19). If we iterate through all the Ω, we can locate all duplicates of each unique frequency and sum their amplitudes together to get the set of spectral magnitudes, which we denote by $D(\omega)$. The time domain signal $d(t)$ can also be synthesized directly from (7) too if desired.

Function *ComplexFM*{ω_c, ω_m, $\mathbf{I}$, $\boldsymbol{\varphi}$, θ, t } {
K=length($\mathbf{I}$);
T=1;
TotalNumFreq=0;
TotalUniqueFreq=0
for *index*0=0:K-1 {
$k_i = ceil(1.2185\, I(index0) + 5.625)$;
T=T*2*(k_i-1);
$\mathbf{k}_i(index0) = 2k_i - 1$;
TotalUniqueFreq= *TotalUniqueFreq* +1+2*(*index*0+1)*k_i;

}

$\mathbf{k}_f = [1 \quad \mathbf{k}_i]$;

for *index*1=0 to *T*-1 {

for *index*2=0 to *K*-1 {

$\tau(index2) = index2/T$;

if *index*1<*K*-1

$$x_i(index1, index2) = 0.5(\mathbf{k}_i(index2))saw\left(2\pi \prod_{l=0}^{index2} \mathbf{k}_f(l), \tau(index2)\right) - 0.5\left(saw\left(2\pi \prod_{l=0}^{index2} \mathbf{k}_i(l), \tau(index2)\right)\right);$$

else

$$x_i(K-1, index2) = 0.5(\mathbf{k}_i(K-1))saw\left(2\pi \prod_{l=0}^{K-1} \mathbf{k}_f(l), \tau\right) + 0.5;$$

$$\boldsymbol{\omega}(index1, index2) = \omega_m * index2 * x_i(index1, index2);$$

$$\tilde{\boldsymbol{\varphi}}(index1, index2) = \boldsymbol{\varphi}(index2) * x_i(index1, index2)$$

$$\mathbf{J}(index1, index2) = J_{x_i(index1, index2)}(I(index2))$$

}

$$\Omega(index1) = \sum_{index2} \boldsymbol{\omega}$$

$$\Gamma(index1) = \sum_{index2} \tilde{\boldsymbol{\varphi}}$$

$$A(index1) = \prod_{index2} \mathbf{J}$$

}

TotalNumFreq=*T*;

for *index*3=0 to *TotalUniqueFreq*-1 {

FreqIndex=*index*3-*TotalUniqueFreq*/2;

*index*F=[];

for *index*4=0 to *TotalNumFreq*-1 {

if $\Omega(index4) == FreqIndex * \omega_m$

[*index*F]=[*indexF* *index*4];

}

$$D(\omega_c + FreqIndex * \omega_m) = \sum_{indexF} A$$

}

if *resynthesis*==True

$$d(t) = \sum_{index1} A \sin((\omega_c + \Omega)t + \Gamma + \theta)$$

}

Table 1: *Suggested algorithm for computing the magnitude spectrum and time-domain signal for the Complex FM equation.*

3.2 Sample Results

Figure 6 provides an example of the program output. The parameters used were $\omega_c = \omega_m = 2\pi 100$rad/s, **I**=[1 0.7 0.2], **φ**=[0 0 0], and θ=0. The upper panel of Figure 7 shows the waveform output which perfectly matches the direct implementation using the left-hand side of (7). The lower panel shows the magnitude spectrum computed using the FFT with the solid line. The Spectral magnitudes, computed using the program in Table I, are also shown with asterisks. These can be seen to coincide perfectly to the peaks of the FFT output, verifying the accuracy of the program implementation. The execution time was 0.157 seconds using the Matlab programming environment on a Dell D620 Latitude laptop with an Intel core2 processor running at 1.66Ghz.

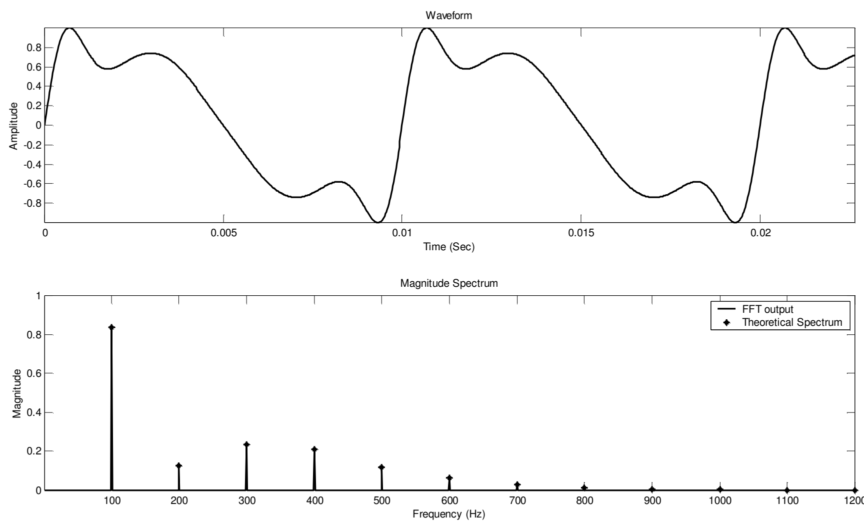


Figure 6: *The upper panel shows the waveform of the Complex FM signal where* $\omega_c = \omega_m = 2\pi 100$rad/s, **I**=[1 0.7 0.2], **φ**=[0 0 0] and θ=0. *The lower panel shows its spectrum computed using the FFT (solid line) and the program in Table (1) that compute the theoretical spectrum using Bessel Functions*

3.3 Discussion

The operation of this program is flexible with regard to the number of modulation terms. However, its implementation can be time consuming particularly with a large number of modulators. Another issue is in cases when the harmonic modulators have large modulation indices as this means that the signal bandwidth will be greater, and thus the evaluations must be carried out at a greater number of frequencies. If a distributed computing environment is available it would be useful to split the computation of (17) and (18) across several threads or processors. This is possible because the dependence of (17) and (18) on the variable τ means that the evaluation can be cut into shorter time segments with each machine processing an individual segment. The loop for computing the final spectral magnitudes could also be split, but the final compilation of these would need to be done altogether.

This program could also be vectorized for a serial machine if a matrix oriented programming environment such as Matlab is available. However, if the number of frequency components Ω is very high (greater than $1\text{x}10^6$) then there will be memory allocation issues and the program may not run. It is then necessary to implement this either on a sample by sample basis or to work with smaller vectors, store the results, and then combine them at the end of the evaluation.

4. Conclusion

This paper has offered an implementation for the equation for Complex Frequency Modulation where the modulation signal can have many harmonically related terms. It aimed to avoid the need for a large number of nested loops and to ensure that the program could cope with an arbitrary number of modulation terms. Finally, it suggested how it could be computed in a distributed manner if such an environment was available. This could significantly reduce the computation time. Along with making a complete open source implementation available, future work will include benchmarking tests for the program. This will be done to assess performance in relation to the number of modulators and the magnitudes of the modulation indices.

5. References

[1] J. Chowning, 'The Synthesis of Complex Audio Spectra by Means of Frequency Modulation', *Journal of the Audio Engineering Society*, Vol. 21, no. 7, Sept. 1973, pp. 525-534.

[2] J. Chowning and D. Bristow, *FM theory and applications – By Musicians for musicians*, Yamaha, Tokyo, Japan, 1986.

[3] M. Ishibashi, Electronic Musical Instrument, *U.S. Patent no. 4,658,691*, 1987.

[4] J. Timoney, V. Lazzarini, B. Carty and J. Pekonen, 'Phase and amplitude distortion methods for digital synthesis of classic analogue waveforms', *AES Convention 126*, Munich, Germany, May 7-10, 2009.

[5] FM8, Native Instruments, 2011,
http://www.native-instruments.com/#/en/products/producer/fm8/
[Accessed April 11th 2011]

[6] V. Lazzarini and J. Timoney, 'New Perspectives on Distortion Synthesis for Virtual Analogue Oscillators', *Computer Music Journal*, Vol. 34, No. 1, Mar. 2010, pp. 28-40.

[7] V. Lazzarini, J. Timoney and T. Lysaght, 'The generation of natural-synthetic spectra by means of adaptive frequency modulation', *Computer Music Journal*, Vol. 32, no. 2, summer 2008, pp. 9-22.

[8] K. Steiglitz, *A digital signal processing primer*, Prentice Hall, USA, Jan. 1996.

[9] B. Schottstaedt, 'The Simulation of Natural Instrument Tones Using Frequency Modulation with a Complex Modulating Wave', *Computer Music Journal*, Winter 1977, Vol. 1, no. 4, pp.46-50.

[10] B. Tan et al, 'Real-time implementation of double frequency modulation (DFM) synthesis', *Journal of Audio Engineering Society*, Vol. 42, no.11, Nov. 1994, pp. 918-926.

[11] B. Tan S. Lim, 'Automated parameter optimization for double frequency modulation synthesis using the genetic annealing algorithm', *Journal of Audio Engineering Society*, Vol. 44, no.1/2, Feb. 1996, 3-15.

[12] J. Timoney and V. Lazzarini, 'Exponential frequency modulation bandwidth criterion for virtual analog applications', submitted to *Digital Audio Effects (Dafx) 2011*, March 2011.

[13] J. Timoney, V. Lazzarini, J. Pekonen and V. Valimaki, 'Spectrally rich phase distortion sound synthesis using an allpass filter', *Proc. of IEEE ICASSP*, Taipei, Taiwan, April 2009, pp. 293-296.

[13] D. Benson, *Music: a mathematical offering*, Cambridge University Press, UK, Nov 2006.

[14] L. Giacoletto, 'Generalized theory of multitone amplitude and frequency modulation', *Proc. IRE*, Vol. 35, July 1947, pp. 680-693.

[15] M. LeBrun, 'A Derivation of the Spectrum of FM with a Complex Modulating Wave', *Computer Music Journal*, Vol. 1, no. 4, Winter 1977, pp.51-52.

[16] T. Mitchell and J. Sullivan, 'Frequency modulation tone matching using a fuzzy clustering evolution strategy', *AES Convention 118*, Barcelona, Spain, May 2005.

[17] H. Rowe, *Signals and noise in communications systems*, D. Van Nostrand, Princeton, NJ, USA, 1965.

[18] O. Esbach et al, *Esbach's handbook of engineering fundamentals*, 4th ed., Wiley, USA, 1990.

[19] R. Millane, 'Polynomial approximations to Bessel functions', *IEEE Trans. Ant. And Prop.*, June 2003, Vol. 51, no. 6, pp. 1398-1400.

[20] L. Li, 'A new polynomial approximation for J_v Bessel functions', Applied mathematics and computation, Vol. 183, no. 2, Dec. 2006, pp. 1220-1225.

[21] M. AbuelMa'atti, 'Trigonometric approximations for some Bessel functions', *Active and passive Elec. comp.*, Vol. 22, pp-75-85.