Audio Engineering Society

Convention Paper

Presented at the 120th Convention
2006 May 20–23 Paris, France

Signal Analysis Using the Complex Spectral Phase Evolution (CSPE) Method

Kevin M. Short[1] and Ricardo A. Garcia[2]

[1] Chaoticom Technologies, Andover, MA, 01810, USA
kevin @ chaoticom . com

[2] Chaoticom Technologies, Andover, MA, 01810, USA
rago @ chaoticom . com

ABSTRACT

The Complex Spectral Phase Evolution (CSPE) method is introduced as a tool to analyze and detect the presence of short-term stable sinusoidal components in an audio signal. The method provides for super-resolution of frequencies by examining the evolution of the phase of the complex signal spectrum over time-shifted windows. It is shown that this analysis, when applied to a sinusoidal signal component, allows for the resolution of the true signal frequency with orders of magnitude greater accuracy than the DFT. Further, this frequency estimate is independent of the frequency bin and can be estimated from "leakage" bins far from spectral peaks. The method is robust in the presence of noise or nearby signal components, and is a fundamental tool in the front-end processing for the KOZ compression technology.

1. INTRODUCTION

Many audio applications require signals to be represented as a discrete sum of sinusoidal components. In most transform-based processing techniques, audio is decomposed through the use of the Discrete Fourier transform (DFT) or the Fast Fourier transform (FFT) applied to windows of N data samples. While these transforms are one-to-one on discretely-sampled data, they suffer from several limitations that add to the complexity of the representation of the audio signal as a sum of sinusoidal components. One such characteristic is that the sinusoidal elements are all of integer periods over the N samples, whereas real signals have no such limitation. This causes the DFT and FFT to have a limited frequency resolution equal to the sampling rate divided by N. In the particular application of interest in this paper, a more efficient representation is desired for compressed audio.

In compression applications, the front end processing often requires modeling the sound as a decomposition into sinusoids plus transients plus noise [1][2]. Most of these methods depend on a mechanism for estimating a suitable set of parameters. Estimation of the parameters for sinusoidal components is usually done in the transformed frequency domain. Some approaches have used fixed frequency filterbanks to estimate the dominant frequencies in a short segment of data. Other attempts use methods such as parabolic interpolation between frequency bins in a short-time Fourier transform to estimate the frequency of the underlying tones. Most of these frequency-based methods are very dependent on the type of analysis window used to preprocess the data, and they suffer from the traditional frequency vs. time accuracy tradeoffs [3].

In this paper we will introduce a frequency domain technique that has proven capable of detecting, analyzing and modeling sinusoids with great accuracy in a short time frame. Additionally, the method proposed is computationally efficient and has been used in laboratory and real product situations, and provides the front-end analysis for the KOZ compression technology [4][5][6].

First we will show a generalized version of the CSPE method for complex sinusoids. Next, we will focus on real-valued signals and discuss some of the implications regarding analysis windows, high resolution analysis and frequency domain representations. Then we will follow a few numerical examples, one with real data from a CD, the other with laboratory data.

The last section will discuss other uses of the method at our research facility, further extensions and implications.

2. COMPLEX SPECTRAL PHASE EVOLUTION

The Complex Spectral Phase Evolution Method (CSPE) is a technique that allows for the detection of oscillatory components in the frequency spectrum of a signal and gives improved frequency resolution when compared to the resolution of a typical transform-based analysis [4]. The discussion below will assume that the calculations are done with the Discrete Fourier Transform (DFT) or the Fast Fourier Transform (FFT); however, other discrete and continuous transforms can be used as well.

The development here will focus on the discrete, sampled case, since sampled signals are the primary practical application in digital audio; however, the mathematics has an equivalent formulation in continuous integral transforms. The interested reader is referred to the references. The definition of CSPE is most easily seen in the complex domain, and the definition and examples in this section will assume that calculations are done in the complex domain. The presentation is designed to be accessible even if one is more familiar with analysis in a real or magnitude domain, and it is hoped that effort to gain comfort with the complex approach will be fruitful. Sections 2.2 and 2.3 will connect the complex sinusoids to real signals, so the links between the different formulations should be readily accessible.

2.1. CSPE for Complex Signals

Define a signal, **s**(t), and sampled version of the same signal, $\vec{\mathbf{s}} = (s_0, s_1, s_2, s_3, \ldots)$. If we take N samples of the signal, we can calculate the DFT of the signal by first defining the DFT matrix (and note that the FFT matrix is of the same form, with the restriction that N is equal to a power of 2).

Let $W = e^{-j\frac{2\pi}{N}}$ and then the matrix can be written in the simple form:

$$\mathbb{W} = \begin{bmatrix} 1 & 1 & 1 & 1 & \cdots & 1 \\ 1 & W & W^2 & W^3 & \cdots & W^{N-1} \\ 1 & W^2 & W^4 & W^6 & \cdots & W^{2(N-1)} \\ 1 & W^3 & W^6 & W^9 & \cdots & W^{3(N-1)} \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & W^{N-1} & W^{2(N-1)} & W^{3(N-1)} & \cdots & W^{(N-1)(N-1)} \end{bmatrix} \quad (1\ )$$

where each column of the matrix is a complex sinusoid oscillating an integer number of periods over the N-point sample window.

For a given block of N samples, define:

$$\vec{\mathbf{s}}_0 = \begin{bmatrix} s_0 \\ s_1 \\ s_2 \\ s_3 \\ \vdots \\ s_{N-1} \end{bmatrix}, \vec{\mathbf{s}}_1 = \begin{bmatrix} s_1 \\ s_2 \\ s_3 \\ s_4 \\ \vdots \\ s_N \end{bmatrix}, \cdots, \vec{\mathbf{s}}_i = \begin{bmatrix} s_i \\ s_{i+1} \\ s_{i+2} \\ s_{i+3} \\ \vdots \\ s_{i+N-1} \end{bmatrix}, \cdots \quad (2\ )$$

The DFT of the signal is then

$$F(\vec{\mathbf{s}}_i) =$$
$$\begin{bmatrix} 1 & 1 & 1 & 1 & \dots & 1 \\ 1 & W & W^2 & W^3 & \dots & W^{N-1} \\ 1 & W^2 & W^4 & W^6 & \dots & W^{2(N-1)} \\ 1 & W^3 & W^6 & W^9 & \dots & W^{3(N-1)} \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & W^{N-1} & W^{2(N-1)} & W^{3(N-1)} & \dots & W^{(N-1)(N-1)} \end{bmatrix} \begin{bmatrix} s_i \\ s_{i+1} \\ s_{i+2} \\ s_{i+3} \\ \vdots \\ s_{i+N-1} \end{bmatrix} \quad (3\ )$$

The goal of the CSPE is to analyze the phase evolution of the components of the signal between an initial sample of N points and a time-delayed sample of N points. Let the time delay be designated by Δ, and consider the product of $\boldsymbol{F}(\vec{\mathbf{s}}_i)$ and the complex conjugate of $\boldsymbol{F}(\vec{\mathbf{s}}_{i+\Delta})$. The CSPE is defined as the product (taken on an element-by-element basis) $CSPE = F(\vec{\mathbf{s}}_i) \odot F^*(\vec{\mathbf{s}}_{i+\Delta})$, along with an associated frequency, $f_{CSPE} = \measuredangle(F(\vec{\mathbf{s}}_i) \odot F^*(\vec{\mathbf{s}}_{i+\Delta}))$ where the operator $\measuredangle$ indicates that we take the angle of the complex entry resulting from the product. This frequency estimate will provide improved resolution over that which is inherent in the DFT.

To illustrate the method on sinusoidal data, take a signal of the form of a complex sinusoid that has period $p = q + \delta$, where q is an integer and δ is a fractional deviation of magnitude less than 1, i.e., $|\delta| \le 1$. The samples of the complex sinusoid can be written (the phase term just pulls out of the calculation, so it is set to zero here for simplicity):

$$\vec{\mathbf{s}}_0 = \begin{bmatrix} e^0 \\ e^{j2\pi\frac{(q+\delta)}{N}} \\ e^{j2\pi 2\frac{(q+\delta)}{N}} \\ e^{j2\pi 3\frac{(q+\delta)}{N}} \\ \vdots \\ e^{j2\pi(N-1)\frac{(q+\delta)}{N}} \end{bmatrix} \quad (4\ )$$

If we take a shift of one sample, then Δ=1 in the CSPE, and we have:

$$\vec{\mathbf{s}}_1 = \begin{bmatrix} e^{j2\pi\frac{(q+\delta)}{N}} \\ e^{j2\pi 2\frac{(q+\delta)}{N}} \\ e^{j2\pi 3\frac{(q+\delta)}{N}} \\ e^{j2\pi 2\frac{(q+\delta)}{N}} \\ \vdots \\ e^{j2\pi N\frac{(q+\delta)}{N}} \end{bmatrix} = e^{j2\pi\frac{(q+\delta)}{N}} \begin{bmatrix} e^0 \\ e^{j2\pi\frac{(q+\delta)}{N}} \\ e^{j2\pi 2\frac{(q+\delta)}{N}} \\ e^{j2\pi 3\frac{(q+\delta)}{N}} \\ \vdots \\ e^{j2\pi(N-1)\frac{(q+\delta)}{N}} \end{bmatrix} = e^{j2\pi\frac{(q+\delta)}{N}} \vec{\mathbf{s}}_0 \quad (5\ )$$

Then if we plug these into the conjugate product of the transforms, the result is

$$\begin{aligned} F(\vec{\mathbf{s}}_i) \odot F^*(\vec{\mathbf{s}}_{i+1}) &= e^{-j2\pi\frac{(q+\delta)}{N}} F(\vec{\mathbf{s}}_i) \odot F^*(\vec{\mathbf{s}}_i) \\ &= e^{-j2\pi\frac{(q+\delta)}{N}} \left\| F(\vec{\mathbf{s}}_i) \right\|^2 \end{aligned} \quad (6\ )$$

The CSPE can be used to find the frequency of the sinusoidal; it is found simply by taking the angle of the product above, and adjusting by the scaling factor $N/2\pi$, so we find that

$$\begin{aligned} f_{CSPE\,k} &= \frac{-N\measuredangle(F(\vec{\mathbf{s}}_0) \odot F^*(\vec{\mathbf{s}}_1))}{2\pi} = \frac{-N\left(-\dfrac{2\pi(q+\delta)}{N}\right)}{2\pi} \\ f_{CSPE\,k} &= (q+\delta) \end{aligned} \quad (7\ )$$

Note that if $\Delta > 1$, the scaling factor is readjusted accordingly. If this is compared to the information in the standard DFT calculation, the frequency bins are in integer multiples of $2\pi/N$, and so the CSPE calculation tells us that instead of the signal appearing at integer multiples of $2\pi/N$, it is actually at a fractional multiple given by $q + \delta$. An example of this for a "forward-spinning" or positive frequency complex sinusoid with $q + \delta = 116.78915, N = 512$, is given in the following figure:

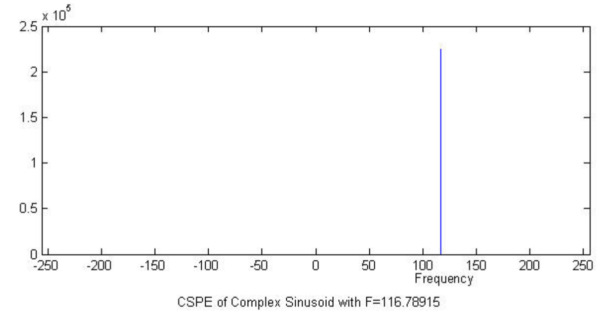

Figure 1 Positive Frequency complex sinusoid

Note that every frequency bin in the DFT spectrum has been reassigned to $q+\delta = 116.78915 \pm 1E^{-8}$. If we let the complex sinusoid be a "backward-spinning" or negative frequency sinusoid at $-(q+\delta)$, we get a similar result at the negative frequency:

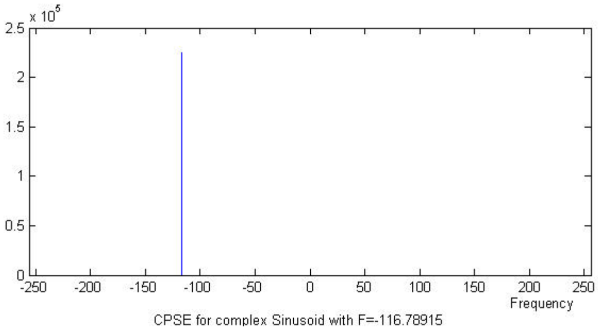

Figure 2 Negative Frequency complex sinusoid

Here all of the frequency bins have been reassigned to $q+\delta = -116.78915 \pm 1E^{-8}$. This improved frequency resolution is consistent even when we are dealing with real signals, since real sinusoids can be taken to be linear combinations of complex sinusoids, as shown in the next section.

It is important to note that the resulting reassignment of frequencies is independent of the frequency bin under consideration, so the CSPE allows one to determine the correct underlying frequency no matter what bin in the frequency domain is considered. If one were to look at the DFT of the same signal, the signal would have maximum power in frequency bin q-1, q, or q+1, and if $\delta \neq 0$, the signal power would leak to frequency bins well outside this range of bins. The CSPE, on the other hand, allows the power in the frequency bins of the DFT to be re-assigned to the correct underlying frequencies that produced the signal power, and this is true anywhere in the frequency spectrum.

2.2. CSPE for Real Signals

Consider a real sinusoid with period $p = q+\delta$ where p is an integer and δ is a fractional deviation of magnitude less than 1, i.e. $|\delta| \leq 1$, with amplitude "a" and arbitrary phase (chosen to be zero here for simplicity). The samples of a real sinusoid can be written as linear combinations of complex sinusoids, such as the following:

$$\vec{\mathbf{s}}_{0(n)} = \frac{a}{2} e^{j\frac{2\pi(q+\delta)}{N}n} + \frac{a}{2} e^{-j\frac{2\pi(q+\delta)}{N}n} \tag{8 }$$

and the one sample shift would be

$$\vec{\mathbf{s}}_{1(n)} = \frac{a}{2} e^{j\frac{2\pi(q+\delta)}{N}n} e^{j\frac{2\pi(q+\delta)}{N}} + \frac{a}{2} e^{-j\frac{2\pi(q+\delta)}{N}n} e^{-j\frac{2\pi(q+\delta)}{N}} \tag{9 }$$

if we define $D = e^{j\frac{2\pi(q+\delta)}{N}}$, the vectors could be written as:

$$\begin{aligned} \vec{\mathbf{s}}_{0(n)} &= \frac{a}{2} D^{n} + \frac{a}{2} D^{-n} \\ \vec{\mathbf{s}}_{1(n)} &= \frac{a}{2} D^{n} D + \frac{a}{2} D^{-n} D^{-1} \end{aligned} \tag{10 }$$

The DFT of each one of these vectors would then be:

$$\begin{aligned} F(\vec{\mathbf{s}}_0) &= F(\frac{a}{2} D^{n} + \frac{a}{2} D^{-n}) \\ F(\vec{\mathbf{s}}_0) &= \frac{a}{2} F(D^{n}) + \frac{a}{2} F(D^{-n}) \\ F(\vec{\mathbf{s}}_1) &= F(\frac{a}{2} D^{n} D + \frac{a}{2} D^{-n} D^{-1}) \\ F(\vec{\mathbf{s}}_1) &= \frac{a}{2} DF(D^{n}) + \frac{a}{2} D^{-1} F(D^{-n}) \end{aligned} \tag{11 }$$

The CSPE is computed using the complex product $F(\vec{\mathbf{s}}_0) \odot F^{*}(\vec{\mathbf{s}}_1)$ of the shifted and unshifted transforms, where the product operator $\odot$ is defined as the complex product taken element-by-element in the vector:

$$
\begin{aligned}
&F(\vec{\mathbf{s}}_0)F^*(\vec{\mathbf{s}}_1)\\
&=\left[\frac{a}{2}F(D^n)+\frac{a}{2}F(D^{-n})\right]\odot\left[\frac{a}{2}DF(D^n)+\frac{a}{2}D^{-1}F(D^{-n})\right]^*\\
&=\left(\frac{a}{2}\right)^2\left[F(D^n)+F(D^{-n})\right]\odot\left[D^*F^*(D^n)+DF^*(D^{-n})\right]
\end{aligned}
\tag{12 }
$$

Expanding the product:

$$
F(\vec{\mathbf{s}}_0)F^*(\vec{\mathbf{s}}_1)=\left(\frac{a}{2}\right)^2\begin{bmatrix} D^*F(D^n)\odot F^*(D^n)\\ +DF(D^n)\odot F^*(D^{-n})\\ +D^*F(D^{-n})\odot F^*(D^n)\\ +DF(D^{-n})\odot F^*(D^{-n})\end{bmatrix}
\tag{13 }
$$

Simplifying gives a form similar to that derived in the previous section:

$$
F(\vec{\mathbf{s}}_0)F^*(\vec{\mathbf{s}}_1)=\left(\frac{a}{2}\right)^2\begin{bmatrix} D^*\left\|F(D^n)\right\|^2\\ +DF(D^n)\odot F^*(D^{-n})\\ +D^*F(D^{-n})\odot F^*(D^n)\\ +D\left\|F(D^{-n})\right\|^2\end{bmatrix}
\tag{14 }
$$

This can be viewed as a sum of the CSPE for a “forward-spinning” or “positive-frequency” complex sinusoid and a “backward-spinning” or “negative-frequency” complex sinusoid, plus interaction terms. The first and the last terms in the sum are the same as the CSPE calculations in the previous section, but instead of a single complex sinusoid, there is a linear combination of two complex sinusoids, so the contributions to the CSPE from these two terms represent highly-concentrated peaks positioned at $q+\delta$ and $-(q+\delta)$, respectively. The middle two terms in Equation 14 are the interaction terms.

The interaction terms have some interesting properties, and can decrease the accuracy of the algorithm if not handled properly. We will see below and in the next section that the bias introduced by the interaction terms can be minimized by windowing the data. Note first that the interaction terms, Γ, can be simplified as follows:

$$
\begin{aligned}
\Gamma&=\left[DF(D^n)\odot F^*(D^{-n})+D^*F(D^{-n})\odot F^*(D^n)\right]\\
\Gamma&=\left[DF(D^n)\odot F^*(D^{-n})+\left(DF(D^n)\odot F^*(D^{-n})\right)^*\right]\\
\Gamma&=2\,\mathrm{Re}\left\{DF(D^n)\odot F^*(D^{-n})\right\}
\end{aligned}
\tag{15 }
$$

Remembering that $F(D^n)$ is a peak concentrated at frequency position $(q+\delta)$, and that $F(D^{-n})$ is a peak concentrated at frequency position $-(q+\delta)$, and that the product is taken on a element-by-element basis, it should be clear that $\Gamma\approx 0$ for most cases of interest. However, if the data is analyzed using an analysis window, it must be considered that the measured spectrum is found by convolving the true (delta-like) sinusoidal spectrum with the analysis window, so if a rectangular window (i.e., the boxcar window) is used, the leakage into nearby spectral bins is dramatic and is of sufficient strength to produce significant interaction terms and can even cause the $\left\|\bullet\right\|^2$ terms to interfere. The solution is to use a reasonable analysis window, so that the leakage is confined to the neighborhood of $(q+\delta)$ and $-(q+\delta)$, so the $\Gamma\approx 0$ case is the most common situation.

Now that the CSPE has been calculated, we can reassign the frequencies by extracting the angle information as in the previous section. For the positive frequencies (k>0), we find:

$$
\begin{aligned}
f_{CSPE\,k}&=\frac{-N\measuredangle\left(F_k(\vec{\mathbf{s}}_0)F_k^*(\vec{\mathbf{s}}_1)\right)}{2\pi}\\
&=\frac{-N\measuredangle\left(\left(\frac{a}{2}\right)^2\left\|F_k(D^n)\right\|^2e^{-j\frac{2\pi(q+\delta)}{N}}\right)}{2\pi}\\
&=\frac{-N\left(-\frac{2\pi(q+\delta)}{N}\right)}{2\pi}\\
f_{CSPE\,k}&=(q+\delta)
\end{aligned}
\tag{16 }
$$

and for the negative frequencies (k<0), we get the opposite value, $f_{CSPE\,k}=-(q+\delta)$, as expected. Consequently, in the case of real signals (for $\Gamma\approx 0$), all of the power in the positive frequencies is remapped to

$(q+\delta)$ and all of the power in the negative frequencies is remapped to $-(q+\delta)$. This result is independent of the frequency bin, and allows for extremely accurate estimates of frequencies.

2.3. CSPE for Real Windowed Signals

This section introduces CSPE for real sinusoids that have been windowed with an analysis window.

The developments in the last section can be generalized to include the effects of windowing by defining the basic transform to be a windowed transform. In this section, we will begin with the application of the previous results to windowed transforms, and then we will proceed to develop the windowing effects in more detail.

For the case where data is windowed before computing the DFT, we can define an arbitrary analysis window, A(t), and its sampled version A$_n$. The transforms are then defined as before, just with pre-multiplying by the analysis window:

$$F(\vec{\mathbf{s}}_0) \Rightarrow F(\vec{A}\odot\vec{\mathbf{s}}_0) \equiv F_W(\vec{\mathbf{s}}_0) \qquad (17\,)$$

where the W subscript indicates that we are taking the windowed transform.

The previous results carry over for the windowed transform, and we get the same form of the result even in the presence of windowing:

$$F_W(\vec{\mathbf{s}}_0)F_W^{\,*}(\vec{\mathbf{s}}_1) = \left(\frac{a}{2}\right)^2 \begin{bmatrix} D^*\left\|F_W(D^n)\right\|^2 \\ +2\,\mathrm{Re}\left\{DF_W(D^n)\odot F_W^{\,*}(D^{-n})\right\} \\ +D\left\|F_W(D^{-n})\right\|^2 \end{bmatrix} \qquad (18\,)$$

As long as the analysis window is chosen to decay rapidly the leakage into nearby frequency bins is minimized, and in this form it is evident that the interference terms are effectively negligible in most cases.

While the mathematics necessary is all contained in the previous equation, it is easier to see explicitly the effect of windowing if this expression is examined further. For explanatory purposes, we will assume that the signal goes through $(q+\delta)$ cycles in N samples, with amplitude a and initial phase b. Also, imagine the signal can be extended as necessary for K samples, where $K = N\gamma$, and γ is any integer number. This will change the number of cycles of our sinusoidal to be $(q+\delta)\gamma = \beta$. For illustrative purposes, assume that the parameter γ is chosen so that β will be an integer. The new signal will have then β full cycles over K samples.

$$\begin{aligned} \vec{\mathbf{s}}_{0(n)} &= a\cos\left(\frac{2\pi\beta}{K}n+b\right) \\ \vec{\mathbf{s}}_{0(n)} &= \frac{a}{2}e^{j\left(\frac{2\pi\beta}{K}n+b\right)}+\frac{a}{2}e^{-j\left(\frac{2\pi\beta}{K}n+b\right)} \\ \vec{\mathbf{s}}_{0(n)} &= \frac{a}{2}e^{jb}e^{j\frac{2\pi\beta}{K}n}+\frac{a}{2}e^{-jb}e^{-j\frac{2\pi\beta}{K}n} \end{aligned} \qquad (19\,)$$

Similarly, for a 1 sample delayed signal $\vec{\mathbf{s}}_1$:

$$\begin{aligned} \vec{\mathbf{s}}_{1(n)} &= a\cos\left(\frac{2\pi\beta}{K}(n+1)+b\right) \\ \vec{\mathbf{s}}_{1(n)} &= \frac{a}{2}e^{j\left(\frac{2\pi\beta}{K}(n+1)+b\right)}+\frac{a}{2}e^{-j\left(\frac{2\pi\beta}{K}(n+1)+b\right)} \\ \vec{\mathbf{s}}_{1(n)} &= \frac{a}{2}e^{jb}e^{j\frac{2\pi\beta}{K}n}e^{j\frac{2\pi\beta}{K}}+\frac{a}{2}e^{-jb}e^{-j\frac{2\pi\beta}{K}n}e^{-j\frac{2\pi\beta}{K}} \end{aligned} \qquad (20\,)$$

Figure 3 shows the time domain vectors $\vec{\mathbf{s}}_0$ and $\vec{\mathbf{s}}_1$:

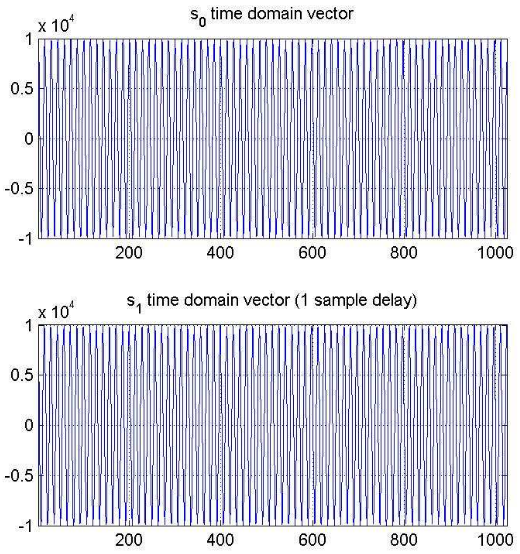

Figure 3 $\vec{\mathbf{s}}_{0(n)}$ and $\vec{\mathbf{s}}_{1(n)}$ vectors in time domain, with $N = 128, \gamma = 8, \beta = 72$

Figure 4 shows plots of the K-point DFTs.

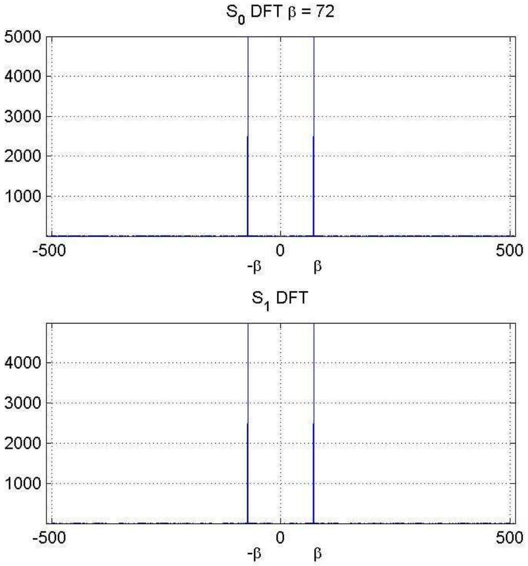

Figure 4 $\|F(\vec{\mathbf{s}}_0)\|$ and $\|F(\vec{\mathbf{s}}_1)\|$ vectors in frequency domain, with $N = 128, \gamma = 8, \beta = 72$. Only two spectral lines at β and $-\beta$.

Since the DFT of an integer-period sinusoid is in the center of a frequency bin, and the rectangular window (implicitly used) produces no leakage into other bins in this case, the resulting DFT elements have energy only in bins β and $-\beta$.

NOTE: In what follows, particularly in the definition of the convolutions, all the bin numbers and frequencies are assumed to be in standard "wrap around" order.

If we define $D = e^{j\frac{2\pi\beta}{K}}$, the n^{th} element of vectors vectors $\vec{\mathbf{s}}_0$ and $\vec{\mathbf{s}}_1$: could be written as:

$$\begin{aligned} \vec{\mathbf{s}}_{0(n)} &= \frac{a}{2} e^{jb} \vec{D}^n + \frac{a}{2} e^{-jb} \vec{D}^{-n} \\ \vec{\mathbf{s}}_{1(n)} &= \frac{a}{2} e^{jb} \vec{D}^n D + \frac{a}{2} e^{-jb} \vec{D}^{-n} D^{-1} \end{aligned} \qquad (21\,)$$

Where $\vec{D}^n$ is the complex vector $\left(D^0, D^1, D^2, \ldots\right)$. For the reminder of this section, we will drop the vector symbol on $\vec{D}^n$ and $\vec{D}^{-n}$ since it is obvious when the vector is needed.

Now, an analysis window $\vec{A}_{(n)}$ is introduced.

$$\vec{A}_{(n)} = \begin{cases} \text{non-zero} & 0 < n < N-1 \\ 0 & \text{otherwise} \end{cases} \qquad (22\,)$$

Some common options for analysis window are Hamming, Hanning, and Rectangular windows (it can be shown that when no windowing is applied, the rectangular window is nonetheless present). In this development, the only important issue is that the analysis window is known and computable.

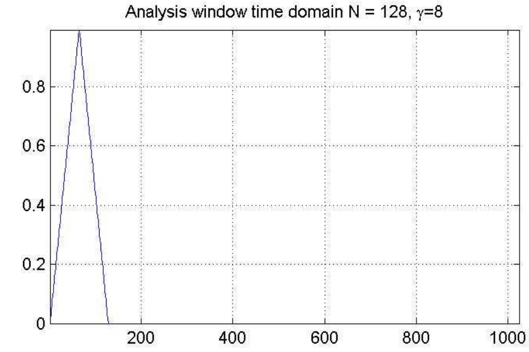

Figure 5 Analysis window $\vec{A}_{(n)}$ $N = 128$ samples are non-zero (Bartlett window)

And the frequency response of this analysis window looks like

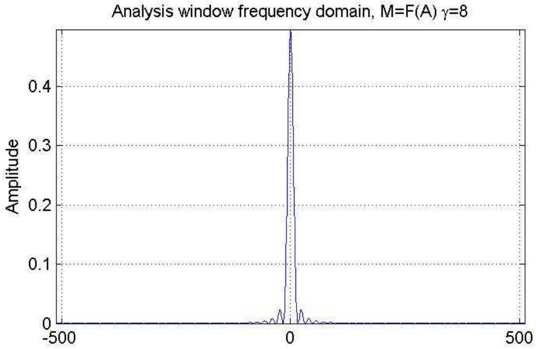

Figure 6 Analysis window frequency response magnitude $\left\|F(\vec{A})\right\|$

The vectors $\vec{\mathbf{s}}_0$ and $\vec{\mathbf{s}}_1$ can be windowed by multiplying sample by sample with the analysis window $\vec{A}_{(n)}$.

$$\begin{aligned} \vec{\mathbf{s}}_{w0(n)} &= \vec{\mathbf{s}}_{0(n)} \odot \vec{A}_{(n)} \\ \vec{\mathbf{s}}_{w1(n)} &= \vec{\mathbf{s}}_{1(n)} \odot \vec{A}_{(n)} \end{aligned} \qquad (23\ )$$

The windowed signals look like:

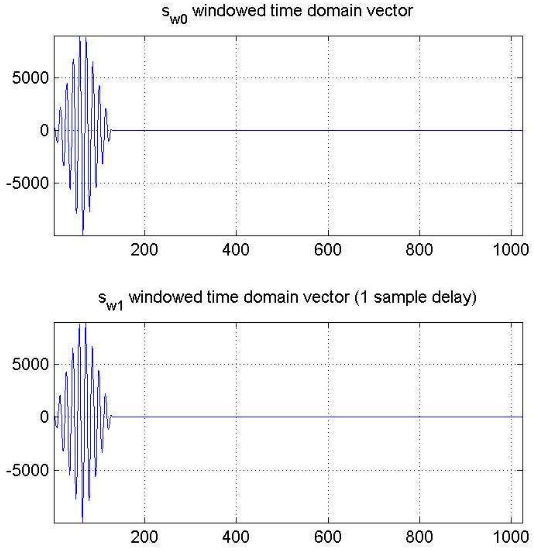

Figure 7 $\vec{\mathbf{s}}_{w0(n)}$ and $\vec{\mathbf{s}}_{w1(n)}$ windowed vectors in time domain

The multiplication of two signals in the time domain is seen as the circular convolution ($*$) of their frequency responses:

$$\begin{aligned} F(\vec{\mathbf{s}}_{w0}) &= F(\vec{\mathbf{s}}_0 \odot \vec{A}) = F(\vec{\mathbf{s}}_0) * F(\vec{A}) \\ F(\vec{\mathbf{s}}_{w1}) &= F(\vec{\mathbf{s}}_1 \odot \vec{A}) = F(\vec{\mathbf{s}}_1) * F(\vec{A}) \end{aligned} \qquad (24\ )$$

We define $M = F(\vec{A})$, to denote the frequency response vector of analysis window $\vec{A}_{(n)}$.

$$\begin{aligned} F(\vec{\mathbf{s}}_{w0}) &= F(\vec{\mathbf{s}}_0) * M \\ &= F\left(\frac{a}{2}e^{jb}D^n + \frac{a}{2}e^{-jb}D^{-n}\right) * M \\ &= \left[\frac{a}{2}e^{jb}F(D^n) + \frac{a}{2}e^{-jb}F(D^{-n})\right] * M \\ F(\vec{\mathbf{s}}_{w0}) &= \frac{a}{2}e^{jb}\left(F(D^n) * M\right) + \frac{a}{2}e^{-jb}\left(F(D^{-n}) * M\right) \end{aligned} \qquad (25\ )$$

and similarly for $F(\vec{\mathbf{s}}_{w1})$

$$\begin{aligned} F(\vec{\mathbf{s}}_{w1}) &= F(\vec{\mathbf{s}}_1) * M \\ &= F\left(\frac{a}{2}e^{jb}D^nD + \frac{a}{2}e^{-jb}D^{-n}D^{-1}\right) * M \\ &= \left[\frac{a}{2}e^{jb}DF(D^n) + \frac{a}{2}e^{-jb}D^{-1}F(D^{-n})\right] * M \end{aligned}$$

$$F(\vec{\mathbf{s}}_{w1}) = \frac{a}{2}e^{jb}D\big(F(D^n) * M\big) + \frac{a}{2}e^{-jb}D^{-1}\left(F(D^{-n}) * M\right) \tag{26}$$

This can be seen to produce replicas of analysis windows shifted to β and $-\beta$, as observed in Figure 8.

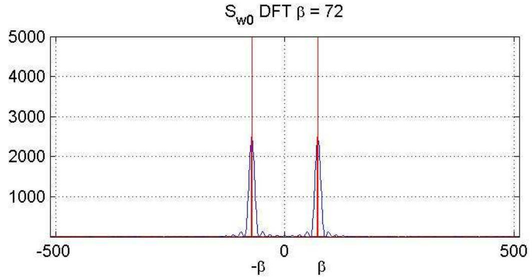

Figure 8 $\left\|F(\vec{\mathbf{s}}_{w0})\right\| = \left\|F(\vec{\mathbf{s}}_0) * F(\vec{A})\right\|$, superimposed in $\left\|F(\vec{\mathbf{s}}_0)\right\|$

The actual value of $F(\vec{\mathbf{s}}_{w0})$ for a given frequency bin k needs to be computed and then we will see the value of the convolution $\left(F(D^n) * M\right)$ for the frequency bin k.

The circular convolution $\left(F(D^n) * M\right)$ at bin k is then (recall that bin numbers wrap around):

$$\left(F(D^n) * M\right)\Big|_k = \sum_{l=0}^{K-1} F_l(D^n)M_{(k-l)} \tag{27}$$

If we define the DFT matrix as in section 2.1, then $W = e^{-j\frac{2\pi}{K}}$. For any given bin m we have:

$$\begin{aligned} F_m(D^n) &= \frac{1}{K}\sum_{n=0}^{K-1}\left(D^nW^{mn}\right) \\ &= \frac{1}{K}\sum_{n=0}^{K-1}\left(e^{j\frac{2\pi\beta}{K}n}e^{-j\frac{2\pi}{K}mn}\right) \\ &= \frac{1}{K}\sum_{n=0}^{K-1}\left(e^{j\frac{2\pi}{K}n(\beta-m)}\right) \end{aligned} \tag{28}$$

This sum is zero for all the possible values of m except when $m = \beta$. This makes $F_\beta(D^n) = 1$, a delta function at frequency position β (this is nothing more than the orthogonality property of complex integer-period sinusoids). This result can be used to simplify equation 27 to:

$$\begin{aligned} \left(F(D^n) * M\right)\Big|_k &= F_\beta(D^n)M_{(k-\beta)} \\ &= M_{(k-\beta)} \end{aligned} \tag{29}$$

Similarly, the circular convolution $\left(F(D^{-n}) * M\right)$ for the frequency bin k is defined by:

$$\left(F(D^{-n}) * M\right)\Big|_k = \sum_{l=0}^{K-1} F_l(D^{-n})M_{(k-l)} \tag{30}$$

For any given bin m we have:

$$\begin{aligned} F_m(D^{-n}) &= \frac{1}{K}\sum_{n=0}^{K-1}\left(D^{-n}W^{mn}\right) \\ &= \frac{1}{K}\sum_{n=0}^{K-1}\left(e^{-j\frac{2\pi\beta}{K}n}e^{-j\frac{2\pi}{K}mn}\right) \\ &= \frac{1}{K}\sum_{n=0}^{K-1}\left(e^{-j\frac{2\pi}{K}n(\beta+m)}\right) \end{aligned} \tag{31}$$

This sum is zero for all the possible values of m except when $m = -\beta$. This makes $F_\beta(D^{-n}) = 1$. This result can be used to simplify equation 30 to:

$$\begin{aligned} \left(F(D^{-n}) * M\right)\Big|_k &= F_\beta(D^{-n})M_{(k+\beta)} \\ &= M_{(k+\beta)} \end{aligned} \tag{32}$$

Equation 25 for bin k is then simplified to:

$$
\begin{aligned}
F_k(\vec{\mathbf{s}}_{w0}) &= \frac{a}{2}e^{jb}\left(F_k(D^n)*M\right)+\frac{a}{2}e^{-jb}\left(F_k(D^{-n})*M\right)\\
&= \frac{a}{2}e^{jb}M_{(k-\beta)}+\frac{a}{2}e^{-jb}M_{(k+\beta)}
\end{aligned}
\qquad (33\ )
$$

Similarly, for $F_k(\vec{\mathbf{s}}_{w1})$ we have:

$$
\begin{aligned}
F_k(\vec{\mathbf{s}}_{w1}) &= \frac{a}{2}e^{jb}D\left(F_k(D^n)*M\right)+\frac{a}{2}e^{-jb}D^{-1}\left(F_k(D^{-n})*M\right)\\
&= \frac{a}{2}e^{jb}DM_{(k-\beta)}+\frac{a}{2}e^{-jb}D^{-1}M_{(k+\beta)}
\end{aligned}
\qquad (34\ )
$$

The CSPE for bin k is then:

$$
\begin{aligned}
&CSPE_k = F_k(\vec{\mathbf{s}}_{w0})F_k^*(\vec{\mathbf{s}}_{w1})\\
&= \left[\frac{a}{2}e^{jb}M_{(k-\beta)}+\frac{a}{2}e^{-jb}M_{(k+\beta)}\right]\left[\frac{a}{2}e^{jb}DM_{(k-\beta)}+\frac{a}{2}e^{-jb}D^{-1}M_{(k+\beta)}\right]^*\\
&= \left[\frac{a}{2}e^{jb}M_{(k-\beta)}+\frac{a}{2}e^{-jb}M_{(k+\beta)}\right]\left[\frac{a}{2}e^{-jb}D^{-1}M^*{}_{(k-\beta)}+\frac{a}{2}e^{jb}DM^*{}_{(k+\beta)}\right]
\end{aligned}
\qquad (35\ )
$$

Expanding the terms:

$$
CSPE_k = \begin{bmatrix}
\left(\frac{a}{2}\right)^2 D^{-1}M_{(k-\beta)}M^*{}_{(k-\beta)} + \left(\frac{a}{2}\right)^2 e^{j2b}DM_{(k-\beta)}M^*{}_{(k+\beta)}\\
+\left(\frac{a}{2}\right)^2 e^{-j2b}D^{-1}M^*{}_{(k-\beta)}M_{(k+\beta)} + \left(\frac{a}{2}\right)^2 DM_{(k+\beta)}M^*{}_{(k+\beta)}
\end{bmatrix}
$$

$$
CSPE_k = \begin{bmatrix}
\left(\frac{a}{2}\right)^2 D^{-1}\left\|M_{(k-\beta)}\right\|^2\\
+2\,\mathrm{Re}\left\{\left(\frac{a}{2}\right)^2 e^{j2b}DM_{(k-\beta)}M^*{}_{(k+\beta)}\right\}\\
+\left(\frac{a}{2}\right)^2 D\left\|M_{(k+\beta)}\right\|^2
\end{bmatrix}
\qquad (36\ )
$$

Equation 36 is of the same form that Equation 18. In the typical case where the magnitude of the analysis window at $(k+\beta)$ is very close to zero $\left\|M_{(k+\beta)}\right\| \approx 0$, the last three terms of Equation 36 can be ignored.

$$
\begin{aligned}
CSPE_k &= F_k(\vec{\mathbf{s}}_{w0})F_k^*(\vec{\mathbf{s}}_{w1})\\
CSPE_k &\approx \left(\frac{a}{2}\right)^2 D^{-1}M_{(k-\beta)}M^*{}_{(k-\beta)}\\
CSPE_k &\approx \left(\frac{a}{2}\right)^2 D^{-1}\left\|M_{(k-\beta)}\right\|^2
\end{aligned}
\qquad (37\ )
$$

2.3.1. Frequency Estimation

The CSPE for bin k can be used to compute an estimate for the frequency of the underlying real sinusoidal.

$$
\begin{aligned}
f_{CSPE(k)} &= \frac{-N\measuredangle(CSPE_k)}{2\pi} = \frac{-N\measuredangle(F_k(\vec{\mathbf{s}}w_0)F_k^*(\vec{\mathbf{s}}w_1))}{2\pi}\\
&= \frac{-N\measuredangle\left(\left(\frac{a}{2}\right)^2 D^{-1}\left\|M_{(k-\beta)}\right\|^2\right)}{2\pi}\\
&= \frac{-N\measuredangle\left(\left(\frac{a}{2}\right)^2 e^{-j\frac{2\pi\beta}{K}}\left\|M_{(k-\beta)}\right\|^2\right)}{2\pi}\\
&= \frac{-N\left(-\frac{2\pi\beta}{K}\right)}{2\pi} = \frac{-N\left(-\frac{2\pi(q+\delta)\gamma}{N\gamma}\right)}{2\pi}\\
f_{CSPE(k)} &= (q+\delta)
\end{aligned}
\qquad (38\ )
$$

Note that the selection of γ has no influence on the precision of the frequency estimate, so it was introduced primarily for illustrative purposes. A selection of $\gamma = 1$ would make $K = N$ and still be a valid selection. This is to be expected since the CSPE method holds for any sinusoidal frequency.

2.3.2. Magnitude and Phase Remapping

If the same assumptions hold, i.e., that the frequency response of the analysis window falls off sufficiently quickly, we can use equation 33 and simplify it like:

$$
\begin{aligned}
F_k(\vec{\mathbf{s}}_{w0}) &= \frac{a}{2}e^{jb}M_{(k-\beta)}+\frac{a}{2}e^{-jb}M_{(k+\beta)}\\
F_k(\vec{\mathbf{s}}_{w0}) &\approx \frac{a}{2}e^{jb}M_{(k-\beta)}
\end{aligned}
\qquad (39\ )
$$

and use our estimated value for the frequency of the tone to be $f_{CSPE(k)}$:

$$ae^{jb} = \frac{2F_k(\vec{\mathbf{s}}_{w0})}{M_{(k-f_{CSPE(k)})}} \qquad (40)$$

Equation 40 allows us to estimate all of the parameters needed to recreate the underlying sinusoidal signal. To do so, we choose k to be one of the frequency bins near frequency position $q+\delta$, since that gives the best signal-to-noise ratio. The numerator in Equation 40 is a complex value that can be read off from the DFT spectrum. The denominator is also known, since M comes from the chosen analysis window, and we have calculated f_{CSPE}. Then the left hand side of Equation 40 gives us an accurate estimate of the magnitude and phase of the real sinusoidal signal.

The net effect is that when reconstructing the signal component, all that is necessary is to take a representation of the analysis window, M, shift it to frequency f_{CSPE}, rotate it by angle e^{jb}, multiply it by magnitude a and resample it onto the grid of frequency bins in the DFT.

3. EXAMPLES

While the discussion above focuses on the application of the CSPE super-resolution algorithm to a single signal component, it works much more generally than that, and can be used to resolve many signals components so long as there is some separation between the signal frequencies. When multiple signals are present, the super-resolution of the frequencies is most accurate near spectral frequency bins that are dominated by an individual signal component, and the regions of the spectrum that are away from the signal centers are generally remapped to the nearest dominant signal frequency.

An example of this process is given for a signal composed of three sinusoids. The signals do not lie in the center of frequency bins, but the algorithm successfully recalculates the true underlying frequencies with good accuracy. Figure 9 gives a graphical representation of this process. The original FFT and remapped spectrums are shown; the remapped spectrum is effectively a line spectrum. For this example, the exact frequencies (in frequency bin numbers) are 28.7965317, 51.3764239, and 65.56498312, while the estimated frequencies are 28.7960955, 51.3771794, and 65.5644420. If these spectra were calculated from music sampled at CD sampling rates of 44100samples/sec, the resolution of each freq bin would be approximately 21.53Hz/bin, so the measured signals are accurate to approximately ± 0.001bins, which is equivalent to ± 0.02153Hz.

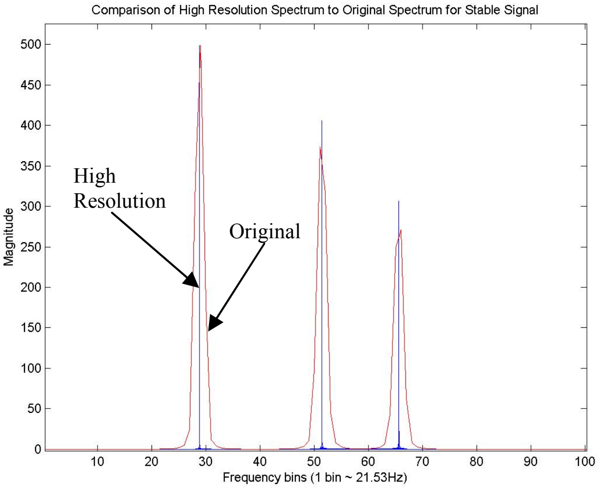

Figure 9 Comparison of original spectrum to high resolution CSPE for stable signal

In real-world music the data is not as clean and stable, and the accuracy of the computed high-resolution spectrum is affected by the presence of nearby signals that interfere, modulations of the frequencies, and noise-like signals that have a broadband spectrum. Even so, in these situations, the high-resolution analysis generally gives signal accuracy on the order of 0.1Hz for any signal component that is relatively stable over the sample window. An example is given for a window of data taken from a track by Norah Jones and the remapped spectrum appears in Figure 10, where the original signal is the dotted line the remapped signal is the solid line.

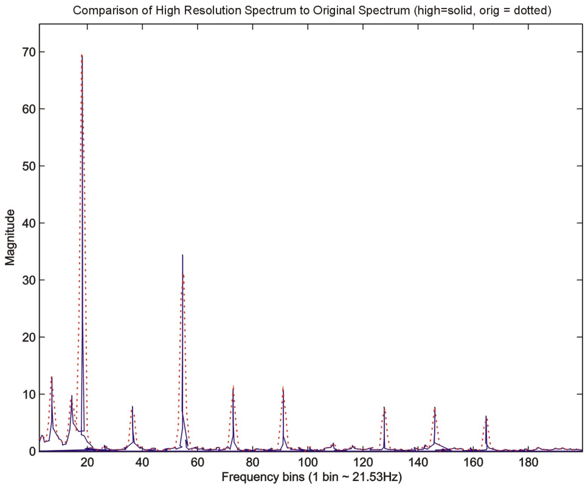

Figure 10 Comparison of original spectrum to high resolution CSPE for music segment

The frequency values are used to find the correct magnitude values and the correct angular rotations. When all of the pieces are reassembled, the spectrum of the reconstructed signal is given in Figure 11. The differences between the original and the reconstructed spectrum are nearly invisible to the eye, and are psychoacustically imperceptible.

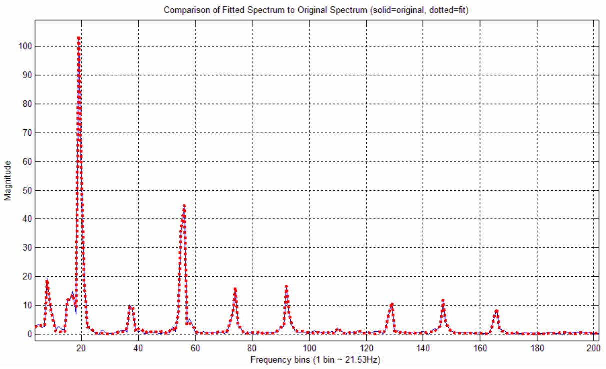

Figure 11 Comparison of resynthesized (fitted) spectrum to original

4. DISCUSSION AND CONCLUSION

The techniques described in this paper provide a practical way to do super-resolution of frequencies by studying the phase evolution of signal components. The ability to do accurate signal estimation has been a valuable tool in developing the KOZ compressed music format, but has also found application in analysis of a wide range of signals from machine vibrations to image and video data.

The CSPE method is the first in a family of signal processing techniques that are built up from this mathematical framework. Variations on these techniques have been developed to handle a number of other cases:

- To provide accurate estimates of low frequency signals where the interference terms are non-negligible.
- Separation and detection of overlapping signal components.
- To resolve linearly modulating signals onto single spectral lines at the root frequency along with the modulation rate parameter.
- To resolve groups of modulating harmonics where the modulation is coupled.
- Detection and estimation of the frequency of signals above the Nyquist frequency. (Aliased component estimation)

The CSPE techniques require no more than a standard discrete transform, but by conducting the analysis in the complex domain rather than the magnitude domain, the phase evolution information has proven to be extremely valuable for applications.

5. REFERENCES

[1] Serra, X. 1989. A System for Sound Analysis/Transformation/Synthesis based on a Deterministic plus Stochastic Decomposition. Ph.D. Dissertation, Stanford University.

[2] T. F. Quatieri and R. J. McAulay, Speech Analysis/Synthesis Based on a Sinusoidal Representation. Technical Report 693, Lincoln Laboratory, M. I. T., 1985

[3] M. Desainte-Catherine and S. Marchand. “High-Precision Fourier Analysis of Sounds Using Signal Derivatives”, Journal of the Audio Engineering Society, Vol 48, No 7/8, July/August 2000

[4] D. J. Nelson and K. M. Short. "A channelized cross spectral method for improved frequency resolution." Proceedings of the IEEE-SP International Symposium on Time-Frequency and Time-Scale Analysis," IEEE Press, October 1998

[5] Short et al, "Method and apparatus for compressed chaotic music synthesis", United States Patent 6,137,045; October 2000 (end patent pending)

[6] A. T. Parker and K. M. Short, "Method and apparatus for secure digital chaotic communication", United States Patent 6,363,153; March 2002 (end patent pending)

[7] Alan V. Oppenheim , Ronald W. Schafer , John R. Buck, Discrete-time signal processing (2nd ed.), Prentice-Hall, Inc., Upper Saddle River, NJ, 1999