On time localization of sinusoids from the DFT of their sum

M. Vinyes
Technical Report, Music Technology Group (Pompeu Fabra University)

June 20, 2006

Abstract

This document discusses techniques that allow to estimate the time localization of a set of summed sinusoids that are analyzed using the DFT. Time localization means knowing the beginning and end of the sinusoid within the input time frame to the DFT. Although those results could be used in other contexts, this work tries to solve the problem for audio signals, which have rich spectrum (a great number of harmonics coexist). In previous literature, this report would have used the terms onset detection or time reassignment but here it rather focuses on the low level part of the problem. Detailed proofs are provided so that the reader could read them fast. Latex, matlab and maple sources are provided, and the reader is encouraged to contribute improvements or extract results as long as the same philosophy (Free Documentation License+GNU Public License) is applied.

Contents

1 Introduction

1.1 Notation and problem statement

$N \in \mathbb{N}$ samples are extracted from a digital signal in order to perform the DFT with optional prewindowing. N is supposed to be a power of 2 in order to use the FFT algorithm.

We will name discrete signals with lower case letters and name their discrete independent variable n (eg. $x[n]$).

We will refer to their Fourier Transforms with their corresponding capital letters and to their continuous independent variable with w (eg. $X(w)$).

Finally their Discrete Fourier Transforms will be written as well with their corresponding capital letters but with a discrete independent variable k (eg. $X[k]$). Hence:

Let $x[n]$ be a N samples frame from a signal (i.e. $x[n] = 0 \; \forall n < 0$ and $\forall n > N-1$)

$$X(w) \equiv A(w)e^{j\phi(w)} \equiv FT\{x[n]\}(w) \equiv \sum_{n=-\infty}^{\infty} x[n]e^{-jwn} = \sum_{n=0}^{N-1} x[n]e^{-jwn}$$

$$X[k] \equiv A[k]e^{j\phi[k]} \equiv DFT\{x[n]\}[k] \equiv \sum_{n=0}^{N-1} x[n]e^{-j\frac{2\pi k}{N}n} = X(w)\big|_{w=\frac{2\pi k}{N}}$$

$h[n]$ be a window of N samples, symmetric around $(N-1)/2$ (i.e. $h[k] = h[N-1-k]$)

$x_h[n] \equiv x[n]h[n]$ be the windowed signal frame

$$X_h[k] \equiv A_h[k]e^{j\phi_h[k]} = DFT\{x[n]h[n]\}[k]$$

$$\Pi_{B,E}[n] \equiv \begin{cases} 1, & \text{if } n \in B..E \\ 0, & \text{otherwise} \end{cases}$$

Now, lets suppose that $x[n]$ is a sum of discrete sinusoids with normalized frequency $f_k \in [0,1]$, phase Φ_k and amplitude A_k beginning in sample B_k and ending in sample E_k.

$$x[n] = \sum_{k=1}^{M} A_k \cdot cos(2\pi f_k n + \Phi_k) \cdot \Pi_{B_k,E_k}[n]$$

The goal of this document is to present methods that estimate B_k and E_k from $X_h[k]$ without knowing the original values of M,f_k, A_k and Φ_k.

1.2 Motivations

The problem isn't formulated in terms of "onsets and "time reassignment" as it has been usually done in previous literature for various reasons.

It is difficult to define what do we mean by a musical onset. Most musicians would agree about identifying musical onsets in some piece of musics but the mathematical characterization remains unclear. Some approaches consider energy variation or phase discontinuities, however this document doesn't intend to understand those perceptual and psychological concepts, it will only focus on signal processing tools that might be useful in the future for this task.

The same problem exists for time reassignment. Time reassignment consists on assigning to the DFT coefficient the time where the signal associated to the DFT coefficient begins. This might have some sense for the detection of single tones, however in music, the notion of a signal that groups certain DFT coefficients is ambiguous.

Finally it must be said that both terms only refer to the beginning of a signal within the analysis frame. On the other hand, signals that begin at the same sample of the analysis frame and end at different samples will have very different DFT coefficients and properties. Hence onset detection and time reassignment techniques that forget about the end of this signal within the frame, are not considering the whole problem and won't work in all situations.

1.3 Frequency analysis of x[n]

Our solution will be based on the frequency domain, hence, a deep study about the FT and the DFT of our signal will help us to understand the limitations of each method. Later references to this section should be expected.

First the DFT is written in terms of the FT of the signal. That is because the frequency-continuous Fourier Transform is easier to operate than the DFT and allows us to see the influence of the buffer size N.

Let $*$ indicate a "discrete" convolution, i.e. $FT\{x[n]*y[n]\} = \frac{1}{2\pi}\int_{-\pi}^{\pi} X(\lambda)Y(w-\lambda)d\lambda$

$$X_h[k] = DFT\{x[n]h[n]\}[k] = [FT\{x[n]h[n]\}(w)]_{w=\frac{2\pi k}{N}}$$

Now the compute the Fourier transform of x[n] is computed,

$$\begin{aligned}
&FT\{x[n]\}(w) \\
&= FT\{A_k \cdot cos(2\pi f_k n + \Phi_k)\Pi_{B_k,E_k}[n]\}(w) \\
&= FT\{A_k \cdot cos(2\pi f_k n + \Phi_k)\}(w) * FT\{\Pi_{B_k,E_k}[n]\}(w) \\
&= \frac{A_k}{2}\sum_{m=-\infty}^{\infty}(e^{j\Phi_k}\delta(w-2\pi f_k+2\pi m) + e^{-j\Phi_k}\delta(w+2\pi f_k+2\pi m)) * e^{-j\frac{B_k+E_k}{2}w}\frac{sin(\frac{E_k-B_k}{2}w)}{sin(\frac{1}{2}w)} \\
&= \frac{A_k}{2}(e^{-j(\frac{B_k+E_k}{2}w-2\pi f_k-\Phi_k)}\frac{sin(\frac{E_k-B_k}{2}(w-2\pi f_k))}{sin(\frac{1}{2}w)} + e^{-j(\frac{B_k+E_k}{2}w+2\pi f_k+\Phi_k)}\frac{sin(\frac{E_k-B_k}{2}(w+2\pi f_k))}{sin(\frac{1}{2}w)})
\end{aligned}$$

For an unwindowed x[n] ($h[n]$ is simply rectangular window) $x[n]h[n] = x[n]$, hence,

$$X_h[k] = \left[\frac{A_k}{2}(e^{-j(\frac{B_k+E_k}{2}w-2\pi f_k-\Phi_k)}\frac{sin(\frac{E_k-B_k}{2}(w-2\pi f_k))}{sin(\frac{1}{2}w)} + e^{-j(\frac{B_k+E_k}{2}w+2\pi f_k+\Phi_k)}\frac{sin(\frac{E_k-B_k}{2}(w+2\pi f_k))}{sin(\frac{1}{2}w)})\right]_{w=\frac{2\pi k}{N}} \quad (1)$$

This expression already makes us think that, when $h[n]$ is a rectangular window, the phase derivative of the Fourier Transform may be used to estimate the center of the sinusoid support ($\frac{B_k+E_k}{2}$). In fact, if both sinc's are considered to decrease fast to 0, and that in one of the main lobes there's a majority contribution of one of them, we obtain that:

$$\frac{\mathrm{d}Phase(X)}{\mathrm{d}w} = \frac{\mathrm{d}}{\mathrm{d}w}\left(\frac{B_k+E_k}{2}w - 2\pi f_k - \Phi_k\right) = \frac{B_k+E_k}{2}$$

In next chapter a method is presented to numerically compute the derivative from two DFTs of x[n]. However some simulations reveal that this expression is still quite unstable depending of the proximity of the other main lobe, and its phase shift. That is why, in chapter 3 a more robust estimator (that depends on various values of this derivative around the main lobe) is explained. Finally, also in chapter 3, we will see that, still with $h[n]$ a rectangular window, it is possible to estimate the length of the sinusoid's support by taking into account its energy computed in the frequency domain.

Otherwise, for a windowed x[n] we've got the following expression:

$$X_h[k] = \left[\frac{A_k}{2}(e^{j\Phi_k}Y(w-2\pi f_k) + e^{-j\Phi_k}Y(w+2\pi f_k))\right]_{w=\frac{2\pi k}{N}} \quad (2)$$

Where

$$Y(w) = e^{-j(\frac{B_k+E_k}{2}w+2\pi f_k+\Phi_k)}\frac{sin(\frac{E_k-B_k}{2}(w+2\pi f_k))}{sin(\frac{1}{2}w)} * H(w) \quad (3)$$

And the $H(w)$ of two popular windows would be:

$$\text{Let } D(w) = e^{j\frac{1}{2}w}\frac{sin(\frac{N}{2}w)}{sin(\frac{1}{2}w)}$$

$$H_{hanning}(w) = 0.5D(w) + 0.25\left(D(w-\frac{2\pi}{N}) + D(w+\frac{2\pi}{N})\right)$$

$$= 0.5e^{-j\frac{N-1}{2}w}\left(\frac{sin(\frac{N}{2}w)}{sin(\frac{1}{2}w)} + 0.5e^{-j\frac{\pi}{N}}\frac{sin(\frac{N}{2}(w-\frac{2\pi}{N}))}{sin(\frac{1}{2}(w-\frac{2\pi}{N})} + 0.5e^{j\frac{\pi}{N}}\frac{sin(\frac{N}{2}(w+\frac{2\pi}{N}))}{sin(\frac{1}{2}(w+\frac{2\pi}{N})}\right)$$

$$H_{blackman-harris(92)}(w) = .35875D(w) + .48829\left(D(w-\frac{2\pi}{N}) + D(w+\frac{2\pi}{N})\right)$$

$$+ .14128\left(D(w-2\frac{2\pi}{N}) + D(w+2\frac{2\pi}{N})\right) + .01168\left(D(w-3\frac{2\pi}{N}) + D(w+3\frac{2\pi}{N})\right)$$

Taking into account that the above expression may be too complex, we won't develop new methods for this case, but we will try to bring the problem to the unwindowed case. More details are given in section 3.4.

1.4 Does a N/2 shift help?

In problem statement, windows $h[n]$ have been chosen symmetric around (N-1)/2, i.e. symmetrical in a pair N samples buffer. Therefore, those windows have a Fourier Transform $H(w) = e^{-j\frac{N-1}{2}w}HR(w)$ where $HR(w)$ is real. We can get rid of most of the phase w dependent shift by shifting in the time domain $x_h[n]$ by $N/2$ samples (i.e. $x_h(shifted)[n] = x_h[n - N/2]$ in a circular buffer). That way,

$$X_h(shifted)(w) = X(shifted)(w)e^{+j\frac{N}{2}w}e^{-j\frac{N-1}{2}w}HR(w) = X(shifted)(w)e^{-j\frac{1}{2}w}HR(w)$$

And, because usually a non time-localized $x(shifted)[n]$ looks like the original $x[n]$ (when it is a sinusoid or whatever periodic signal with period significantly smaller than the buffer size) the phase picture has less variation: $e^{-j\frac{1}{2}w}$. However this doesn't seem to be very helpful for our problem. We have seen that time localization, modeled as a pulse, contributes also a phase shift and it may have a stronger impact on the $X_h(w)$ spectrum than the window in much cases. Hence, as we don't know where this pulse will be centered, a $N/2$ doesn't seem to solve anything because this pulse may also contribute a big w dependent phase shift.

2 Preliminary results

2.1 Parseval

Extracted from [PG96]. However, a more specific proof is given here.

Lemma 2.1 (Parseval)**.**

$$\boxed{\sum_{n=0}^{N-1} x[n]^2 = \frac{1}{N}\sum_{k=0}^{N-1} |X[k]|^2}$$

Proof.

$$\sum_{n=0}^{N-1} x[n]^2 = \sum_{n=0}^{N-1} x[n]x^*[n] = \sum_{n=0}^{N-1} x[n]\frac{1}{N}\sum_{k=0}^{N-1} X^*[k]e^{-j\frac{2\pi k}{N}n} = \frac{1}{N}\sum_{n=0}^{N-1}\sum_{k=0}^{N-1} x[n]X^*[k]e^{-j\frac{2\pi k}{N}n}$$
$$= \sum_{k=0}^{N-1} X^*[k]\sum_{n=0}^{N-1} x[n]e^{-j\frac{2\pi k}{N}n} = \frac{1}{N}\sum_{k=0}^{N-1} X^*[k]X[k] = \frac{1}{N}\sum_{k=0}^{N-1} X[k]^2$$

□

2.2 Time-scaled Parseval

The name "Time-scaled Parseval" will be used to refer to this result. This result was extracted from [WM03].

Lemma 2.2 (Time-scaled Parseval)**.**

$$\boxed{\sum_{n=0}^{N-1} nx[n]^2 = -\frac{1}{N}\sum_{k=0}^{N-1} |X[k]|^2 \frac{\mathrm{d}\phi}{\mathrm{d}w}[k]}$$

Proof.

$$\begin{aligned}\sum_{n=0}^{N-1} nx[n]^2 &= \sum_{n=0}^{N-1} nx[n]x^*[n] \\ &= \sum_{n=0}^{N-1} nx[n]\frac{1}{N}\sum_{k=0}^{N-1} X^*[k]e^{-j\frac{2\pi k}{N}n} \\ &= \frac{1}{N}\sum_{n=0}^{N-1}\sum_{k=0}^{N-1} n\cdot x[n]X^*[k]e^{-j\frac{2\pi k}{N}n} \\ &= \frac{1}{N}\sum_{k=0}^{N-1} X^*[k]\sum_{n=0}^{N-1} nx[n]e^{-j\frac{2\pi k}{N}n}\end{aligned}$$

$$\frac{\mathrm{d}X(w)}{\mathrm{d}w} = \frac{\mathrm{d}A(w)}{\mathrm{d}w}e^{j\phi(w)} + A(w)e^{j\phi(w)}j\frac{\mathrm{d}\phi(w)}{\mathrm{d}w}$$

$$
\begin{aligned}
\sum_{n=0}^{N-1} nx[n]^2 &= \sum_{k=0}^{N-1} X^*[k] \sum_{n=0}^{N-1} nx[n]e^{-j\frac{2\pi k}{N}n} \\
&= \frac{1}{N}\sum_{k=0}^{N-1} A[k]e^{-j\phi[k]} j\frac{\mathrm{d}X}{\mathrm{d}w}[k] \\
&= \frac{1}{N}\sum_{k=0}^{N-1} jA[k]\frac{\mathrm{d}A}{\mathrm{d}w}[k] - A[k]^2\frac{\mathrm{d}\phi}{\mathrm{d}w}[k] \\
&= \frac{1}{N}\sum_{k=0}^{N-1} -A[k]^2\frac{\mathrm{d}\phi}{\mathrm{d}w}[k]
\end{aligned}
$$

The last step is based on the fact that the original result is real, so the imaginary part must cancel out.

□

2.3 Computing phase and amplitude frequency derivatives with the DFT

Next we'll see more results from [WM03].

Lemma 2.3 (DFT phase derivative)**.**

$$
\boxed{\frac{\mathrm{d}\phi}{\mathrm{d}w}[k] = -\Re\left(\frac{DFT\{nx[n]\}[k]}{X[k]}\right)}
$$

Proof.

$$
\begin{aligned}
\frac{\mathrm{d}\phi}{\mathrm{d}w}[k] &= \frac{\mathrm{d}}{\mathrm{d}w}\Im\left(\log(A[k]) + j\phi[k]\right) \\
&= \frac{\mathrm{d}}{\mathrm{d}w}\Im\left(\log(X[k])\right) \\
&= \Im\left(\frac{1}{X[k]}\frac{\mathrm{d}}{\mathrm{d}w}\sum_{n=-\infty}^{\infty} x[n]e^{jwn}\right) \\
&= \Im\left(\frac{1}{X[k]}(-j)\sum_{n=-\infty}^{\infty} nx[n]e^{jwn}\right) \\
&= -\Re\left(\frac{DFT\{nx[n]\}[k]}{X[k]}\right)
\end{aligned}
$$

□

Lemma 2.4 (DFT amplitude derivative)**.**

$$
\boxed{\frac{\mathrm{d}A}{\mathrm{d}w}[k] = A[k]\Im\left(\frac{DFT\{nx[n]\}[k]}{X[k]}\right)}
$$

Proof.

$$\begin{aligned}
\frac{\mathrm{d}A}{\mathrm{d}w}[k] &= A[k]\frac{\mathrm{d}}{\mathrm{d}w}\log(A[k]) \\
&= A[k]\frac{\mathrm{d}}{\mathrm{d}w}\Re\left(\log(X[k])\right) \\
&= A[k]\Re\left(\frac{1}{X[k]}\frac{\mathrm{d}}{\mathrm{d}w}\sum_{n=-\infty}^{\infty} x[n]e^{jwn}\right) \\
&= A[k]\Re\left(\frac{1}{X[k]}(-j)\sum_{n=-\infty}^{\infty} nx[n]e^{jwn}\right) \\
&= A[k]\Im\left(\frac{DFT\{nx[n]\}[k]}{X[k]}\right)
\end{aligned}$$

□

3 Problem Solution

3.1 Road map

In this section we will first look at the solution of our problem for $M = 1$, i.e. for an input signal:

$$x[n] = A_k \cdot cos(2\pi f_k n + \Phi_k) \cdot \Pi_{B_k,E_k}[n]$$

This solution doesn't make a lot of sense when there's only one sinusoid, because B_k and E_k could perfectly be estimated temporally. However, those results are developed with the aim to generalize them in for finite sums of sinusoids, that is the DFT decomposition of any real signal.

So first two ways are presented to estimate, from the DFT coefficients of an unwindowed signal (i.e. $h[n]$ is a rectangular window), the center of the sinusoid's support and its length. Combining both, the values of B_k and E_k may be obtained. The error of these estimations is also analyzed and bounded.

Next the effect of windowing the signal is discussed, and a method is suggested to "unwindow" the sinusoid DFT in order to use the previously explained methods.

Finally we study what happens when M sinusoids coexist:

$$x[n] = \sum_{k=1}^{M} A_k \cdot cos(2\pi f_k n + \Phi_k) \cdot \Pi_{B_k,E_k}[n]$$

We will see that the above methods can be still used to localize each one in time.

3.2 Estimation of the sinusoid's support length in the frequency domain

Lemma 3.1. *Let* $0 < f_{min} < f < f_{max} < 0.5$*, and a* $N \in \mathbb{N}$ *samples sinusoid be:*

$$x[n] = A_k \cdot cos(2\pi f_k n + \Phi_k) \cdot \Pi_{B_k,E_k}[n]$$

with known A_k*. Then* $E_k - B_k$ *can be estimated as follows:*

$$\boxed{E_k - B_k = -1 + \frac{2}{NA_k^2} \sum_{k=0}^{N-1} |X[k]|^2 \pm \varepsilon}$$

with

$$|\varepsilon| < 1 + \frac{1}{tan(2\pi \min(f_{min}, 0.5 - f_{max}))}$$

Proof. By lemma 2.1 (Parseval),

$$\frac{1}{N} \sum_{k=0}^{N-1} |X[k]|^2 = \sum_{n=0}^{N-1} x[n]^2$$

The finite sum is computed by grouping it into geometric series:

$$\sum_{n=0}^{N-1} x[n]^2 = A_k^2 \sum_{n=B_k}^{E_k} cos(2\pi f_k + \Phi_k)^2$$
$$= \frac{A_k^2}{2} \sum_{n=B_k}^{E_k} (1 + cos(4\pi f_k + 2\Phi_k))$$
$$= \frac{A_k^2}{2} \left(E_k - B_k + 1 + \frac{1}{2} \sum_{n=B_k}^{E_k} e^{j(4\pi f_k n + 2\Phi_k)} + \frac{1}{2} \sum_{n=B_k}^{E_k} e^{-j(4\pi f_k n + 2\Phi_k)} \right)$$
$$= \frac{A_k^2}{2} \left(E_k - B_k + 1 + \frac{1}{2} \frac{e^{j(4\pi f_k B_k + 2\Phi_k)} + e^{j(4\pi f_k (E_k+1) + 2\Phi_k)} + e^{-j(4\pi f_k (B_k-1) + 2\Phi_k)} + e^{-j(4\pi f_k E_k + 2\Phi_k)}}{e^{j4\pi f_k} - 1} \right)$$
$$= \frac{A_k^2}{2} \left(E_k - B_k + 1 + \frac{e^{j2\pi f_k}}{e^{j4\pi f_k} - 1} (sin(4\pi f_k (E_k + 0.5) + 2\phi_k) - sin(4\pi f_k (B_k - 0.5) + 2\phi_k)) \right)$$
$$= \frac{A_k^2}{2} \left(E_k - B_k + 1 + \frac{sin(4\pi f_k (E_k + 0.5) + 2\phi_k) - sin(4\pi f_k (B_k - 0.5) + 2\phi_k)}{2sin(2\pi f_k)} \right)$$
$$= \frac{A_k^2}{2} \left(E_k - B_k + 1 + \frac{cos(4\pi f_k E_k + 2\Phi) + cos(4\pi f_k B_k + 2\Phi_k)}{2} + \frac{sin(4\pi f_k E_k + 2\phi_k) - sin(4\pi f_k B_k + 2\phi_k)}{2tan(2\pi f_k)} \right)$$

Hence,

$$E_k - B_k = -1 + \frac{2}{NA_k^2} \sum_{k=0}^{N-1} |X[k]|^2 - \frac{cos(4\pi f_k E_k + 2\phi_k) + cos(4\pi f_k B_k + 2\phi_k)}{2} - \frac{sin(4\pi f_k E_k + 2\phi_k) - sin(4\pi f_k B_k + 2\phi_k)}{2tan(2\pi f_k)}$$

The first error term can be bounded easily:

$$\left| \frac{cos(4\pi f_k E_k + 2\Phi) + cos(4\pi f_k B_k + 2\Phi_k)}{2} \right| < 1$$

The second one is unstable in frequencies around 0 or 0.5 (Nyquist) where $tan(2\pi f_k)$ is 0 valued. Therefore it can be only bounded if we know the minimum and maximum f that will be considered. Let f_{min} the minimum and f_{max} the maximum. Figure 1 illustrates this result.

$$\frac{sin(4\pi f_k E_k + 2\phi_k) - sin(4\pi f_k B_k + 2\phi_k)}{2tan(2\pi f_k)} < 1 + \frac{1}{tan(2\pi \min(f_{min}, 0.5 - f_{max}))}$$

□

Theorem 3.2. *Let* $0 < f_{min} < f < f_{max} < 0.5$*, and a* $N \in \mathbb{N}$ *samples sinusoid be:*

$$x[n] = A_k \cdot cos(2\pi f_k n + \Phi_k) \cdot \Pi_{B_k, E_k}[n]$$

with ***unknown*** A_k*.*
Let $|X_{max}| = \max |FT\{x[n]\}(w)|$*. Then* $E_k - B_k$ *can be estimated as follows:*

$$\boxed{E_k - B_k = \frac{2N|X_{max}|^2}{\sum_{k=0}^{N-1} |X[k]|^2} \pm \alpha}$$

with

$$|\alpha| < 2 + \frac{1}{tan(2\pi \min(f_{min}, 0.5 - f_{max}))}$$

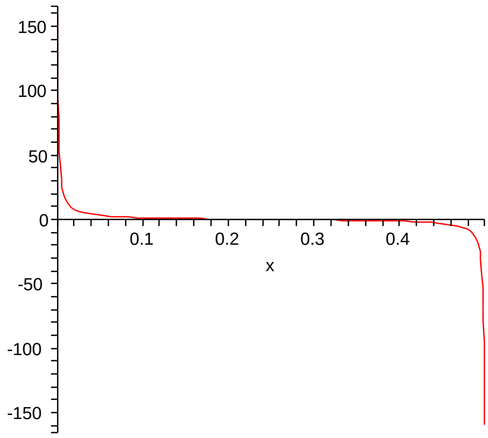


Figure 1: Graph of $\frac{1}{tan(2\pi f_k)}$ in the interval $[0, 0.5]$

This theorem gives us a practical way to estimate $E_k - B_k$. $|X_{max}|$ can be estimated from $DFT\{x[n]\}(w)$, which is a sampled version of $FT\{x[n]\}(w)$. However the reader should bear in mind that **this estimation is very sensitive to the error made in the measurement of the true** $|X_{max}|$, specially for big N, where the main lobe is narrower. So **as much zero padding as possible should be done** or at least some interpolation should be implemented to estimate the maximum of the sinc's main lobe.

An example may help to understand the importance this point: in an experiment without windowing, $N = 2048$, no zero padding and a single sinusoid of $\phi_k = \frac{\pi}{2}$, $f_k = \frac{1.3}{256}, \frac{E_k+B_k}{2} - N = 199$, $E_k - B_k = 1648$ I got an error of 467 samples instead of the theoretical maximum error of 33 samples.

If this issue is taken into account, this theorem works reasonably well (small ε) when dealing with signals that have bounded frequencies. The 0.5 bound is not a problem, because Nyquist theorem tells us that there's aliasing for greater frequencies. And the other bound is not dramatic: let us consider $x[n]$ is a musical sinusoid sampled at 44100Hz. Let $f_{min} = \frac{150}{44100}, f_{max} = \frac{21900}{44100}$, then the length of the sinusoid support can be estimated blindly with an error of $|\varepsilon| = 49$ samples (1ms).

Proof. Looking at formula 1 ($FT\{x[n]\}(w)$), we can see that its maximum corresponds to the center of the main lobe of the sinc and it has magnitude $|X_{max}| = \frac{A_k}{2}(E_k - B_k)$. Hence we can obtain A_k as follows: $A_k = \frac{2}{E_k - B_k}|X_{max}|$.

Let $L := E_k - B_k$ for easier manipulation. If we put this together with the result of the previous lemma (3.1), we obtain:

$$L = -1 + \frac{2}{N\frac{4}{L^2}|X_{max}|^2} \sum_{k=0}^{N-1} |X[k]|^2 \pm \varepsilon$$

$$\left(\frac{\sum_{k=0}^{N-1} |X[k]|^2}{2N|X_{max}|^2} \right) L^2 - L - 1 \pm \varepsilon$$

Solving this equation, we get:

$$L = \frac{+1 + \sqrt{1 + 2(1 \pm \varepsilon)\left(\frac{\sum_{k=0}^{N-1}|X[k]|^2}{N|X_{max}|^2}\right)}}{\left(\frac{\sum_{k=0}^{N-1}|X[k]|^2}{N|X_{max}|^2}\right)} = \frac{2N|X_{max}|^2}{\sum_{k=0}^{N-1}|X[k]|^2} + \alpha$$

On the other hand,

$$\text{Let } z = \frac{\sum_{k=0}^{N-1}|X[k]|^2}{N|X_{max}|^2} \qquad \alpha(z) = \frac{\sqrt{1+2(1+\varepsilon)z} - 1}{z}$$

Applying Hï¿½pital's rule, we get: $\lim_{z \to 0} \alpha(z) = \varepsilon + 1$.

$$\begin{aligned}\frac{\mathrm{d}\alpha}{\mathrm{d}z} &= \frac{\sqrt{1+2(1+\varepsilon)z} - (1 + (1+\varepsilon)z)}{x^2\sqrt{1+2(1+\varepsilon)z}} \\ &= \frac{\sqrt{2(0.5 + (1+\varepsilon)z)} - (0.5 + (1+\varepsilon)z) - 0.5}{x^2\sqrt{1+2(1+\varepsilon)z}}\end{aligned}$$

Solving, $\sqrt{2}m - m^2 - 0.5 = 0$ we obtain $m = \frac{\sqrt{2}}{2}$ so the above derivative is negative when $0.5 + (1+\varepsilon)z > \frac{\sqrt{2}}{2}^2 = 0.5$ which is always true because $z > 0$.

Hence, $\alpha(z)$ is decreasing for positive values of z and its maximum is achieved in $z = 0$. In other words, $|\alpha| < \varepsilon + 1$. And from the previous lemma (3.1) we know that:

$$|\varepsilon| < 1 + 1 + \frac{1}{tan(2\pi \min(f_{min}, 0.5 - f_{max}))}$$

□

3.3 Estimation of the center of a sinusoid's support in the frequency domain

This result was extracted from [Röb03].

The continuous centroid was first analyzed because I thought the discrete case would come as a particular case, after considering the error between the continuous and the discrete centroid. However, this proved to be much more difficult than directly computing the discrete centroid and bound the error. In other words, the discretization of the centroid does affect significantly its estimation of the center of a sinusoid's support. In particular, when the frequency f of the sinusoid is near 0 or 0.5 (Nyquist).

Hence, the following result is useless for our final purposes, but I decided to keep it in this report because it may be useful to compare it with its discrete analog.

Lemma 3.3 (Continuous centroid)**.** *Let a continuous sinusoid be:*

$$x(t) = A_k \cdot cos(2\pi f_k t + \Phi_k) \cdot \Pi_{B_k, E_k}[t]$$

with $E_k - B_k \geq \frac{1}{4f_k}$*, then*

$$\boxed{\frac{\int_{-0.5}^{N-0.5} t x(t)^2 dt}{\int_{-0.5}^{N-0.5} x(t)^2 dt} = \frac{B_k + E_k}{2} \pm \varepsilon}$$

with

$$|\varepsilon| < \frac{1}{4f_k}$$

The centroid will give exactly the center of the support of $x[n]$ as long as $x[n]$ is symmetrical around the center of its support. However, the signal may be asymmetrical depending on B_k and E_k and make the value differ from the center of the support. This variation is relatively (to $1/f_k$) small and maybe considered as an error ε that is bounded next.

Proof.

$$\begin{aligned}
\varepsilon :&= \frac{\int_{-0.5}^{N-0.5} tx(t)^2 dt}{\int_{-0.5}^{N-0.5} x(t)^2 dt} - \frac{B_k + E_k}{2} \\
&= \frac{\int_{B_k}^{E_k} tx(t)^2 dt}{\int_{B_k}^{E_k} x(t)^2 dt} - \frac{B_k + E_k}{2} \\
&= \frac{\int_{B_k}^{E_k} (t - \frac{B_k+E_k}{2}) x(t)^2 dt}{\int_{B_k}^{E_k} x(t)^2 dt} \\
&= \frac{\int_{B_k - \frac{B_k+E_k}{2}}^{E_k - \frac{B_k+E_k}{2}} sx(s + \frac{B_k+E_k}{2})^2 ds}{\int_{B_k - \frac{B_k+E_k}{2}}^{E_k - \frac{B_k+E_k}{2}} x(s + \frac{B_k+E_k}{2})^2 ds} \\
&= \frac{\int_{B_k - \frac{B_k+E_k}{2}}^{E_k - \frac{B_k+E_k}{2}} s\cos(2\pi f_k s + 2\pi f_k \frac{B_k+E_k}{2} + \Phi_k)^2 ds}{\int_{B_k - \frac{B_k+E_k}{2}}^{E_k - \frac{B_k+E_k}{2}} \cos(2\pi f_k s + 2\pi f_k \frac{B_k+E_k}{2} + \Phi_k)^2 ds}
\end{aligned}$$

Let $\Delta C := \frac{E_k - B_k}{2}$, $\Omega := 2\pi f_k \frac{B_k+E_k}{2} + \Phi_k$,

$$\varepsilon = \frac{\int_{-\Delta C}^{\Delta C} s\cos(2\pi f_k s + \Omega)^2 ds}{\int_{-\Delta C}^{\Delta C} \cos(2\pi f_k s + \Omega)^2 ds}$$

Let $\theta = 4\pi f_k \Delta C$,

$$\begin{aligned}
\int_{-\Delta C}^{\Delta C} s\cos(2\pi f_k s + \Omega)^2 ds &= \left[\frac{2\pi f_k t sin(4\pi f_k t + 2\Omega) + cos(2\pi f_k t + \Omega)^2 + 4f_k^2\pi^2 t^2}{16 f_k^2 \pi^2}\right]_{-\Delta C}^{\Delta C} \\
&= \frac{1}{16 f_k^2 \pi^2}(2\pi f_k \Delta C sin(4\pi f_k \Delta C + 2\Omega) - 2\pi f_k(-\Delta C) sin(-4\pi f_k \Delta C + 2\Omega) \\
&\quad + cos(2\pi f_k \Delta C + \Omega)^2 - cos(-2\pi f_k \Delta C + \Omega)^2) \\
&= \frac{1}{32 f_k^2 \pi^2}(\theta sin(\theta + 2\Omega) + \theta sin(\theta - 2\Omega) + cos(\theta + 2\Omega) - cos(\theta - 2\Omega)) \\
&= \frac{1}{16 f_k^2 \pi^2}(\theta cos(2\Omega) - sin(2\Omega)) sin(\theta)
\end{aligned}$$

$$\begin{aligned}\int_{-\Delta C}^{\Delta C} \cos(2\pi f_k s + \Omega)^2 ds &= \left[\frac{1}{2}t + \frac{1}{8\pi f_k} sin(4\pi f_k t + 2\Omega)\right]_{-\Delta C}^{\Delta C} \\ &= \Delta C + \frac{1}{8\pi f_k} sin(4\pi \Delta C + 2\Omega) + \frac{1}{8\pi f_k} sin(4\pi f_k \Delta C - 2\Omega) \\ &= \frac{1}{4\pi f_k}(\theta + cos(2\Omega) sin(\theta))\end{aligned}$$

Therefore, let $\theta = 4\pi f_k \Delta C$, $x = cos(2\Omega) = cos(4\pi f_k \frac{B_k + E_k}{2} + 2\Phi_k)$,

$$\varepsilon = \frac{1}{4f_k}\left(\frac{1}{\pi}\frac{(\theta x - \sqrt{1-x^2}) sin(\theta)}{\theta + x sin(\theta)}\right)$$

with $x \in [-1, 1]$ and $\theta \in (0, 2\pi max(f_k)N]$
Now, if we let $\theta > \theta_0$ with $\theta_0 > 1$, it can be seen that $|\theta x - \sqrt{1-x^2}| \leq \theta$ for $x \in [-1, 1]$. So,

$$\begin{aligned}|f(\theta, x)| &= \left|\frac{1}{\pi}\frac{(\theta x - \sqrt{1-x^2}) sin(\theta)}{\theta + x sin(\theta)}\right| \\ &\leq \frac{1}{\pi}\frac{\theta}{\theta - 1} \\ &\leq \frac{1}{\pi}\frac{1}{1 - 1/\theta_0}\end{aligned}$$

And if we force $|f(\theta, x)| \leq 1$, we obtain, $\theta_0 = \frac{1}{1-1/\pi} \approx 1.47$

On the other hand, this is a natural condition that is verified if we set $\Delta C \geq \frac{1}{8f_k}$ (equivalent to $E_k - B_k \leq \frac{1}{4f_k}$).
Finally, it can be said that this bound is correct as it is almost reached by $\theta = \theta_0, x = -1$:

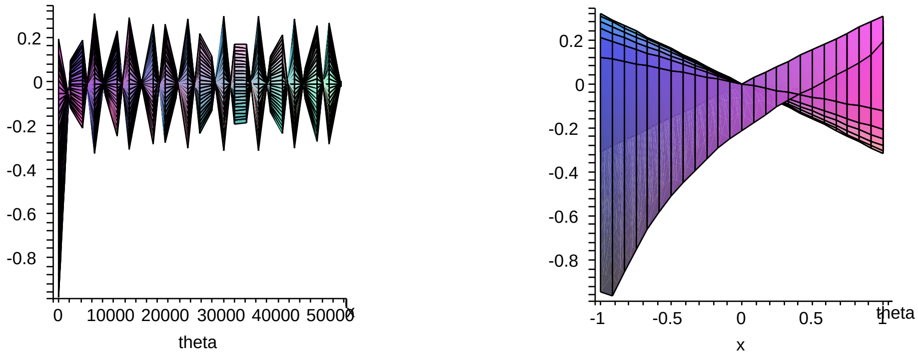
$$\begin{aligned}|f(\theta, x)| &= \left|\frac{1}{\pi}\frac{(\theta x - \sqrt{1-x^2}) sin(\theta)}{\theta + x sin(\theta)}\right| \\ &\leq \frac{1}{\pi}\frac{-\theta_0 sin(\theta_o)}{\theta - sin(\theta_0)} \\ &\approx 0.98\end{aligned}$$

But if we consider $\theta >> \theta_0$, some simplifications may be made to $f(\theta, x)$ in order to obtain an approximate bound of $f(\theta, x) < \frac{1}{\pi}$.

$$\begin{aligned}|f(\theta, x)| &= \left|\frac{1}{\pi}\frac{(\theta x - \sqrt{1-x^2}) sin(\theta)}{\theta + x sin(\theta)}\right| \\ &\approx \left|\frac{1}{\pi}\frac{(\theta x) sin(\theta)}{\theta}\right| \\ &\leq \frac{1}{\pi}\end{aligned}$$

As it can be seen in the graphs 2(a), 2(a) this bound better corresponds with "most probable" error that the user will see when doing several experiments with random values of B_k, E_k, f_k, ϕ_k.

□



(a) θ (b) x

Figure 2: $f(\theta, x)$ projected to the θ axis or the x axis, with $\theta > 1.5$.

Lemma 3.4 (Discrete centroid)**.** *Let a $N \in \mathbb{N}$ samples sinusoid be:*

$$x[n] = A_k \cdot cos(2\pi f_k n + \Phi_k) \cdot \Pi_{B_k, E_k}[n]$$

with $\frac{150}{44100} = 0.0034 < f < 0.4966 = \frac{21900}{44100}$ and $N \leq 8192$, then

$$\boxed{\frac{\sum_{n=0}^{N-1} n \cdot x[n]^2}{\sum_{n=0}^{N-1} x[n]^2} = \frac{B_k + E_k}{2} \pm \varepsilon}$$

with approximately (error maximized by numerical evaluation on a grid of points),

$$|\varepsilon| < 25.2$$

Proof. Let $B := 4\pi f_k B_k + 2\phi_k$,$E := 4\pi f_k E_k + 2\phi_k$,$\Delta := 2\pi f_k$ for easier manipulation,
We recall from lemma 3.1 that:

$$\begin{aligned}
&\sum_{n=0}^{N-1} x[n]^2 \\
&= \frac{A_k^2}{2}\left(E_k - B_k + 1 + \frac{cos(4\pi f_k E_k + 2\Phi) + cos(4\pi f_k B_k + 2\Phi_k)}{2} + \frac{sin(4\pi f_k E_k + 2\phi_k) - sin(4\pi f_k B_k + 2\phi_k)}{2tan(2\pi f_k)}\right) \\
&= \frac{A_k^2}{2}\left(E_k - B_k + 1 + \frac{cos(E) + cos(B)}{2} + \frac{sin(E) - sin(B)}{2tan(\Delta)}\right)
\end{aligned}$$

And its time-scaled version is computed:

$$\sum_{n=0}^{N-1} nx[n]^2$$
$$= \frac{A_k^2}{2}\sum_{n=0}^{N-1} n + \frac{A_k^2}{2}\sum_{n=0}^{N-1} ncos(4\pi f_k n + \phi_k)$$
$$= \frac{A_k^2}{4}(E_k(E_k+1) - (B_k-1)B_k) + \left(\frac{A_k^2}{8\pi}\frac{\mathrm{d}}{\mathrm{d}f_k}\sum_{n=0}^{N-1} sin(4\pi f_k n + \phi_k)\right)$$
$$= \frac{A_k^2}{4}(E_k(E_k+1) - (B_k-1)B_k)$$
$$+ \frac{A_k^2}{j16\pi}\frac{\mathrm{d}}{\mathrm{d}f_k}\left(\frac{1}{e^{j4\pi f_k}-1}(e^{j(4\pi f_k(E_k+1)+2\phi_k)} - e^{j(4\pi f_k B_k+2\phi_k)} - e^{-j(4\pi f_k(B_k-1)+2\phi_k)} + e^{-j(4\pi f_k E_k+2\phi_k)})\right)$$
$$= \frac{A_k^2}{4}(E_k+B_k)(E_k-B_k+1) + \frac{A_k^2}{j16\pi}\frac{\mathrm{d}}{\mathrm{d}f_k}\left(\frac{2e^{j2\pi f_k}}{e^{j4\pi f_k}-1}(cos(4\pi f_k(E_k+0.5)+2\phi_k) - cos(4\pi f_k(B_k-0.5)+2\phi_k))\right)$$
$$= \frac{A_k^2}{4}(E_k+B_k)(E_k-B_k+1) - \frac{A_k^2}{16\pi}\frac{\mathrm{d}}{\mathrm{d}f_k}\left(\frac{cos(4\pi f_k(E_k+0.5)+2\phi_k) - cos(4\pi f_k(B_k-0.5)+2\phi_k)}{sin(2\pi f_k)}\right)$$
$$= \frac{A_k^2}{4}(E_k+B_k)(E_k-B_k+1) - \frac{A_k^2}{8sin(2\pi f_k)}((2B_k-1)sin(4\pi f_k B_k + 2\phi_k - 2\pi f_k) - (2E_k+1)sin(4\pi f_k E_k + 2\phi_k + 2\pi f_k))$$
$$- \frac{A_k^2}{8sin(2\pi f_k)tan(2\pi f_k)}(cos(4\pi f_k B_k + 2\phi_k - 2\pi f_k) - cos(4\pi f_k E_k + 2\phi_k + 2\pi f_k))$$

after developing the above expression, we obtain:

$$\sum_{n=0}^{N-1} nx[n]^2 = \frac{A_k^2}{4}(E_k+B_k)(E_k-B_k+1) + \frac{A_k^2}{4tan(\Delta)}(E_k sin(E) - B_k sin(B))$$
$$+ \frac{A_k^2}{8}(2B_k cos(B) + 2E_k cos(E) + cos(E) - cos(B)) + \frac{A_k^2}{8tan(\Delta)^2}(cos(E) - cos(B))$$

Hence,

$$\sum_{n=0}^{N-1} n\cdot x[n]^2 - \frac{B_k+E_k}{2}\sum_{n=0}^{N-1} x[n]^2$$
$$= \frac{A_k^2}{4tan(\Delta)}(E_k sin(E) - B_k sin(B)) + \frac{A_k^2}{8}(2B_k cos(B) + 2E_k cos(E) + cos(E) - cos(B)) + \frac{A_k^2}{8tan(\Delta)^2}(cos(E) - cos(B))$$
$$- \frac{A_k^2}{8}(cos(E) + cos(B))(E_k+B_k) - \frac{A_k^2}{8tan(\Delta)}(sin(E) - sin(B))(E_k+B_k)$$
$$= \frac{A_k^2}{8tan(\Delta)}(E_k-B_k)(sin(E)+sin(B)) + \frac{A_k^2}{8tan(\Delta)^2}(cos(E) - cos(B)) + \frac{A_k^2}{8}(E_k-B_k+1)(cos(E) - cos(B))$$

So,

$$\varepsilon = \frac{\sum_{n=0}^{N-1} n\cdot x[n]^2}{\sum_{n=0}^{N-1} x[n]^2} - \frac{B_k+E_k}{2} = \frac{\sum_{n=0}^{N-1} n\cdot x[n]^2 - \frac{B_k+E_k}{2}\sum_{n=0}^{N-1} x[n]^2}{\sum_{n=0}^{N-1} x[n]^2}$$
$$= \frac{\frac{A_k^2}{8tan(\Delta)}(E_k-B_k)(sin(E)+sin(B)) + \frac{A_k^2}{8tan(\Delta)^2}(cos(E)-cos(B)) + \frac{A_k^2}{8}(E_k-B_k+1)(cos(E)-cos(B))}{\frac{A_k^2}{2}\left(E_k-B_k+1+\frac{cos(E)+cos(B)}{2}+\frac{sin(E)-sin(B)}{2tan(\Delta)}\right)}$$
$$= \frac{1}{4}\frac{\frac{1}{tan(\Delta)}(E_k-B_k)(sin(E)+sin(B)) + \frac{1}{tan(\Delta)^2}(cos(E)-cos(B)) + (E_k-B_k+1)(cos(E)-cos(B))}{E_k-B_k+1+\frac{1}{2}(cos(E)+cos(B))+\frac{1}{2tan(\Delta)}(sin(E)-sin(B))}$$
$$= \frac{\frac{1}{2\Delta tan(\Delta)}(E-B)(sin(E)+sin(B)) + \frac{1}{2\Delta}(E-B)(cos(E)-cos(B)) + \frac{1}{tan(\Delta)^2}(cos(E)-cos(B)) + (cos(E)-cos(B))}{\frac{2}{\Delta}(E-B)+\frac{2}{tan(\Delta)}(sin(E)-sin(B))+4+2(cos(E)+cos(B))}$$
$$= \frac{\frac{1}{2tan(\Delta)}(E-B)(sin(E)+sin(B)) + (\frac{1}{2}(E-B)+\frac{\Delta}{tan(\Delta)^2}+\Delta)(cos(E)-cos(B))}{2(E-B)+\frac{2\Delta}{tan(\Delta)}(sin(E)-sin(B))+\Delta(4+2cos(E)+2cos(B))}$$

This expression couldn't be maximized symbolically. Attached simulations with Maple show that although there are critical continuous values of E, B for whom the error is huge, the discrete values of E_k, B_k seem to play an important role in the stability of the error.

In the end, some bounds had to be obtained by evaluating this expression numerically in a grid of points. This simulation was programmed in C++ (file `error_support_center_estimation_maximization.cc`), which defines the following grid of points:

- $f_{min} = 150/44100$,$f_{max} = 0.5 - f_{min}, N \leq 8192$
- $f_k = f_{min} + \frac{f_{max}-f_{min}}{1000}m$ with $m = 0..1000$
- $E = 0 + \frac{2\pi}{1000}m$ with $m = 0..1000$
- $B = E + 4\pi f_k m$ with $m = 0..8192$

In this way we get obtain that, for $\frac{150}{44100} = 0.0034 < f < 0.4966 = \frac{21900}{44100}$ and $N \leq 8192$

$$\varepsilon < 25.2$$

On the other hand, in most cases the error will be even significantly smaller. The reader will find how the successive local maximums are found for this set of parameters and $0.0034 < f < 0.25$ in file:
`error_support_center_estimation_maximization_data_fmin150_44100.txt`.

In this data we notice that for $0.0428571 < f < 0.25$ the error is smaller than 2, and that as we approach values that are next to 0, the error grows faster to reach the specified bound. The same behavior occurs in frequencies nearby 0.5. In conclusion, error seems to be maximum next to 0 and Nyquist (0.5), which is intuitively comprehensible, but this estimation works much better for frequencies that are away from both ends.

□

Theorem 3.5. *Let a $N \in \mathbb{N}$ samples sinusoid be:*

$$x[n] = A_k \cdot cos(2\pi f_k n + \Phi_k)\cdot \Pi_{B_k,E_k}[n]$$

with $\frac{150}{44100} = 0.0034 < f < 0.4966 = \frac{21900}{44100}$ *and* $N \leq 8192$*, and let us chose a time sample reference* $k_0 \in [0, N-1]$

$$\boxed{\frac{1}{\sum_{k=0}^{N-1} X[k]^2}\sum_{k=0}^{N-1}|X[k]|^2\Re\left(\frac{DFT\{(n-k_0)x[n]\}[k]}{X[k]}\right) = \frac{B_k+E_k}{2} - k_0 \pm \varepsilon}$$

with approximately (error maximized by numerical evaluation on a grid of points),

$$|\varepsilon| < 25.2$$

Contrary to the suggested estimation of the sinusoid's length, **this estimation doesn't seem to depend on zero padding**. On the other hand this theorem also works reasonably well (small ε) when dealing with signals that have bounded frequencies. Let us consider the same example that was used in the estimation of the sinusoid's support length. Let $x[n]$ be a musical sinusoid sampled at 44100Hz, $f_{min} = \frac{150}{44100}, f_{max} = \frac{21900}{44100}$, then the center of the sinusoid support seem to be estimated blindly with an error of $|\varepsilon| = 25.2$ samples (0.6ms). In this report we couldn't prove that the maximums seem to occur when $f = f_{min}$ and $f = f_{max}$, and we couldn't give an error bound in terms of f_{min} and f_{max}. Future work should fix this. However, by now, we recommend the reader to obtain new approximate error bounds for other f_{min} by running the C++ simulation program (file `error_support_center_estimation_maximization.cc`).

Proof.

$$\begin{aligned}
&\frac{1}{\sum_{k=0}^{N-1} |X[k]|^2} \sum_{k=0}^{N-1} |X[k]|^2 \Re\left(\frac{DFT\{(n-k_0)x[n]\}[k]}{X[k]}\right) \\
&= \frac{1}{\sum_{k=0}^{N-1} |X[k]|^2} \sum_{k=0}^{N-1} |X[k]|^2 \Re\left(\frac{DFT\{nx[n]\}[k] - k_0 DFT\{x[n]\}[k]}{X[k]}\right) \\
&= \frac{1}{\sum_{k=0}^{N-1} |X[k]|^2} \sum_{k=0}^{N-1} |X[k]|^2 \Re\left(\frac{DFT\{nx[n]\}[k]}{X[k]} - k_0\right) \\
&= -k_0 + \frac{1}{\sum_{k=0}^{N-1} |X[k]|^2} \sum_{k=0}^{N-1} |X[k]|^2 \Re\left(\frac{DFT\{nx[n]\}[k]}{X[k]}\right) \\
&= -k_0 + \frac{1}{\sum_{k=0}^{N-1} |X[k]|^2} \sum_{k=0}^{N-1} |X[k]|^2 (-1) \frac{\mathrm{d}\phi}{\mathrm{d}w}[k] \\
&= -k_0 + \frac{N}{N \sum_{n=0}^{N-1} x[n]^2} \sum_{n=0}^{N-1} n \cdot x[n]^2 \\
&= \frac{B_k + E_k}{2} - k_0 \pm \varepsilon
\end{aligned}$$

Using Parseval, lemma 2.2 and lemma 2.3. □

3.4 Time-localization of windowed sinusoids: deconvolving the window

It easy to see intuitively that previous results are not valid when a non rectangular $h[n]$ is used (what we will call "windowed sinusoids"). The estimation of the sinusoid's support center is based on the centroid, which now will be displaced to the center of the window, where the window weights $x[n]$ with greater values. On the other hand, the estimation of the sinusoid's support length relies on the fact that its quadratic amplitude remains constant, which is no longer true when it is windowed. In other words, if we still use the above formulas we will compute the centroid and the area of the windowed squared sinusoid, which don't correspond at all with its original values. Figure 3 illustrate those ideas.

Now, two directions could be taken: adapt the formulas to be valid for a windowed sinusoid or adapt the windowed spectra to turn it into non-windowed spectra and use the previous results. No results were obtained with the first approach, so the second one was chosen.

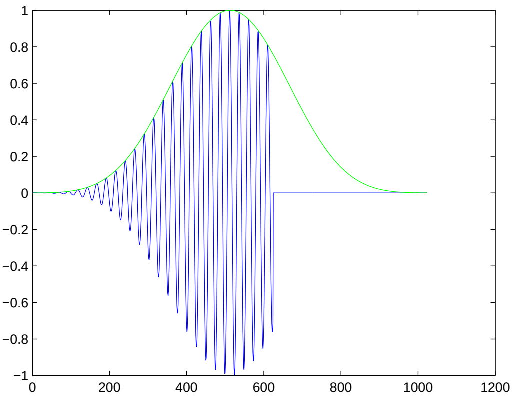


Figure 3: Sinusoid windowed with a Blackman-Harris -92dB window.

So let us try to "unwindow" the DFT. The convolution theorem in the discrete domain says ([PG96]):

$$DFT\{u[n]v[n]\}[k] = \sum_{m=0}^{N-1} U[m]V[k-m]$$

Therefore, from $X_h[k]$, we may obtain $X[k]$ as follows:

$$X[k] = DFT\{x[n]\} = DFT\left\{x_h[n]\frac{1}{h[n]}\right\}[k] = \sum_{m=0}^{N-1} X_h[m]DFT\left\{\frac{1}{h[n]}\right\}[k-m]$$

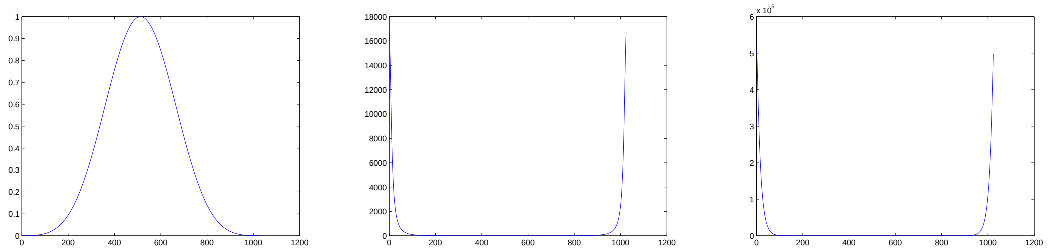
Now, we have seen a way to obtain $X[k]$ from $X_h[k]$ but nothing has improved. Windowing is often performed because sinusoids are better separated using windows with lower secondary lobes which consequently cause much less overlap. If we deconvolve the window, then we exactly obtain the same result that we would have had with the unwindowed DFT, where there's much more overlapping between sinusoids. So, what we have done until now is still completely useless.

However, things get better if we **only deconvolve the main lobes of the sinusoid**. Taking into account that the window usually concentrates the energy in the main lobe of each sinusoid, a deconvolution that ignores the other terms may be seen a possible good approximation which could mix the good things from the windowed analysis (sinusoid separation) and the unwindowed analysis (time-localization).

But this isn't true for a simple reason: $\frac{1}{h[n]}$ tends to infinity in both ends and this introduces huge values in its DFT. Those huge values make the small values coming from the secondary lobes of the sinusoid significant. See figures 4(a),4(b),4(c).

That is why a further step was needed. The following considerations were taken into account: Usually frame processing with the DFT is performed over overlapped frames of the input which are then overlapped-added to the output (one of the first articles about this is [WL84]). In particular it is possible to take only the central part of the window (where signals are less sensitive to the error in the DFT processing that temporal unwindowing amplifies in both ends) and simply crossmix those parts from consecutive frames. This is how we did in article [MV06] with excellent results concerning to audio quality. This process is illustrated in figure 4.

So, after those explanations it makes sense to **unwindow only the part of the frame that will be used for the output**, i.e. the central part that for a hopsize of $N/4$ corresponds to samples $[\frac{N}{4}..\frac{N}{4}-1]$. Now, in figure 5(c) we can see as the DFT has much smaller values.



(a) $h[n]$ (b) $\frac{1}{h[n]}$ (c) $DFT\left\{\frac{1}{h[n]}\right\}$

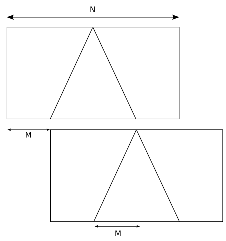
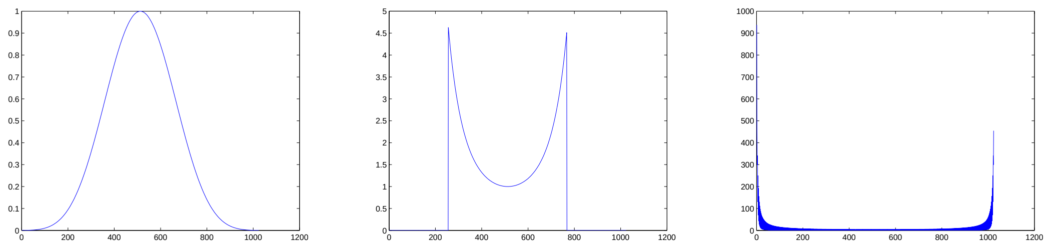


Figure 4: Overlap and add with hopsize M=N/4



(a) $h[n]$ (b) $\frac{\Pi_{\frac{N}{4},\frac{N}{4}-1}[n]}{h[n]}$ (c) $DFT\left\{\frac{1}{h[n]\Pi_{\frac{N}{4},\frac{N}{4}-1}[n]}\right\}$

Consequently the deconvolution will provide much more precise values by only using the coefficients of the main lobes of the sinusoid's DFT.

In particular, some matlab simulations are provided (see `time_localization_1sinusoid_limited_spectrum.m`) that implement this method only using the first main lobe of the sinusoid. The following algorithm is used:

- The main lobe of the windowed DFT is detected: maximum surrounded by two local minima.
- Circular convolution of the main lobe by $DFT\left\{\frac{\Pi_{\frac{N}{4},\frac{N}{4}-1}}{h[n]}\right\}$, considering the other coefficients 0. The result is stored in a separate buffer.
- We find the closest local maximum from the result of the convolution. If such maximum is lower than the maximum of the windowed lobe, this sinusoid is considered to be out of the unwindowed zone, so it won't be

processed. Otherwise, the algorithms developed for the unwindowed case are now executed, and we get the time-localization of the signal in the part of the frame that will be used for the output.

In simulations, this procedure was able to reproduce the original unwindowed sinusoid's main lobe with reasonable accuracy and improved the time-localization estimators' values obtained with an unwindowed DFT in some cases.

3.5 Time-localization of M sinusoids: main lobes processing

All previous results have been developed to apply them in the future to real signals where sinusoids are separated only if big N and windows with low secondary lobes are used. For example, recent work on audio blind separation ([MV06]) made us work with $N = 8192$ and a Blackman-Harris -92dB window. This is why, in fact, we felt that time-localization techniques should be revised to face up this new difficult high resolution framework.

So all the previous techniques have been chosen or thought to obtain results only from the main lobes of each sinusoid present on the complex DFT. Time-localization techniques depend on sums of energy coefficients that are spread all over the lobe and most of the energy of the sinusoid is in its main lobes, so this shouldn't introduce a lot of error even when some coefficients vary slightly from their ideal value because of overlap of other sinusoid lobes. Matlab simulations back up this hypothesis. File `time_localization_1sinusoid_limited_spectrum.m` finds the time-localization parameters of a sinusoid from its limited range of coefficients of the DFT and file `time_localization_Msinusoids.m` estimates the time parameters of multiple randomly time-localized summed sinusoids.)

When sinusoids overlap, a critical point of the algorithm consists in choosing appropriately which lobes are processed together as a single sinusoid. We took two approaches:

- Select only the main lobe
- Select all lobes that are between the peak's maximum and the middle point between consecutive peak maximums may be considered.

In `time_localization_Msinusoids.m` these two approaches can be selected by setting variable `get_main_lobe_only=1` or `get_main_lobe_only=0` respectively but preliminary simulations proved that `get_main_lobe_only=1` worked better.

Finally, the most delicate step was to find the way to take available data from the sinusoid from the windowed DFT and then transform it to be able to use the time-localization techniques developed for the unwindowed case. In the previous section we already discussed how to do that with a partial unwindowing (convolution for its inverse) that takes into account that frame processing in the context of overlap-and-add only uses the central part of the resulting DFT.

4 Conclusions

Several tests of time-localization of sinusoids from random sums of them were done. Results are encouraging, although in order to extend those methods to real signals in practical tasks, there still a lot of work to be done.

4.1 Contributions

- In section Frequency analysis (1.3), a model for the DFT of a time-localized sinusoid is provided and should help the reader to better understand the DFT of real musical signals. Often, in an academic context, the sinusoid deltas are said to be appear convoluted with the window DFT, however this is only true for synthetic signals. For real signals, it has been shown that time-localization may have a much stronger effect, causing the appearance of broad sincs even if the window DFT is narrow.
- This report provides to the reader, in one single paper, all the significant results reviewed in [WM03] proved and explained in detail for practical use (results are given in terms of the DFT i.e. in the context of Digital Signal Processing).
- Such results are extended, providing a way to estimate the length of a sinusoid support. Previous research ([Röb03],[BDDS04],[WM03]) only computed the center of the time-localized sinusoid's support and relied on the particular cases $E_k = N - 1$ or $B_k = 0$. This is not true for short attacks, like for instance drums, specially when audio is analyzed and/or processed with big windows (> 2048 samples).
- Adhoc procedures ([Röb03]) to correct the effect of non-rectangular windowing on the presented methods have been substituted with a time limited deconvolution that gives promising results.

4.2 Ideas for future work

- More testing of the presented methods is needed, in particular for real signals.
- There's a lot of room to improve section 3.4 "Time-localization of windowed sinusoids". Instead of $\frac{\Pi_{\frac{N}{4},\frac{N}{4}-1}[n]}{h[n]}$ we could look for a signal that set an optimum trade-off between its bandwidth the time-localization estimators error. An unwindowing signal with narrower band would enable a more precise deconvolution and this might be possible adding a minimal error in the time-localization estimators. For example the abrupt discontinuous transitions at both ends could be replaced by smoother ones.
- On the other hand, also in the deconvolution section, the windowed DFT could be deconvoluted in zones instead of peaks. In fact, in audio processing, the lower frequencies are those whose overlap is greater with the other, so it may be enough to deconvolve the lower frequencies and the higher ones. A more general approach would be to deconvolve only the biggest peaks and deconvolve all the other peaks together.
- A multiresolution bank of filters combined with unwindowed DFT could be a good alternative to selective deconvolution. In fact, the later was inspired in this first idea. Both methods could be compared.
- The error caused by taking only the main lobes of the sinusoid DFT should be studied and bounded.
- Formula 2 could be developed in the case of a Blackman-Harris or Hanning window. The result may be too complex but it may help to understand how to improve the deconvolution process.
- The suggested models of time-localized sinusoids (formulas 1 or 2) could be used in frequency sound synthesis, to add time resolution to the generated sounds. For that purpose it will be particularly useful to develop formula 2 and approximate it with a simpler one.

- Information from the phase shift of one frame through two consecutive frames (with minimal hopsize) is also used to detect onsets in some references ([BDDS04],[MA06]). This could help to improve the presented methods too.

- Only a numerical error bound for the estimation of the sinusoid's support center was found. A symbolic bound would be nicer, in order to know it for different frequency bounds.

- Time-localization estimators give bad results for sinusoids with frequencies close to 0 or 0.5. Maybe some operation could be done in the spectrum to correct distortion caused by its close alias.

- Errors are given as bounds. However, from an engineering point of view, where the error is not critical for the application (repeated experiment, audio processing, etc), it could be better to give the probabilistic distribution of the error, given different distributions of the blind parameters (f_k,E_k,B_k,A_k). In fact the reader can check that given bounds for the error in the estimation of the support center or length are only achieved by very particular set of values of those parameters. Maybe the notion of mean error would be more appropriate. Hence future work on this issue would consist in obtaining the probability density function of the random variable "Error" conditional to the values of the blind parameters: $f_{Error}(Error = e|\text{choose some between } f_k,E_k,B_k,A_k)$. In this way the engineer could integrate this expression over the area of expected blind parameters (that may change depending on the problem) and obtain the probability of each error value. As a particular case, the support of this function would give us the error bound corresponding to the worst case.

References

[BDDS04] Juan P. Bello, Chris Duxbury, Mike Davies, and Mark Sandler. On the use of phase and energy for musical onset detection in the complex domain. *IEEE Signal Processing Letters*, 11, 2004.

[MA06] Kevin M.Short and Ricardo A.Garcia. Signal analysis using the complex spectral phase evolution (cspe) method. *AES 120th Convention*, 2006.

[MV06] A. Loscos M. Vinyes, J. Bonada. Demixing commercial music productions via human-assisted time-frequency masking. *AES 120th Convention*, 2006.

[PG96] John G. Proakis and Dimitris G.Manolakis. *Digital Signal Processing*. Prentice-Hall, 1996.

[Röb03] Axel Röbel. A new approach to transient processing in the phase vocoder. *DAFX*, 2003.

[WL84] Daniel W.Griffin and Jae S. Lim. Signal estimation from modified short-time fourier transform. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 2, 1984.

[WM03] Stephen W.Hainsworth and Malcolm D. Macleod. Time frequency reassignment: A review and analysis. Technical report, Cambridge University Engineering Department, 2003.