Title: Frequency estimation by zooming

Author 1: Manuel D. Ortigueira

Address: DEE / FCT of UNL and UNINOVA[1]

Campus da FCT da UNL, Quinta da Torre, 2829 – 516 Caparica, Portugal,

Phone: +351 21 2948520

Fax: +351 21 2957786

E-mail: mdo@fct.unl.pt

Author 2: **António S. Serralheiro***

Address: Academia Militar and INESC

R. Alves Redol, 9, 2º, 1000-029, Lisbon, Portugal

Phone: +351 21 3100354

Fax: +351 21 3145843

E-mail: antonio.serralheiro@inesc-id.pt

Abstract

In this paper we deal with a frequency estimation problem using a zoom transform algorithm. In many a signal processing problems, it is of paramount importance the exact determination of the frequency of a signal. Some techniques derived from the FFT, just pad the signal with enough zeros in order to better sample its Discrete-Time Fourier Transform. Our approach, although FFT-based, does not rely on such method, using instead a "zoom" function built around the well-known *sinc(.)* function, resulting in an exact and deterministic method. Its analytic formulation is presented and illustrated with some simulation results over short-time based signals.

---

[1] Also with INESC

1. Introduction

Frequency estimation usually involves the usage of FFT-based estimators, either directly, as it is the case of periodogram, or indirectly as in the MEM, MUSIC and other similar methods. In all the cases, we are looking for the exact position of the spectral peaks in a given spectral estimate $S(f)$, $|f| < 1/2$. However, when using the FFT, $S(f)$ is sampled over an uniform grid and, as a consequence, it is very unlikely that we are successful in obtaining the true peak positions. Better approximations to the positions of the spectra peaks can be obtained using large zero padding, thus leading to very large FFT lengths. In [2] a warped discrete Fourier transform is used and its performance compared with several other procedures, namely: Dichotomous-search, Tretter's linear regression, Kay's phase difference and chirp Z-transform methods. This being an interpolation problem in frequency domain, however, it is very special because we know the interpolating function: it is the Fourier transform. The real problem appears because we are using a DFT implementation. As this defines completely the Fourier transform, we can approach this problem from a quite different point of view: the zooming of a small portion of the spectrum that includes the peak position. To perform the spectral zoom, two different methods of interpolation have been proposed and usually referred as the zoom transform [1]. But, since these methods imply a return to time, modulation and filtering, they are not very useful when dealing with short-time signals. An alternative and simpler algorithm was proposed in [3] that merely explores the fact that the FFT (DFT) is a sampling of the Fourier Transform and, so, it has the whole information we need. This paper further details this topic and is organized as follows: in section 2, we present a general formulation that allows us to choose the frequency search grid. Furthermore, it is also shown that the proposed algorithm is useful in converting a spectral representation from a linear to logarithm scale. In section 3, we present some simulation results and section 4 is devoted to the conclusions and final discussion.

2. The Zoom Algorithm

Let , $x(n)$, $n = 0, \ldots, L - 1$ denote an L-length sample sequence. Every $N \geq L$ point DFT sequence represents samples of the Discrete-Time Fourier Transform (DTFT):

$$X(e^{j\omega}) = \sum_{n=0}^{L-1} x(n)\, e^{-j\omega n} \tag{1}$$

Let $X_N(k)$ denote the DFT of $x(n)$, corresponding to sampling $X(e^{j\omega})$ at a uniform grid:

$$X_N(k) \;= DFT[x(n)] = X(e^{j\frac{2\pi}{N}k}) \qquad k = 0,\ \ldots,\ N - 1 \qquad N \geq L \tag{2}$$

Its inverse, (DFT^{-1}) is

$$X(n) = \frac{1}{N} \sum_{k=0}^{N-1} X_N(k).e^{j\frac{2\pi}{N}kn} \qquad n = 0,\ \ldots,\ N - 1 \tag{3}$$

Substituting equation (3) into equation (1) results in:

$$X(e^{j\omega}) = \frac{1}{N} \sum_{k=0}^{N-1} X_N(k).G(\omega,\ k) \tag{4}$$

where $G(\omega, k)$ is given by

$$G(\omega,\ k) = \frac{1 - e^{-j(\omega - \frac{2\pi}{N}k)L}}{1 - e^{-j(\omega - \frac{2\pi}{N}k)}} \qquad \$\$(5)$$

for $|\omega| \leq \pi$ and $0 \leq k < N$. It is not hard to show that:

$$G(\omega,\ k) = L.\frac{Sa\left[(\omega - \frac{2\pi}{N}k)\,\frac{L}{2}\right]}{Sa\left[(\omega - \frac{2\pi}{N}k)\,\frac{1}{2}\right]}\, e^{j(\omega - \frac{2\pi}{N}k)\frac{L-1}{2}} \tag{6}$$

where $Sa(x) = \frac{sin(x)}{x}$. Since $\omega = 2\pi f$, we obtain:

$$G(f,\ k) = L.\frac{sinc\left[(f - \frac{k}{N})L\right]}{sinc\left[f - \frac{k}{N}\right]}\, e^{j\pi(f - \frac{k}{N})(L-1)} \tag{7}$$

So, equations (4) and either (6) or (7) allows us to zoom into the frequency region of interest. Of course, we are not interested in zooming the whole spectrum[1], just a given band.

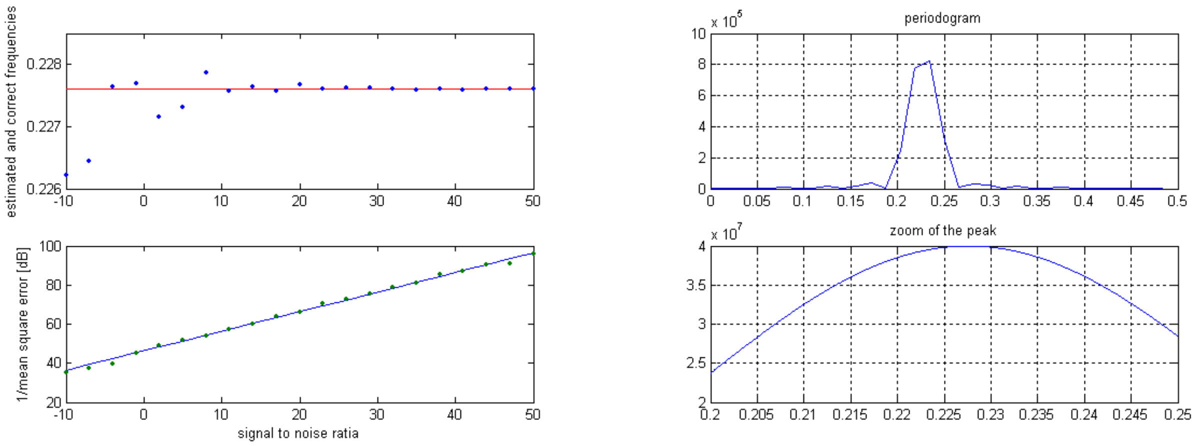
---

[1] But we can do it, if we find it useful.

3. Simulation results

3.1. Frequency Estimation

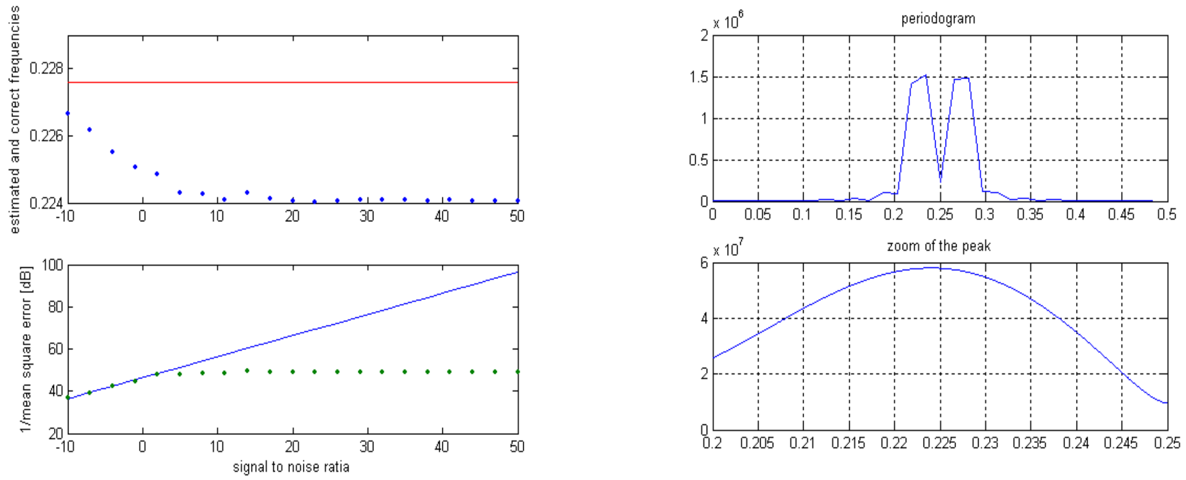
In this section, we present some simulation results obtained with a sinusoidal signal of angular frequency $\omega_0 = 1.43$ $(f_0 = 0.2276)$ for different signal-to-noise ratia (SNR), spanning from -10 to $+50$ dB. The Cramer-Rao bound is included for reference as in [2] and the reciprocal value of the variance in dB is also shown. For these simulations, the following values were used: $L = 24$, $N = 64$, the band of frequencies to be zoomed was $f \in [0.20, 0.25]$ where *1000* points were computed in a total of *100* runs (figure 1a)). Figure 1b) depicts the periodogram and the corresponding peak zoom for the same simulations of figure 1a).



a) *b)*

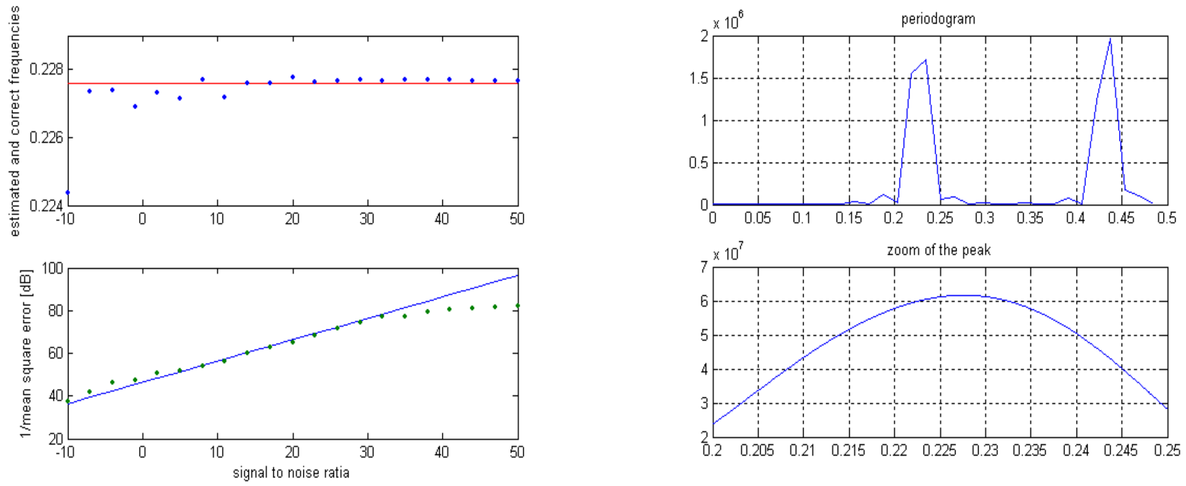
Figure 1 – a) Estimated frequency by zooming showing on red the exact frequency value (top) and the mean square error in (– dB) with the Cramer-Rao bound in blue (bottom), b) Periodogram and zoom of the peak for a sinusoidal signal, $f_0 = 0.2276$.

More simulatins were performed, and their results presented in figures 2 and 3, using the following values: $L = 24$, $N = 64$, and the band of frequencies to be zoomed was $f \in [0.20, 0.25]$ where *1000* points were computed in a total of *100* runs. In both cases, an equal amplitude sinusoid was now added to the original signal, plus noise as before. The new sinusoid has a frequency equal to $1.2f_0$ (fig. 2) and equal to $1.9f_0$ (fig. 3). As it can be seen, the presence of the second sinusoid produces a bias in the estimated frequencies and, as expected, when the two frequencies are closer, the bias becomes higher.



a) *b)*

Figure 2 – Frequency estimation of a sinusoidal signal ($f_0 = 0.2276$) corrupted by an equal amplitude sinusoid $f_1 = 1.2\, f_0$ and noise, showing, a) from top to bottom: estimated frequency, inverse of mean square error (dB), b) from top to bottom: periodogram and zoom of the peak.



a) *b)*

Figure 3 – Frequency estimation of a sinusoidal signal ($f_0 = 0.2276$) corrupted by an equal amplitude sinusoid $f_1 = 1.9 f_0$ and noise, showing a) from top to bottom: estimated frequency, inverse of mean square error (dB), b) from top to bottom: periodogram and zoom of the peak.

3.2. Frequency Scale Conversion

Another application of this algorithm is the conversion of a linear frequency scale into a logarithmic one. In figure 4 we present the use of the algorithm to convert the transfer function of a 10^{th}-order low-pass FIR filter with a bandwidth of *0.0125* from a linear (blue plot) to log frequency scale (green plot). This conversion can be easily done using equation (7) as an interpolation factor for the filter frequency response to the new frequency scale.

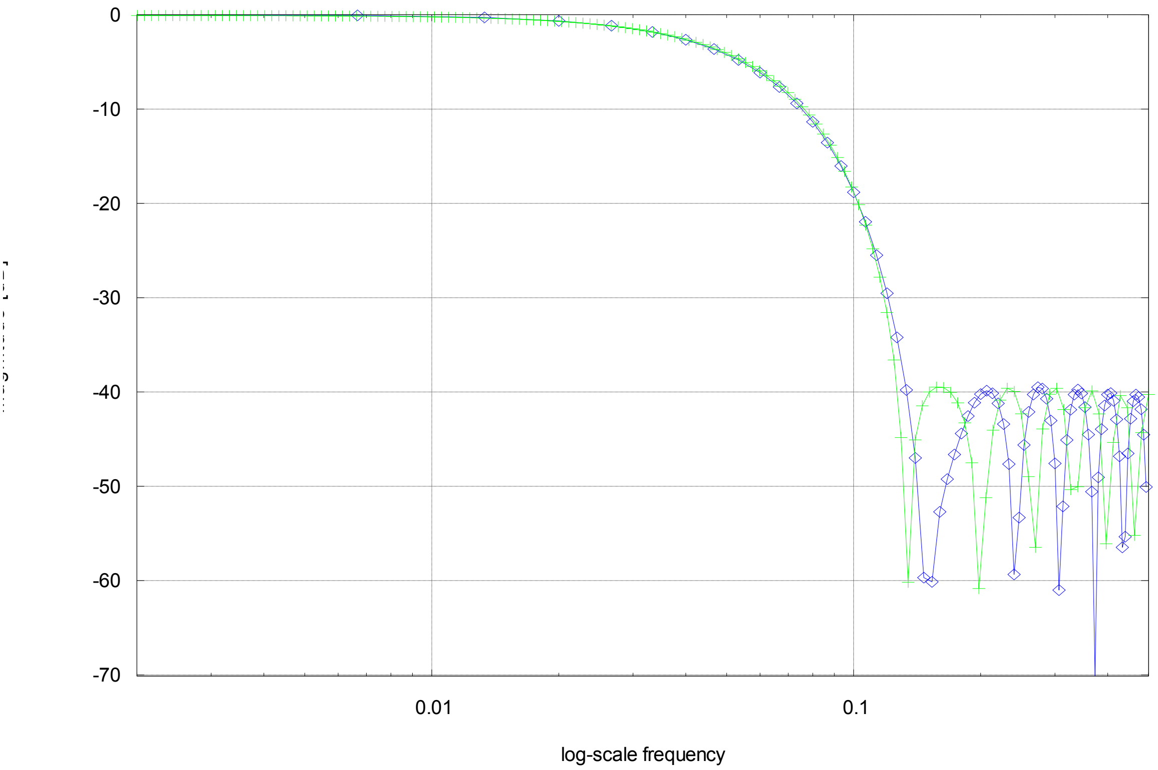


Figure 4 – Example of linear (º and blue line) to log (+ and green line) frequency conversion of a 10^{th}-order low-pass FIR filter with a cut-off frequency of *0.0125*.

4. CONCLUSIONS

A simplified algorithm was presented with applications ranging from frequency estimation to frequency scale conversion. Often the frequency estimation problem involves the usage of FFT-based estimators, either directly (e.g. periodogram) or indirectly as in the MEM, MUSIC and other similar methods. In all the cases, we are looking for the exact position of the peaks in a given spectral estimate. For instance, when using the FFT, the spectral estimate *S(f)* is sampled over an uniform grid and, because of this sampling, the true peak positions are unlikely to be known. This problem can be easily dealt by padding the signal with zeros in order to increase the length of the FFT, thus, refining the sampling grid. The underlying idea is to refine the sampling of the DFT but to the expense of large computational requirements. Other methodologies [1] approach this problem by zooming of a small portion of the spectrum where the peak position is comprised. An alternative, exact (in the sense that all the spectral information lays in the DFT) and simpler algorithm has been proposed in [3] and is further exploited in this paper. As shown, this algorithm is based on the *sinc(.)* function that, although widely known and used for decades, has not been, so far, applied to this spectral estimation topic.

References

1. C.S. Burrus, J.H. McClellan, A.V. Oppenheim, T.W. Parks, R.W. Schafer, and H.W. Schuessler, *Computer-Based Exercices for Signal Processing using MatLab*, Prentice-Hall, 1994.
2. S. Franz, S. K. Mitra, and G. Doblinger, “Frequency estimation using warped discrete Fourier transform”**,** in *Signal Processing*, vol. 83, pp. 1661-1671, 2003.
3. M. D. Ortigueira, C. Matos and M. S. Piedade, “Fractional Discrete-Time Signal Processing: Scale Conversion and Linear Prediction”, in *Nonlinear Dynamics*, vol. 29, pp. 173-190, 2002.