Signal Processing 114 (2015) 245–250

Contents lists available at ScienceDirect

Signal Processing

journal homepage: www.elsevier.com/locate/sigpro

Fast communication

Fine resolution frequency estimation from three DFT samples: Case of windowed data

Çağatay Candan

Department of Electrical and Electronics Engineering, Middle East Technical University (METU), 06800 Ankara, Turkey

ARTICLE INFO

Article history:
Received 24 August 2014
Received in revised form
18 January 2015
Accepted 10 March 2015
Available online 18 March 2015

Keywords:
Frequency estimation
Interpolated DFT
IpDFT
Rife–Vincent windows
Side-lobe suppression

ABSTRACT

An efficient and low complexity frequency estimation method based on the discrete Fourier transform (DFT) samples is described. The suggested method can operate with an arbitrary window function in the absence or presence of zero-padding. The frequency estimation performance of the suggested method is shown to follow the Cramer–Rao bound closely without any error floor due to estimator bias, even at exceptionally high signal-to-noise-ratio (SNR) values.

1. Introduction

Frequency estimation of complex exponential signals is a fundamentally important non-linear parameter estimation problem arising in several applications. Recently, an efficient frequency estimation technique based on the samples of Discrete Fourier Transform (DFT) has been proposed in [1,2]. An important restriction of this technique and several others such as [3–7] is the requirement of DFT calculation with the rectangular window without any zero-padding. The present work aims to remove both of these restrictions by adapting the bias correction factor in [2] to the window of interest.

The frequency estimation method given in [1,2] consists of two stages. In the first stage (coarse frequency estimation), N-point Discrete Fourier Transform (DFT) of the N-point input is calculated. In the second stage (fine frequency estimation), the DFT bin with the maximum magnitude (k_p) and its immediate left (k_p-1) and right neighbors (k_p+1) are used to estimate the fine part of the frequency:

$$\widehat{\delta} = c_N \,\mathrm{Real}\left\{\frac{R[k_p-1]-R[k_p+1]}{2R[k_p]-R[k_p-1]-R[k_p+1]}\right\}. \tag{1}$$

Here c_N is $\frac{\tan(\pi/N)}{\pi/N}$ for the rectangular window. The final frequency estimate is formed by combining the results of both stages, $\widehat{\omega} = 2\pi(k_p+\widehat{\delta})/N$ radians/sample. The first stage of this estimator works with the rectangular window in the absence of zero-padding. Further details on this method can be found in [1,2].

In many applications, the DFT calculation is implemented with a properly selected window to suppress the interference caused by undesired spectrum components [8–10]. For example, in pulse Doppler radars, the desired signal (target echo at a specific Doppler frequency) coexists with other echos such as clutter signal, undesired target echos and jamming signal. With the application of windowing, the impact of interfering components on the desired signal is reduced. Due to emergence of the same problem in many applications, a number of frequency estimation methods with windowed data are given in the instrumentation and measurement literature [11–14]. A particularly well known estimator is the one utilizing Rife–Vincent class-I windows

E-mail address: ccandan@metu.edu.tr

http://dx.doi.org/10.1016/j.sigpro.2015.03.009
0165-1684/

is [12,13]

$$\widehat{\delta} = (M+1)\frac{|R_w[k_p+1]| - |R_w[k_p-1|]}{2|R_w[k_p]| + |R_w[k_p-1]| + |R_w[k_p+1]|}. \quad (2)$$

Here $|R_w[k_p]|$ is the peak value of the *windowed* DFT output where the applied window is the Rife–Vincent class-I window having the parameter $M=\{0,1,2,\ldots\}$. Rife–Vincent class-I windows are equivalent to the rectangular and Hann windows for $M=0$ and $M=1$, respectively. Due to the widespread usage of Hann window $M=1$ case of this estimator is important for many applications.

In an analogy with (1), the window specific correction factor for the estimator in (2) is $(M+1)$. It should be clear that the estimator (2) and its correction factor are specific to a particular window. To the best of our knowledge, apart from Duda's work [14], all other estimators in the literature are also derived for specific windows [9–11]. In [14], Duda presents a novel approach based on compensating the window specific estimator bias through a high order polynomial interpolation. The approach presented in this paper is very similar, in principle, to the one of Duda's. Here, we only adapt the bias compensation factor c_N in (1) to the window. The main advantage of the proposed method is its improved performance in spite of its low computational complexity.

2. Preliminaries

A complex exponential signal with the normalized frequency f in $[0,1)$ and with the complex amplitude A is observed under additive white Gaussian noise:

$$r[n] = Ae^{j(2\pi fn+\phi)} + v[n], \quad n=\{0,\ldots,N-1\}. \quad (3)$$

The frequency f can also be denoted in terms of the DFT bins, $f=(k_p+\delta)/N$ where k_p is an integer in $[0,N-1]$ and δ is a real number in $-1/2<\delta<1/2$ [2]. The noise $v[n]$ is circularly symmetric complex valued Gaussian noise with zero mean and σ_v^2 variance, $v[n]\sim\mathcal{CN}(0,\sigma_v^2)$. The signal-to-noise ratio is defined as $\mathrm{SNR}=A^2/\sigma_v^2$.

In many applications, the complex exponential signal is observed in the presence of interfering signals. For such applications, the DFT is calculated with a proper a window function to reduce the interference on the frequency estimate. Fig. 1 shows the spectrum with the Hamming window. The Hamming window with its low side-lobes reduces the interference leakage at the cost main-lobe widening [8, Chapter 6]. It can be also seen that the curvature around the peak changes significantly with the applied window. This is the main reason that an interpolation based frequency estimation method for a specific window does not work for any other window.

As noted in the introduction section, the first stage of the method described in [1] calculates the N-point DFT of $r[n]$ and then a peak search in the magnitude spectrum is conducted. This stage aims to estimate the coarse part of the frequency (k_p) as shown in Fig. 1. In the second stage, the fractional part of the frequency (δ) is estimated. For the rectangular windowed signal, δ is estimated through the relation (1) with $c_N=\frac{\tan(\pi/N)}{\pi/N}$. Our goal is to use the same relation for δ estimation, but select the bias-correction coefficient c_N according to the window function.

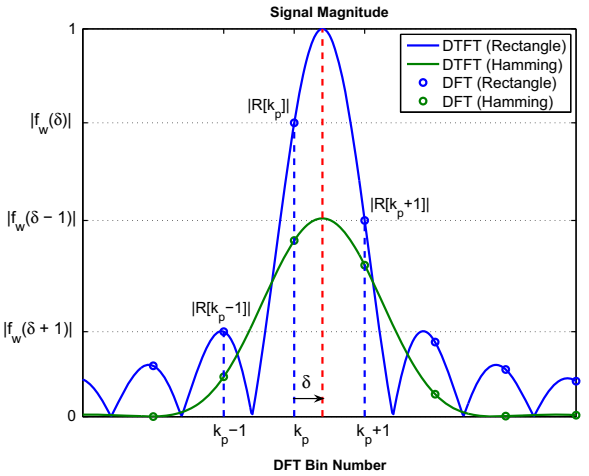

Fig. 1. DTFT and DFT spectrum of the complex exponential waveform with the frequency $k_p+\delta$ bins using rectangle and Hamming windows.

3. Main results

In the first stage of the proposed method, the input signal is transformed to the DFT domain after the application of the real valued window $w[n]$,

$$R[k] = \sum_{n=0}^{N-1} w[n]r[n]e^{-j(2\pi/N_2)kn}, \quad k=\{0,1,2,\ldots,N_2-1\} \quad (4)$$

where N_2 is the number of DFT points, which is possibly larger than N with the application of zero-padding. In the absence of noise, we may take, without any loss generality, $A=1$ in (3) and write $r[n]$ as $r[n]=e^{j2\pi(k_p+\delta)/N_2}$. Then, the spectrum samples $R[k_p+l]$ (l: integer) can be written as follows:

$$R[k_p+l] = \sum_{n=0}^{N-1} w[n]e^{j(2\pi n/N_2)(\delta+l)} = f_w(\delta+l). \quad (5)$$

The window dependent $f_w(\alpha)$ function appearing on the right hand side of (5) is explicitly defined as

$$f_w(\alpha) = \sum_{n=0}^{N-1} w[n]e^{j(2\pi nN_2)\alpha}. \quad (6)$$

For the case of zero-padding ($N_2>N$), the window function $w[n]$ in (6) can be considered as the zero-padded version of N-point window and $f_w(\alpha)$ for integer valued α values is its N_2-point inverse DFT.

3.1. Bias correction factor

An estimate for δ can be produced through the processing of the samples $R[k_p-1]=f_w(\delta+1)$, $R[k_p]=f_w(\delta)$ and $R[k_p+1]=f_w(\delta-1)$ via the relation (1), provided that c_N is made available. Our goal is to set the bias correction factor c_N according the windowing function.

To facilitate the calculation of c_N, we expand the function $f_w(\delta+l)$ into the Taylor series around $\alpha=l$:

$$f_w(\delta+l) = f_w(l) + f_w'(l)\delta + O(\delta^2) \quad (7)$$

Here $f_w'(\alpha)=\frac{j2\pi}{N_2}\sum_{n=0}^{N-1} nw[n]e^{j(2\pi n/N_2)\alpha}$ is the first derivative of the function given in (6). Upon the substitution of

$R[k_p-1]\to f_w(\delta+1)$, $R[k_p]\to f_w(\delta)$ and $R[k_p+1]\to f_w(\delta-1)$ into (1); we get the following relation for δ:

$$\widehat{\delta}=c_N\ \mathrm{Real}\left\{\frac{f_w(\delta+1)-f_w(\delta-1)}{2f_w(\delta)-f_w(\delta+1)-f_w(\delta-1)}\right\}. \tag{8}$$

By substituting the series expansion of $f_w(\delta+l)$ around $l=\{-1,0,1\}$ from (7) into (8), we get

$$\widehat{\delta}=c_N\ \mathrm{Real}\left\{\frac{jA_0+A_1\delta+O(\delta^2)}{B_0+jB_1\delta+O(\delta^2)}\right\} \tag{9}$$

where A_0, A_1 and B_0, B_1 are real valued constants with the following definitions:

$$\begin{aligned}A_0&=\mathrm{Imag}\{f_w(1)-f_w(-1)\}\\ A_1&=f_w'(1)-f_w'(-1)\\ B_0&=2f_w(0)-f_w(1)-f_w(-1)\\ B_1&=\mathrm{Imag}\{2f_w'(0)-f_w'(1)-f_w'(-1)\}.\end{aligned} \tag{10}$$

Once the real part of the ratio given in (9) is calculated, we reach

$$\widehat{\delta}=c_N\frac{A_1B_0+A_0B_1}{B_0^2}\delta+O\left(\delta^2\right). \tag{11}$$

Hence, by selecting c_N as

$$c_N=\frac{B_0^2}{A_1B_0+A_0B_1}, \tag{12}$$

the bias for sufficiently small δ values can be reduced.

It can be verified that the result given in (12) exactly matches the closed formula of $c_N=\tan(\pi/N)/(\pi/N)$ for the rectangular window [1]. Even though it can be possible to find closed form expressions for some windows, such as Rife–Vincent class windows; the multiplicity of windows and the simplicity of the numerical calculation of c_N from (12) render such an effort of limited reward. We do not present any analytical formulas for c_N; but present a MATLAB function for its numerical calculation [15]. The bias correction factors for different windows are shown in Fig. 2. As a further assistance, polynomial fitting operation is applied to c_N values shown in Fig. 2 and the resulting approximating polynomial is given below each curve.

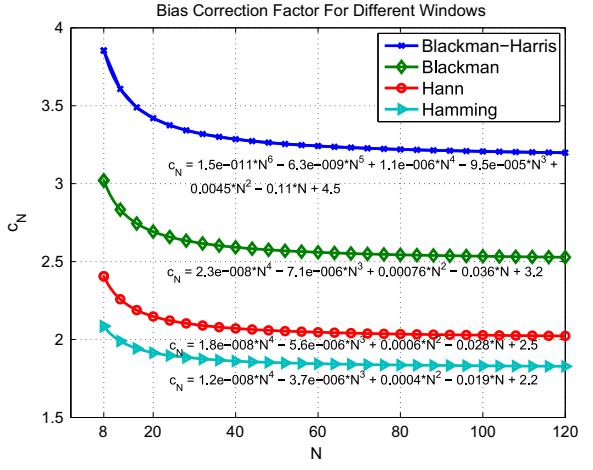

Fig. 2. Bias correction factor c_N for different windows ($N_2=N$).

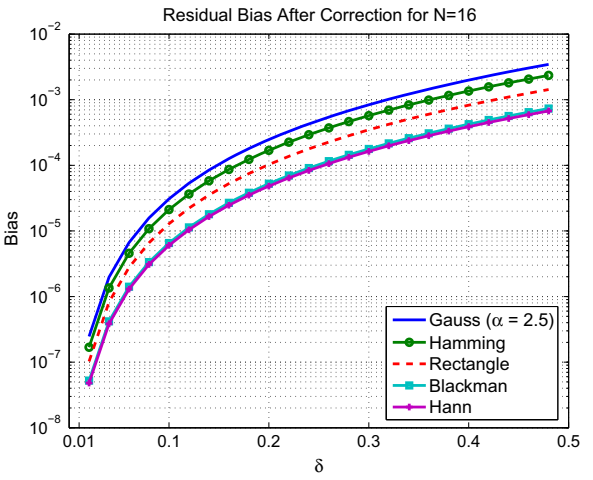

Fig. 3. Residual bias value of the estimator in (1) for different windows after the bias-correction by c_N.

Fig. 3 shows the effect of bias correction for the noiseless case. As expected, for sufficiently small values of δ, such as $|\delta|<0.01$, the correction, practically, removes the bias. Stated differently, the bias after correction becomes so small that the mean square estimation error is not affected by the bias unless operating SNR is extremely large. (The numerical results, given in the following sections, examines this claim in more detail.) Yet, as $|\delta|\to 1/2$, the small δ assumption is challenged and the value of the bias correction by simple scaling depreciates. A remedy to this problem is the application of a non-linearity on $\widehat{\delta}$ as in [2]. Unfortunately, it is difficult to find the "right" non-linear function for each window. Here we follow an alternative route for bias reduction enforcing the operation around $\delta\approx 0$ for all δ values as explained in the next section.

3.2. Proposed method

Previously it has been noted that the bias-correction through the correction factor c_N works well when the true δ magnitude is smaller than 0.01. We suggest to use $\widehat{\delta}$ relation in (1) twice to reduce (virtually eliminate) the bias for all δ values.

The suggested approach is given in Table 1. The first 3 steps of the suggested approach are identical to the bias-corrected operation described earlier. In the 4th step, the input is modulated such that the frequency of the signal after modulation ($r_2[n]$) becomes $k_p+(\delta-\widehat{\delta}_1)=k_p+\delta_2$ in bins. An inspection of Fig. 3 may suggest the smallness of resulting δ_2 after this operation, $|\delta_2|<0.01$. Steps 5 and 6 are conducted to estimate the residual component after modulation (δ_2). Since the true value of δ_2 is expected to be small, the estimate $\widehat{\delta}_2$ is expected to be an accurate representation of δ_2. At the last step, the final frequency estimate is produced. The overall implementation cost is the calculation of N_2 point DFT, calculation of $\widehat{\delta}_1$ (1 complex division and 1 real multiplication), re-calculation $R_2[k_p], R_2[k_p-1], R_2[k_p+1]$ in the second iteration ($3N_2+N$ complex multiplications) and calculation of $\widehat{\delta}_2$ (1 complex division and 1 real multiplication). A detailed implementation is available in [15]. In contrast, Duda's method

Table 1
Proposed method.

1	Calculate the windowed $N_2 \geq N$ point DFT of the input $r[n]$ ($R[k]$)
2	Detect the index (k_p) of the maximum in the DFT spectrum ($0 \leq k_p \leq N_2-1$)
3	Estimate $\widehat{\delta}_1$ from (1) using $R[k_p-1], R[k_p], R[k_p+1]$ and c_N from (12)
4	Construct $r_2[n] = r[n]\exp\left(-j\frac{2\pi}{N_2}\widehat{\delta}_1 n\right), n = \{0,1,\ldots,N-1\}$
5	Calculate the windowed $N_2 \geq N$ point DFT of $r_2[n]$ ($R_2[k]$)
6	Estimate $\widehat{\delta}_2$ from (1) using $R_2[k_p-1], R_2[k_p], R_2[k_p+1]$ and c_N from (12)
7	Output $\widehat{\delta} = \widehat{\delta}_1 + \widehat{\delta}_2$ and $\widehat{\omega} = \frac{2\pi}{N_2}\left(k_p + \widehat{\delta}\right)$ radians/sample

requires, in addition to the first stage DFT calculation, the calculation of at least 5th degree polynomial (10th degree polynomials are used in [14]) for each window function; 3 magnitude calculations (2 real multiplications and 1 square root operation each) and evaluation of at least 5th degree polynomial. The calculation of large degree polynomials and their evaluation are the main disadvantage of Duda's method in comparison to the suggested one.

The main idea of the proposed approach is to use the suggested estimator twice at different operating points. The same idea also appears in the frequency estimator proposed by Aboutanios and Mulgrew for the rectangular windowed data, [4]. The goal of both methods is the same, that is to reduce the estimator bias by moving the "operating point" to an advantageous location.

4. Numerical experiments

Three sets of numerical results are provided to examine the performance of the suggested method. The first set examines the effectiveness of the suggested method over a wide range of SNR values. The second set presents a root mean square error (RMSE) comparison of the estimator for different windows. The third set examines the effect of zero-padding, i.e. N_2-point DFT ($N_2 > N$). In all three experiments, the number of input samples is $N=16$ and the true value of δ is 0.45. Hence, the true frequency is $2\pi(k_p+0.45)/N$ radians/sample where k_p is an arbitrary integer in $[0, N-1]$ with no effect on the fine frequency estimation. The choice of large δ ($\delta \approx 1/2$) and small N leads to a difficult operational scenario, challenging several small δ assumptions in the estimator development. The expression for the Cramer–Rao bound shown in figures is available in [2, Eq. (3)].

Experiment 1: This experiment examines the performance of the method whose steps are given in Table 1 for 16-point DFT ($N_2 = N = 16$) with Hamming window. The correction factor c_N for this experiment is 1.9427.

Fig. 4(a) shows the RMSE of the suggested estimator for the cases of $\widehat{\delta} = \widehat{\delta}_1$ and $\widehat{\delta} = \widehat{\delta}_1 + \widehat{\delta}_2$ where $\widehat{\delta}_1$ and $\widehat{\delta}_2$ are the estimates produced at steps 3 and 6 of Table 1. The goal of this experiment is to examine the effectiveness of the proposed method for two different choices of $\widehat{\delta}$ over a wide range of SNR values.

It can be noted from Fig. 4(a) that the RMSE of the estimator with $\widehat{\delta} = \widehat{\delta}_1$ is limited by the estimator bias at high SNR. Since MSE = (bias)^2 + var(error), where var(error) is the variance of the estimation error; the MSE is limited by the estimator bias at sufficiently high SNR. From Fig. 3, the

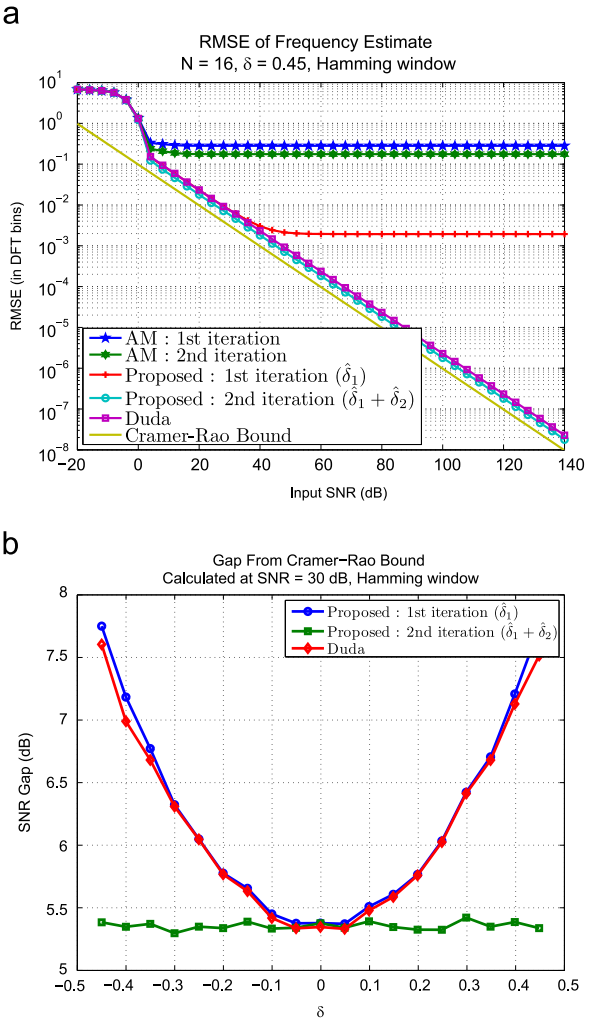

Fig. 4. Performance comparisons for the suggested estimator. (a) RMSE comparison and (b) SNR gap comparison.

bias of Hamming window at the true $\delta = 0.45$ can be read as 2×10^{-3}, which is identical to the RMSE floor observed for $\widehat{\delta} = \widehat{\delta}_1$ in Fig. 4(a) at high SNR. As can be noted from Fig. 4(a), the suggested estimator with $\widehat{\delta} = \widehat{\delta}_1 + \widehat{\delta}_2$ does not suffer from RMSE floor even at high SNR values, such as 140 dB, indicating the effectiveness of the suggested method in the removal of estimator bias. Results for higher N values are similar. In fact, doubling of N or SNR has almost

the same effect in the high SNR region which is the reduction of RMSE by $\sqrt{2}$.

In the same figure the performance of other estimators is also illustrated. The Aboutanios and Mulgrew estimator is an iterative estimator similar to the one presented here [4]. Since the working assumption of Aboutanios and Mulgrew estimator is the utilization of the rectangular window, this estimator works poorly in the presence of any other window. Recently, Aboutanios has extended this work and presented an exact bias compensation which is shown to be only effective at very high SNR values, again for the rectangular windowed data [5]. The performance of the estimator developed in [5] is identical to [4] in the examined scenario.

Duda's estimator operates with arbitrary windows [14]. It can be noted from Fig. 4(a) that the performance of Duda's estimator is almost identical to the first iteration of the proposed estimator; but has a poorer performance than the second iteration by more than 2 dB. This can be most clearly seen from Fig. 4(b) where SNR gap values (required SNR increase to reach the Cramer–Rao bound at a given input SNR) of Duda's and proposed method is given for different true δ values at SNR=30 dB. We further discuss SNR gap of other windows in the next experiment.

Experiment 2: The second experiment compares the estimator performance for three different windows in the absence of zero-padding ($N_2 = N$). The windows used in this experiment are Blackman, Hamming and rectangle windows.

Before the discussion of the experiment results, we would like to emphasize that the application windowing results in the loss of output SNR, which is also denoted as the loss of coherent integration gain in the literature [8, Chapter 6]. The amount of SNR loss for the window of $w[n]$ can be expressed as $\mathrm{SNR}_{\mathrm{loss}}^{\mathrm{windowing}} = 10\log_{10}\left(\frac{\left(\sum_{n=0}^{N-1} w[n]\right)^2}{N\sum_{n=0}^{N-1} w^2[n]}\right)$ dB [8, p. 188]. This loss occurs due to the deviation from SNR maximizing matched filtering type operation. The SNR loss for Hamming and Blackman windows for $N=16$ is 1.54 and 2.65 dB, respectively.

Fig. 5(a) shows the RMSE of the estimators for three different windows. It can be noted that all estimators do not have an error floor and track the Cramer–Rao bound. From high SNR region plot, it can be more easily seen that the estimators are lined up according the incurred $\mathrm{SNR}_{\mathrm{loss}}^{\mathrm{windowing}}$ value.

An analytical expression for the MSE can be derived by following the approach in [2]. The black dashed lines in Fig. 5(a) show the analytical expression, given below, which is reached by treating $\widehat{\delta_1}$ as a random variable independent of noise.

$$\mathrm{MSE} = \frac{2c_N^2}{B_0^2\,\mathrm{SNR}}\sum_{n=0}^{N-1}\left|w[n]\sin\left(\frac{2\pi n}{N}\right)\right|^2 \tag{13}$$

For the rectangle window, $c_N = \frac{\tan(\pi/N)}{\pi/N} \approx 1$, $B_0^{(10)} = 2N$, the summation in (13) is $N/2$ and MSE becomes $\mathrm{MSE} = 1/(4\,\mathrm{SNR})$, which is identical to the expression in [2, Eq. (11)].

Fig. 5(b) shows a typical scenario illustrating the advantages of windowed operation. Here a second complex exponential signal is interfering with the desired signal. The power of interfering signal is 10 dB higher than the desired signal power, i.e interference-to-signal ratio (ISR) is 10 dB. The frequency of interfering signal is 4.5 bins higher than the signal of interest, i.e the frequency difference between desired and interfering signals is $2\pi \times 4.5/N$ radians/sample. Fig. 5(b) shows that SNR loss shown in Fig. 5(a) due to windowing is compensated with the suppression of the interfering signal and windowed operation yields better results at high SNR. In general, the trade-off between SNR loss due to windowing and suppression of interference depends on ISR, frequency of the interfering signal and operating SNR values. (Readers can examine [8, Chapter 6] for details.).

In the absence of interference (ISR = 0), the rectangular windowed data (no windowing) certainly yields the best results. For this case, the ratio of MSE, given in (13), and Cramer–Rao bound at high SNR is 1.68, 3.51 and 5.72 for rectangle, Hamming and Blackman windows, respectively. This loss is partially due to the SNR loss due to windowing and partially due to the estimator. We anticipate that the development of better frequency estimators for arbitrarily

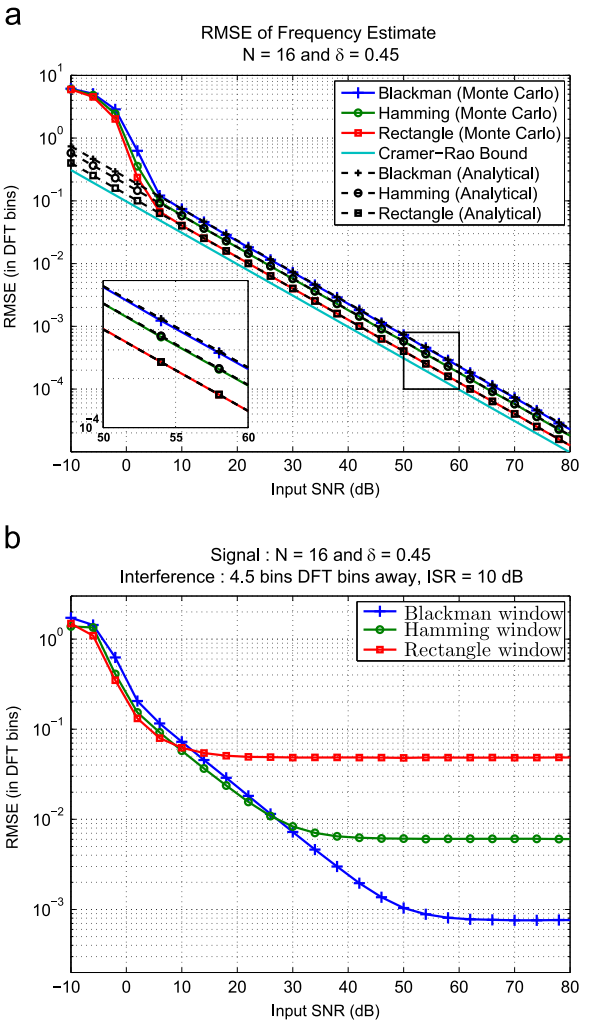

Fig. 5. Performance of the suggested scheme for three different windows. (a) Effect of windowing on RMSE in the absence of interference and (b) effect of windowing on RMSE in the presence of interference.

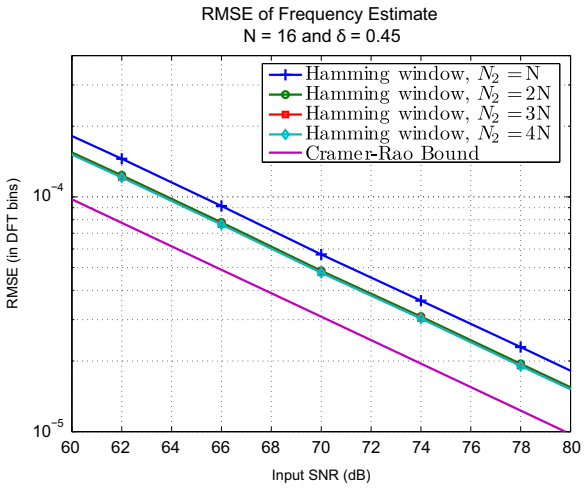

Fig. 6. RMSE for N_2-point DFT of Hamming windowed N-point input.

windowed data, as the Aboutanios and Mulgrew method for the rectangular window, in the near future.

Experiment 3: This experiment shows the effect of zero-padding, N_2-point DFT ($N_2 > N$), on the frequency estimation. As can be seen from Fig. 6, $2N$-point DFT with Hamming window is closer to the Cramer–Rao bound by 1 dB in comparison to the N-point DFT result. Higher point DFT calculations do not seem to yield any additional benefit. Hence, it can be concluded that $2N$-point DFT is sufficient to capture the best estimation performance of the suggested method.

5. Conclusions

The present study extends an earlier fine frequency estimation method, given in [2], limited to the rectangle windowed data to the case of windowing with or without zero-padding. Different from many methods in the literature, the suggested method is readily applicable for any window function and results in an estimation performance closely tracking the Cramer–Rao bound without any apparent bias, even at exceptionally high SNR values. A possible future work can be the inclusion of higher order δ terms in the estimator design to develop a non-iterative algorithm.

References

[1] C. Candan, A method for fine resolution frequency estimation from three DFT samples, IEEE Signal Process. Lett. 18 (2011) 351–354.
[2] C. Candan, Analysis and further improvement of fine resolution frequency estimation method from three DFT samples, IEEE Signal Process. Lett. 20 (2013) 913–916.
[3] B.G. Quinn, Estimating frequency by interpolation using Fourier coefficients, IEEE Trans. Signal Process. 42 (1994) 1264–1268.
[4] E. Aboutanios, B. Mulgrew, Iterative frequency estimation by interpolation on Fourier coefficients, IEEE Trans. Signal Process. 53 (2005) 1237–1242.
[5] E. Aboutanios, Estimation of the frequency and decay factor of a decaying exponential in noise, IEEE Trans. Signal Process. 58 (2010) 501–509.
[6] J.-R. Liao, C.-M. Chen, Phase correction of discrete Fourier transform coefficients to reduce frequency estimation bias of single tone complex sinusoid, Signal Process. 94 (2014) 108–117.
[7] U. Orguner, C. Candan, A fine-resolution frequency estimator using an arbitrary number of DFT coefficients, Signal Process. 105 (2014) 17–21.
[8] B. Porat, A Course in Digital Signal Processing, John Wiley & Sons, 1996.
[9] D. Belega, D. Petri, D. Dallet, Frequency estimation of a sinusoidal signal via a three-point interpolated DFT method with high image component interference rejection capability, Digit. Signal Process. 24 (2014) 162–169.
[10] D. Belega, D. Petri, Sine-wave parameter estimation by interpolated DFT method based on new cosine windows with high interference rejection capability, Digit. Signal Process. 33 (2014) 60–70.
[11] T. Grandke, Interpolation algorithms for discrete Fourier transforms of weighted signals, IEEE Trans. Instrum. Meas. 32 (1983) 350–355.
[12] G. Andria, M. Savino, A. Trotta, Windows and interpolation algorithms to improve electrical measurement accuracy, IEEE Trans. Instrum Meas. 38 (1989) 856–863.
[13] D. Agrez, Weighted multipoint interpolated DFT to improve amplitude estimation of multifrequency signal, IEEE Trans. Instrum Meas. 51 (2002) 287–292.
[14] K. Duda, DFT interpolation Algorithm for Kaiser–Bessel and Dolph–Chebyshev Windows, IEEE Trans. Instrum Meas. 60 (2011) 784–790.
[15] C. Candan, Fine Resolution Frequency Estimation From Three DFT Samples: Windowed Case, (MATLAB Code), 2014, URL: ⟨http://www.eee.metu.edu.tr/∼ccandan/pub.htm⟩.