A Method For Fine Resolution Frequency Estimation From Three DFT Samples

Signal-Processing Final Project

Team-30 (Litelo)
Abhishek Chawla (2019102020)
Pragya Singhal (2019112001)

Overview

The project suggests a nonlinear relation involving three DFT samples already calculated to produce a real valued, fine resolution frequency estimate. The estimator approaches Jacobsen's estimator for large N and presents a bias correction which is especially important for small and medium values of N .

Goals

1. Our goal in this project is to estimate δ , where $|\delta| < 1/2$, from three samples around the peak in the DFT spectrum using different estimators.
2. Finally we comment on the accuracy of various estimators and find the most suitable one!

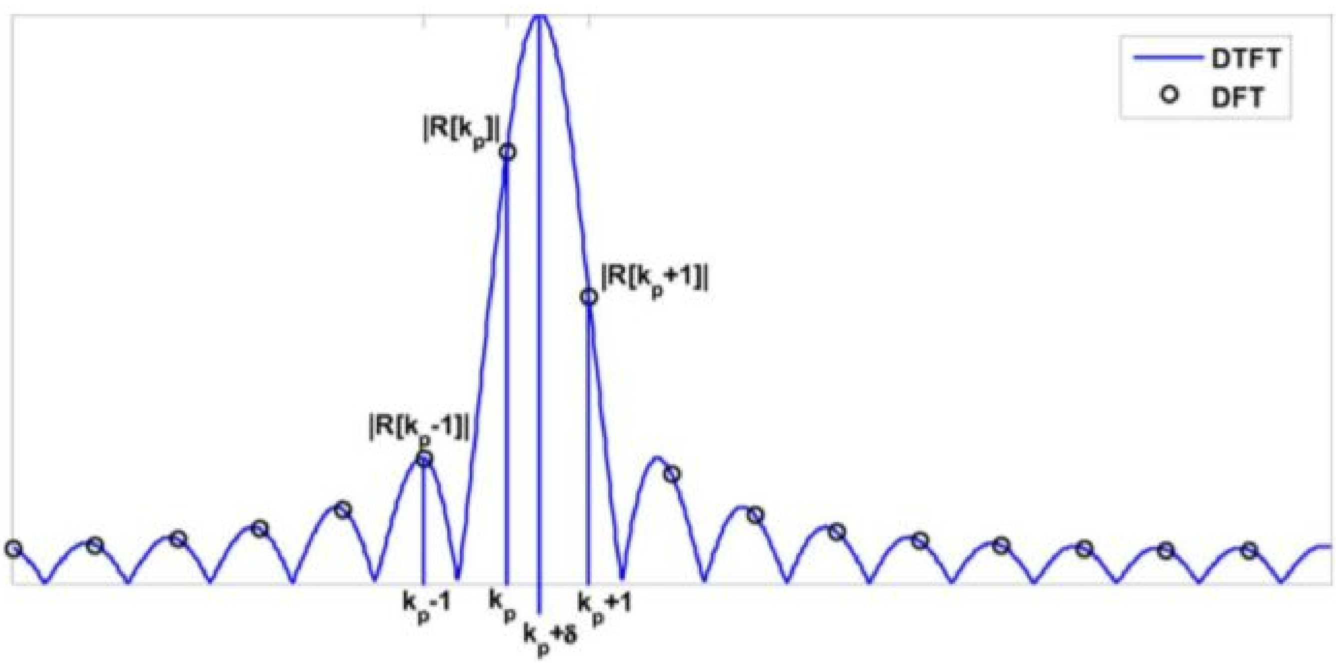
Problem Description

We are given a signal r[n] with the white gaussian noise(w[n]) as follows:

$$r[n] = Ae^{j\omega n} + w[n]$$

Our objective is to calculate the peak location from the given 3 DFT (R[k]) samples namely R[Kp-1], R[Kp] and R[Kp+1]. The peak location in the DFT magnitude spectrum is expected to be around the true frequency ω if the SNR (Signal to noise ratio = $A^2/^2$) is sufficiently large. Hence we assume the below calculations to be done for high SNR only.

The Magnitude spectrum of r[n] for the noiseless case is shown below:



Calculation of δ

We use the three DFT samples given to us to calculate the value of **δ**, after which we can calculate the peak location by the formula (Kp + **δ**).

Given: ω = (2pi/N)(Kp + **δ**),   where N = size of the DFT

Now we have the following DFT-

$$
\begin{aligned}
R[k_p-1] &= A\sum_{n=0}^{N-1} e^{j\frac{2\pi}{N}(\delta+1)n} + \tilde{w}[k-1] \\
&= Af(\delta+1) + \tilde{w}[k_p-1] \\
R[k_p] &= A\sum_{n=0}^{N-1} e^{j\frac{2\pi}{N}\delta n} + \tilde{w}[k] \\
&= Af(\delta) + \tilde{w}[k_p] \\
R[k_p+1] &= A\sum_{n=0}^{N-1} e^{j\frac{2\pi}{N}(\delta-1)n} + \tilde{w}[k+1] \\
&= Af(\delta-1) + \tilde{w}[k_p+1].
\end{aligned}
$$

Here $\tilde{w}[k]$ is the DFT of $w[n]$ which is also white and jointly Gaussian distributed.

The function $f(\cdot)$ appearing on the right hand side of the equations given in (1) is defined as follows:

$$f(\alpha) = \sum_{n=0}^{N-1} e^{j\frac{2\pi}{N}\alpha n}.$$

Here α is a generic variable for the function $f(\cdot)$.

Error in calculating δ

There are a total of 4 equations that are derived for the calculation of δ. **According to the paper - E. Jacobsen and P. Kootsookos, "Fast, accurate frequency estimators,"** they are labelled as error 2, error 3, error 4, error 5 respectively.

$$\delta = \frac{(|X_{k+1}| - |X_{k-1}|)}{(4|X_k| - 2|X_{k-1}| - 2|X_{k+1}|)}. \quad (2)$$

Equation 2- It is a poor estimation with a high error (specially in case with noise). Some simple changes in this equation increases the accuracy and produces the third equation.

$$\delta = -\mathrm{Re}\left[\frac{(X_{k+1} - X_{k-1})}{(2X_k - X_{k-1} - X_{k+1})}\right]. \quad (3)$$

Equation 3- It is very accurate especially in case of rectangular windows. However, for non-rectangular cases we have the below equations.

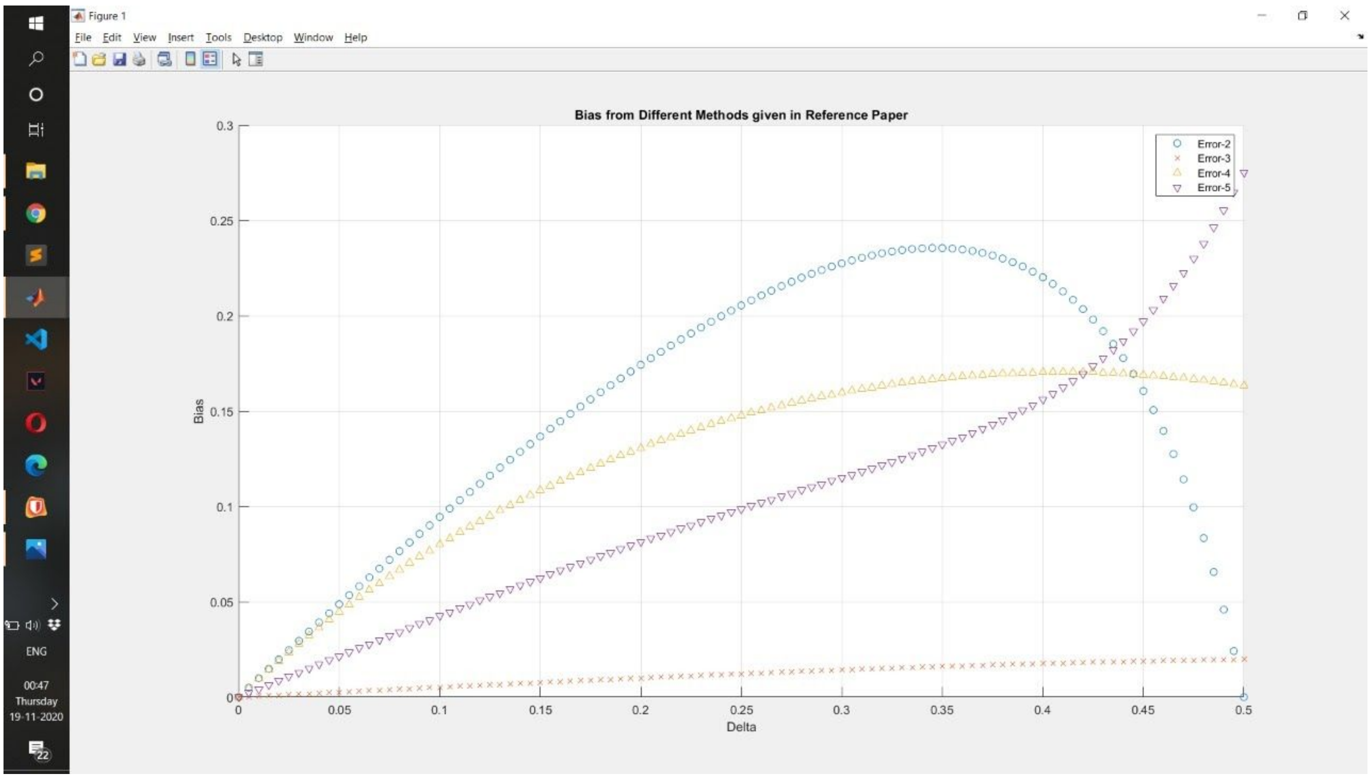
$$\delta = \frac{P(|X_{k+1}| - |X_{k-1}|)}{(|X_k| + |X_{k-1}| + |X_{k+1}|)}, \quad (4)$$

$$\delta = \mathrm{Re}\left[\frac{Q(X_{k-1} - X_{k+1})}{(2X_k + X_{k-1} + X_{k+1})}\right], \quad (5)$$

Equation 4 and 5- They are less accurate than the third one and give more biased outputs but are more accurate when applied to DFT samples of non-rectangular windowed data.

Plot

Below is the plot for different errors :-



Conclusion:- Error 3 is the least and hence the most accurate one. It gives the least biased outputs as compared to other equations. It is observed that **equation 3 is similar to the Jacobsen estimator** (it will come later in this report). Error 2 is the poorest of all.

Different Estimators

Next we apply different estimators as given in the research paper to find out the most accurate value for δ for the best estimation.

After performing all the calculations we arrive with the following equations for different estimators :

I. Parabolic Interpolation- This is the poorest estimation

$$\widehat{\delta} = (|R_{k+1}| - |R_{k-1}|)/(4|R_k| - 2|R_{k-1}| - 2|R_{k+1}|)$$

II. Quinn

$$\begin{array}{ll}
\alpha_1 = \text{Real}(R_{k-1}/R_k), \quad \alpha_2 = \text{Real}(R_{k+1}/R_k) & \\
\delta_1 = \alpha_1/(1-\alpha_1), \quad \delta_2 = \alpha_2/(1-\alpha_2) & \\
\text{if } \delta_1 > 0 \text{ and } \delta_2 > 0, \ \widehat{\delta} = \delta_2 & \\
\text{else} \qquad\qquad\qquad \widehat{\delta} = \delta_1 &
\end{array}$$

III. MacLeod

$$\begin{array}{l}
d = \text{Real}(R_{k-1}R_k^* - R_{k+1}R_k^*)/\text{Real}(2|R_k|^2 + R_{k-1}R_k^* + R_{k+1}R_k^*) \\
\widehat{\delta} = (\sqrt{1+8d^2} - 1)/(4d)
\end{array}$$

IV. Jacobsen

$$\widehat{\delta} = \text{Real}\{(R_{k-1} - R_{k+1})/(2R_k - R_{k-1} - R_{k+1})\}$$

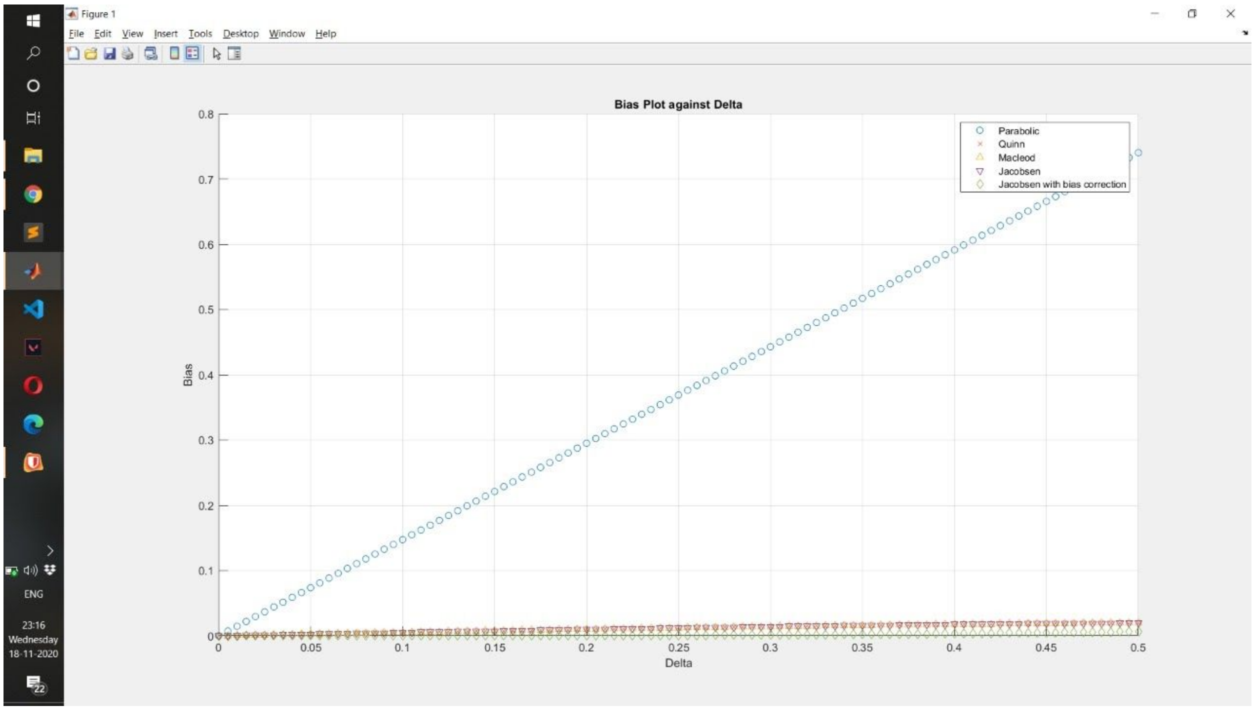
The estimators Quinn, MacLeod, Jacobsen have exactly the same bias in the absence of the noise.

V. Jacobsen with Bias Correction (Proposed) - This is the least biased one

$$\widehat{\delta} = \frac{\tan(\pi/N)}{\pi/N}\text{Real}\{(R_{k-1} - R_{k+1})/(2R_k - R_{k-1} - R_{k+1})\}$$

Plots

Below is the plot for bias of different estimators in the absence of noise-

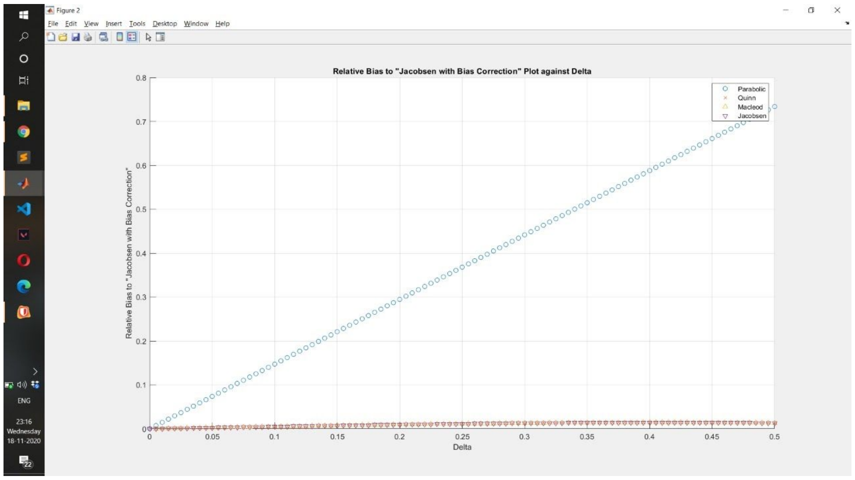


The bias basically describes the error in calculating **δ**, hence more the bias - more will be the error in calculating **δ** - less accurate will be the estimation!

Conclusion:- It is very clear from the above plot that the **Proposed estimator is the most efficient one,** with the least bias (for every N) and the **Parabolic is the poorest** with a high bias. While **Quinn, MacLeod and Jacobsen have the exact same bias!**

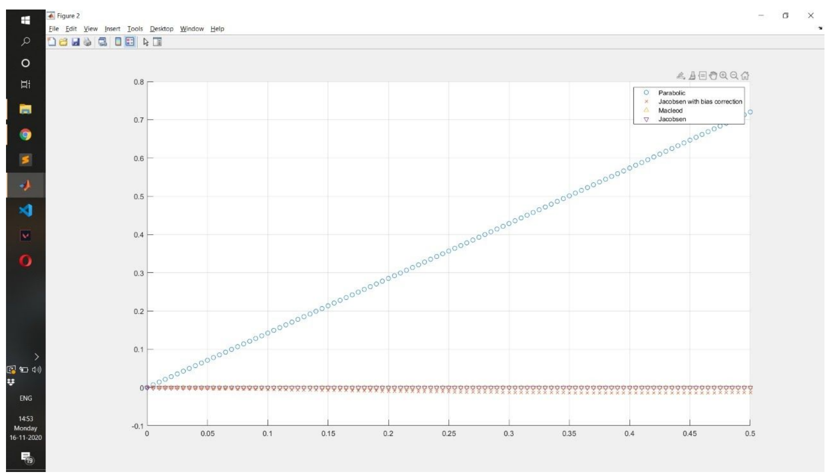
Also the bias decreases as N tends to infinity or SNR increases. However, the above plot is plotted for N=8 and since this is a non-linear estimation we have a positive bias for all the estimators (even for the proposed one).

To have a better look at this, we plotted the relative bias-



This plot is plotted with respect to “The Proposed Estimator”. This shows that the remaining estimators have a positive bias relative to the proposed one and hence they have a higher bias than the proposed and hence are less accurate.

This plot is plotted with respect to “The Quinn Estimator”. As one can see the proposed estimator has a negative relative bias while the parabolic has a positive relative bias. The jacobsen and macleod have a zero relative bias.



Hence the Final Conclusion is -

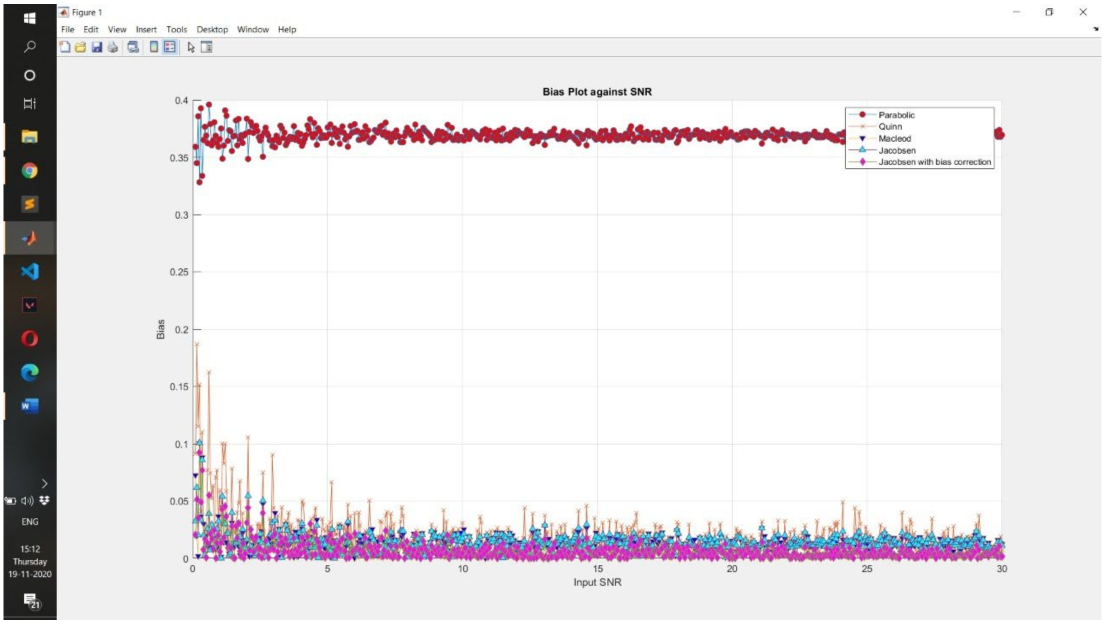
Bias - Proposed < Jacobsen = MacLeod = Quinn < Parabolic

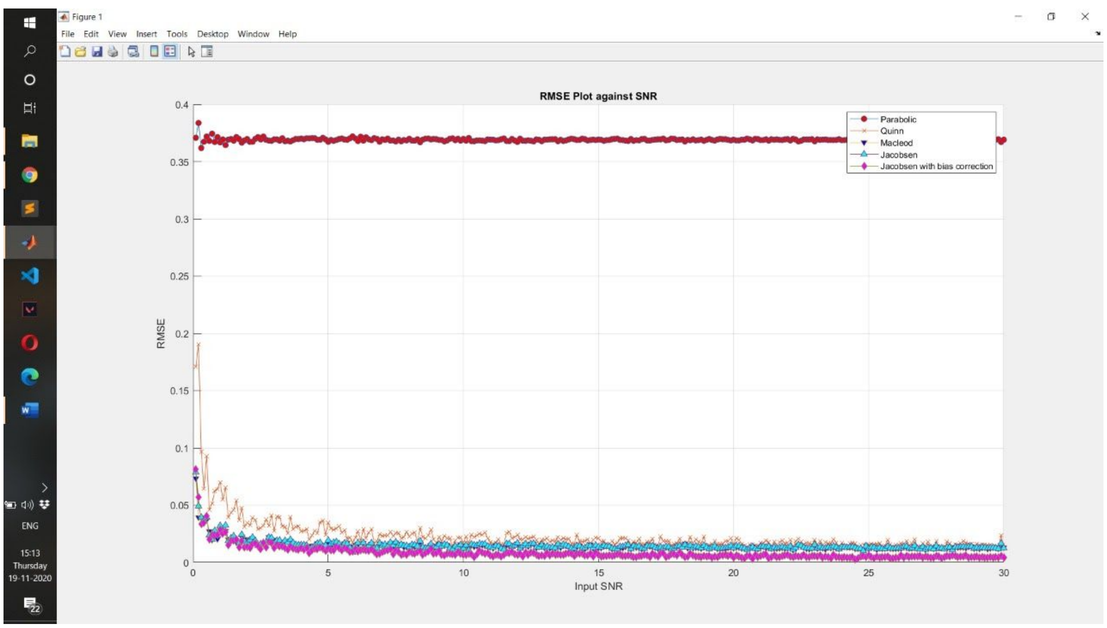
Accuracy - Proposed > Jacobsen = MacLeod = Quinn > Parabolic

***This is for noiseless case (very high SNR).**

In the presence of noise

Here, we fix the value of δ=0.25 and vary the SNR value to examine the bias in presence of noise.





Conclusion:- The plots clearly show that as the value of SNR increases, the Bias as well as RMSE move towards the ideal (noiseless) case.

Final Conclusions

- The value of **δ** can be best calculated by the third equation in case of rectangular windowed data (from the reference paper) as it has the least error and gives less biased outputs.
- The best estimation is done with the help of the Jacobsen which is improved with bias correction (Proposed estimator) among all the estimators (for every N in general).

Bias - Proposed < Jacobsen = MacLeod = Quinn < Parabolic

Accuracy - Proposed > Jacobsen = MacLeod = Quinn > Parabolic

- The Proposed estimator closely follows the Cramer-Rao bound (least error) in the high SNR region.
- When N is large, Jacobsen's formula and its bias corrected version have the same performance for a large SNR range.
- In the presence of noise, we observe that the error decreases as SNR is increased.
- After calculation of **δ**, we can estimate the location of the peak value as follows:

Peak location = (Kp + **δ**)

Special Note : The present work utilizes the data processed by a rectangular window.

References

1. E. Jacobsen and P. Kootsookos, "Fast, accurate frequency estimators,"
2. B. G. Quinn, "Estimating frequency by interpolation using Fourier coefficients,"
3. B. G. Quinn, "Estimation of frequency, amplitude, and phase from the DFT of a time series,"
4. https://matlab.mathworks.com/
5. https://www.wikipedia.org/