PERFORMING LOWPASS, HIGHPASS, BANDPASS AND BANDSTOP FILTERING OF INPUT AUTO AND CROSS CORRELATION SEQUENCES IN LINEAR TIME

Ejaz A. Ansari*
Department of Electrical Engineering, COMSATS Institute of Information Technology
1.5 km Defense Road, off Raiwind Road, 53700, Lahore, Pakistan
**corresponding author's email: {dransari@ciitlahore.edu.pk}*

ABSTRACT: Performing filtering on the input and output data sequences through linear time invariant (*LTI*) systems formulates an important operation of ***Digital Signal*** and ***Image Processing*** fields. This paper proposes a ***simple*** and ***efficient*** technique in discrete-time domain which does the lowpass, highpass, bandpass and bandstop filtering of the input auto and cross correlation sequences of discrete-time real wide sense stationary (*WSS*) process through ***exponential, rectangular, triangular*** and ***trapezoidal*** shaped windows in ***linear*** time. As the frequency responses of the discrete-time windows mentioned above ***resemble*** very much with the frequency response of low pass filters, the proposed algorithm [5] does indeed perform the *Low Pass Filtering* (***LPF***) of the input auto and cross correlation sequences in ***linear*** time. We have already shown in [5] that processing or ***LPF*** of auto and cross correlation sequences through the above four mentioned windows can be achieved in $O(L)$ time, where L denote the size of the input autocorrelation sequence. From the knowledge of impulse response of low pass filter, $h_{lp}[n]$, the impulse response of a high pass filter $h_{hp}[n]$ may be obtained by subtracting $h_{lp}[n]$ from the unit sample sequence, $\delta[n]$. This shows that high pass filtering (***HPF***) of the input auto and cross correlation sequences can also be performed through above four windows in linear time. Finally, the known values of impulse responses of low and high pass filters determine the impulse and frequency responses of a band pass filter, $h_{bp}[n]$. In frequency domain, it is usually implemented with a cascade connection of low and high pass filter's frequency responses with different cut-off frequencies. However, in time domain, $h_{bp}[n]$ becomes equal to the convolution of $h_{lp}[n]$ with $h_{hp}[n]$. Moreover, like $h_{hp}[n]$, the impulse response of band stop filter, $h_{bs}[n]$ is simply equal to the difference of $\delta[n]$ and $h_{bp}[n]$. This reveals that bandpass filtering (***BPF***) and bandstop filtering (***BSF***) operations on the input auto and cross correlation sequences can also be done through above four windows in linear time. We have also shown in [5] that our proposed algorithm ***outperformed*** all the existing techniques in ***both*** time and frequency domains with regards to both ***exact*** number of arithmetic operations and to its ***worst case*** time complexity that grew ***linearly*** with the length of the input auto correlation sequence for the case when $L >> K$, where K denote the size of an exponential window.

Key Words: Auto correlation, Cross correlation, Efficient Filtering, Finite shaped Windows, Time Complexity

1 INTRODUCTION

There are plenty of application examples that require the filtering operation in certain kind, for example in the area of medical image and video processing, radar and sonar signal processing, statistical signal processing and modern communication systems employing coherent receivers [1,2]. Convolution of input data sequence through an *FIR* linear shift invariant system (*LSI*) basically implements both filtering and smoothing operations on the input data. Although, convolution in time domain contains no processing delays but the required computational cost using convolution sum makes it highly impractical. Thus, it is usually implemented in frequency domain using *FFT/IFFT* based overlap-add and overlap-save methods. However, efficiency is achieved at the expense of the delay introduced equal to the filter length [3,5].

The processing of discrete-time wide sense stationary (*WSS*) random processes through *FIR LSI*-systems plays a very useful role in studying the behavior of many wireless communication systems and networks. Filtering operation does not bring about any change in stationarity status of the input process and thus both input and output discrete-time processes become jointly stationary [5-9]. Frequency responses of all discrete-time windows discussed in this paper correspond to the frequency response of low pass filters [6-9].

The use of low pass filtering eliminates the aliasing phenomenon in *A/D* conversion systems due to creation of false lower frequencies when the input signal contains frequency components above half *A/D* sampling rate. A low pass filter when applied to each input channel of *A/D* converter card also eliminates unwanted high frequency noise and interference introduced prior to sampling. This reduces system cost, acquisition storage requirements and analysis time by allowing for a lower sampling rate. It is also a well known fact that low pass filtering of *WSS* input processes brings smoothness, removes noise buried in the uncorrupted input data and blurness present in the underlying image respectively, etc [6-8].

The high pass filter is good for removing a small amount of low frequency noise from a multi-dimensional signal. However, a band pass filter is useful when it is required to eliminate the noise at low and high frequency components of the input signal. Similarly, a bandstop filtering is needed when it is desired to eliminate the unwanted noise from the desired signal within certain band of frequencies [6–8]. To the best of our knowledge, there exists no algorithm so far in the literature either in time or frequency domain that performs lowpass, highpass, bandpass and bandstop filtering of the input auto and cross correlation sequences through *FIR* shaped windows in linear time. We have already discussed the basic idea of the proposed algorithm in discrete time domain in [5]. In this paper, we, thus present a very simple and efficient method in time domain for performing lowpass, highpass, bandpass and bandstop filtering of the auto and cross-correlation sequences through *FIR LTI*-windows. The various types of discrete-time *FIR* windows discussed in this paper are exponential, rectangular (moving average), triangular and trapezoidal shaped ones.

This paper is organized as follows*: Section 2 computes the frequency responses of the four windows discussed in the paper. Section 3 describes the system block diagram that implements the lowpass, highpass, bandpass and bandstop filtering of auto and cross correlation sequences through these windows using the proposed algorithm [5]. Worst case time complexity for four types of filtering operations is discussed in section 4. Variation of magnitude frequency responses of the four windows with regards to their sizes is discussed in section 5. Comparison of various filtering operations with other existing algorithms is discussed in section 6. Finally, we present our conclusions in section 7.*

2 COMPUTATION OF FREQUENCY RESPONSES OF THE FOUR WINDOWS IN DISCRETE-TIME DOMAIN

a) For an exponential window having 'K' samples

An exponential window, $exp[n]$ having K samples can be defined as $exp[n] = a^n$ for $0 \le n \le K - 1$ and zero otherwise, where $|a| < 1$. In terms of unit step sequences, this window can be expressed as [6], $exp[n] = a^n(u[n] - u[n - K])$. Its frequency response, $Exp(e^{jw})$ may be computed from its z-transform tables and properties [6] – [8] as given below:

$$exp[n] = a^n \left(u[n] - u[n-K]\right) \quad and \quad |a| < 1$$

$$a^n u[n] \leftrightarrow \frac{1}{1 - az^{-1}}; \; |z| > |a|$$

$$a^{n-K} u[n-K] \leftrightarrow \frac{z^{-K}}{1 - az^{-1}}; \; |z| > |a|;$$

$$Using \; linearity \; property, we \; have$$

$$Exp(z) = \frac{1}{1 - az^{-1}} - a^K \times \frac{z^{-K}}{1 - az^{-1}}; \; |z| > |a|$$

$$= \frac{1 - a^K z^{-K}}{1 - az^{-1}}$$

$$Exp\left(e^{jw}\right) = Exp(z)\big|_{z = e^{jw}}$$

$$= \frac{1 - a^K e^{-jwK}}{1 - ae^{-jw}} \Leftarrow Frequency \; Response$$

$$Exp^*\left(e^{jw}\right) = \frac{1 - a^K e^{+jwK}}{1 - ae^{+jw}} = Exp\left(e^{-jw}\right)$$

$$\left|Exp\left(e^{jw}\right)\right|^2 = Exp\left(e^{jw}\right) \times Exp^*\left(e^{jw}\right)$$

$$= \left(\frac{1 - a^K e^{-jwK}}{1 - ae^{-jw}}\right) \times \left(\frac{1 - a^K e^{+jwK}}{1 - ae^{+jw}}\right)$$

$$= \left(\frac{\left(1 + a^{2K}\right) - 2a^K \cos(Kw)}{\left(1 + a^2\right) - 2a \cos w}\right)$$

$$\left|Exp\left(e^{jw}\right)\right| = \sqrt{\frac{\left(1 + a^{2K}\right) - 2a^K \cos(Kw)}{\left(1 + a^2\right) - 2a \cos w}}$$

Now, for large K and $|a| < 1$, *the magnitude response reduces to*

$$\left|Exp\left(e^{jw}\right)\right| \approx \frac{1}{\sqrt{\left(1 + a^2\right) - 2a \cos w}}; \; -\pi \le w \le \pi; \qquad (1)$$

Equation (1) describes the magnitude response of an exponential window under the condition of $|a| < 1$ and for large value of K, *i.e.*, sufficient number of samples of the window.

- ***For a moving average / rectangular window having 'K' samples***

Moving average window, $mov[n]$ containing K samples in terms of unit step sequences may be defined as [6], $mov[n] = K^{-1}(u[n] - u[n - K])$ and zero otherwise. Its frequency response, $Mov(e^{jw})$ can also be computed from its z-transform as shown below:

$$mov[n] = K^{-1}\left[u[n] - u[K]\right];$$

$$\Updownarrow \; using \; linearity \; and \; Tables$$

$$Mov(z) = K^{-1}\left[\frac{1}{1 - z^{-1}} - \frac{1}{1 - z^{-1}} \times z^{-K}\right]$$

$$= K^{-1}\left[\frac{1 - z^{-K}}{1 - z^{-1}}\right] = K^{-1} \times \frac{1}{z^{K-1}} \times \left(\frac{z^K - 1}{z - 1}\right);$$

$$RoC \; of \; Mov(z) \equiv entire \; z - plane \; except \; z = 0$$

The above expression reveals that z-transform of moving average window contains one pole of $(K - 1)^{th}$ order at $z = 0$ place (*i.e.*, origin) and $(K - 1)$ zeros lying on the boundary of unit circle with angular spacing of $exp(j2\pi/K)$ between any two adjacent zeros. Due to cancellation of one pole with one zero at $z = 1$ location, Region of Convergence (*RoC*) of the transform consists of entire z-plane except its origin where the pole is located. For z-transform of rectangular window, $rect[n]$ it is suggested that the scaling factor, K is made equal to unity. Thus, we can obtain frequency response, $Rect(e^{jw})$ of the rectangular window using the above result as,

$$Rect(z) = Mov(z)\big|_{K=1} = \frac{1}{z^{K-1}} \times \left(\frac{z^K - 1}{z - 1}\right);$$

$$Rect\left(e^{jw}\right) = Rect(z)\big|_{z = e^{jw}} = \frac{1 - e^{-jwK}}{1 - e^{-jw}}$$

$$= \left(\frac{e^{-jwK/2}}{e^{-jw/2}}\right) \times \left(\frac{e^{+jwK/2} - e^{-jwK/2}}{e^{+jw/2} - e^{-jw/2}}\right);$$

$$Rect\left(e^{jw}\right) = \left(\frac{\sin(wK/2)}{\sin(w/2)}\right) \times e^{-jw/2(K-1)}$$

$$\left|Rect\left(e^{jw}\right)\right| = \left|\left(\frac{\sin(wK/2)}{\sin(w/2)}\right)\right|; \qquad (2)$$

$$and \; \angle Rect\left(e^{jw}\right) = -0.5w\,(K - 1)$$

Equation (2) provides the frequency response of a rectangular window having K samples. We also notice from eqs. (1) and (2) that exponential window, $exp[n]$ reduces to rectangular window, $rect[n]$ by making $a = 1$ and the same expression for magnitude of Fourier Transform given in eq. (2) can be obtained by setting $a = 1$ in eq. (1) alongwith the application of some useful calculus results.

b) For a Triangular window having (2K+1) samples

Triangular window, $tri[n]$ having $(2K + 1)$ samples can be defined as [6], $tri[n] = n$ for $0 \leq n \leq K$ and $tri[n] = 2K - n$ for $K+1 \leq n \leq 2K$ and zero otherwise. We also know from the discussion of Signals and Systems Course that a triangular window can be fabricated from the convolution of two rectangular windows having the same number of samples, *i.e.*, the width. Therefore, we may express $tri[n]$ in the form of a relation with $rect[n]$ as,

$$tri[n] = \begin{cases} n; & for\ 0 \leq n \leq K \\ 2K - n; & for\ K+1 \leq n \leq 2K = rect[n] * rect[n-1]; \\ 0; & otherwise \end{cases}$$

$$\updownarrow$$

$$Tri(z) = Rect(z) \times Rect(z) \times z^{-1} = \left(Rect(z)\right)^2 \times z^{-1};$$

$$Tri\left(e^{jw}\right) = Tri(z)\Big|_{z=e^{jw}} = \left(Rect\left(e^{jw}\right)\right)^2 \times e^{-jw}$$

$$= \left(\left(\frac{sin(wK/2)}{sin(w/2)}\right) \times e^{-jw/2(K-1)}\right)^2 \times e^{-jw} = \left(\frac{sin(wK/2)}{sin(w/2)}\right)^2 \times e^{-jwK};$$

$$\left|Tri\left(e^{jw}\right)\right| = \left|\frac{sin(wK/2)}{sin(w/2)}\right|^2; and\ \angle\ Tri\left(e^{jw}\right) = \text{-}wK \qquad (3)$$

Equation (3) describes the frequency response of triangular shaped window having $(2K + 1)$ samples. On comparison of eq. (3) with eq. (2), we conclude that its frequency response is simply the square of the frequency response of rectangular shaped window having K samples.

c) For a Trapezoidal window having (2K+1) samples

Like triangular shaped window, a trapezoidal shaped window, $trap[n]$ having $(2K + 1)$ samples can also be defined as [6], $trap[n] = n$ for $0 \leq n \leq K_1$, $trap[n] = K_1$ for $K_1 + 1 \leq n \leq K_2 - K_1$, $trap[n] = 2K - n$ for $K_2 - K_1+1 \leq n \leq 2K$ and zero otherwise, where $K = K_1 + K_2$. We also know from the discussion of Signals and Systems course that a trapezoidal shaped window can also be fabricated from the convolution of two rectangular windows having different number of samples, *i.e.*, the widths. Therefore, we may express $trap[n]$ in the form of a relation with $rect[n]$ as,

$$trap[n] = \begin{cases} n; & for\ 0 \leq n \leq K_1 \\ N_1; & for\ K_1 + 1 \leq n \leq K_2 - N_1 \\ 2K - n; & for\ K_2 - K_1 + 1 \leq n \leq 2K\ \ \&\ \ 0;\ otherwise \end{cases}$$

$$= rect_1[n] * rect_2[n-1]; where\ K = K_1 + K_2\ \&\ K_2 > K_1$$

$$\updownarrow$$

$$Trap(z) = Rect_1(z) \times Rect_2(z) \times z^{-1} = \left(Rect_1(z)\right) \times \left(Rect_2(z)\right) \times z^{-1};$$

$$Trap\left(e^{jw}\right) = Trap(z)\Big|_{z=e^{jw}} = \left(Rect_1\left(e^{jw}\right)\right) \times \left(Rect_2\left(e^{jw}\right)\right) \times e^{-jw}$$

$$= \left(\frac{sin(wK_1/2)}{sin(w/2)}\right) \times e^{-j0.5w(K_1-1)} \times \left(\frac{sin(wK_2/2)}{sin(w/2)}\right) \times e^{-j0.5w(K_2-1)} \times e^{-jw}$$

$$= \underbrace{\left(\frac{sin(wK_1/2)}{sin(w/2)} \times \frac{sin(wK_2/2)}{sin(w/2)}\right)}_{\left|Trap\left(e^{jw}\right)\right|} \times e^{-j0.5w(K_1+K_2)} \qquad (4)$$

We realize that for $K_1 = K_2$, the equation (4) reduces to

$$Trap\left(e^{jw}\right) = Tri\left(e^{jw}\right) for\ K_1 = K_2, thus\ magnitude\ and\ phase\ responses$$
$$in\ case\ of\ this\ window\ reduce\ to:$$

$$\left|Trap\left(e^{jw}\right)\right| = \left|\left(\frac{sin(wK_1/2)}{sin(w/2)} \times \frac{sin(wK_2/2)}{sin(w/2)}\right)\right|\ and \qquad (5)$$

$$\angle\ Trap\left(e^{jw}\right) = -0.5w(K_1 + K_2)$$

Equation (4) describes the frequency response of trapezoidal shaped window having $(2K + 1)$ samples. On comparison of eq. (5) with eq. (3), it can easily be concluded that its frequency response simply reduces to the frequency response of triangular shaped window already computed in eq. (3) for the case when $K_1 = K_2$.

3 IMPLEMENTING LOWPASS, HIGHPASS, BANDPASS AND BANDSTOP FILTERING OF AUTO AND CROSS CORRELATION SEQUENCES USING THE PROPOSED ALGRITHM IN [5]

Figure 1 shows the block diagram of our system for implementing lowpass, highpass, bandpass and bandstop filtering operations using the proposed algorithm [5]. In this figure, $R_X[n]$, $R_{XY}[n]$, $h[n]$ and $\delta[n]$ denote the input auto and cross correlation sequences, impulse response of the window under consideration and unit sample sequence respectively. The lower sub system shown in this figure implements lowpass and highpass filtering operations on the input auto and cross correlation sequences. While the upper sub system (which is exact replica of the lower sub system) implements bandpass and bandstop filtering operations of the input auto and cross correlation sequences. These four filtering operations are clearly shown in figure 1 as the outputs of the adders.

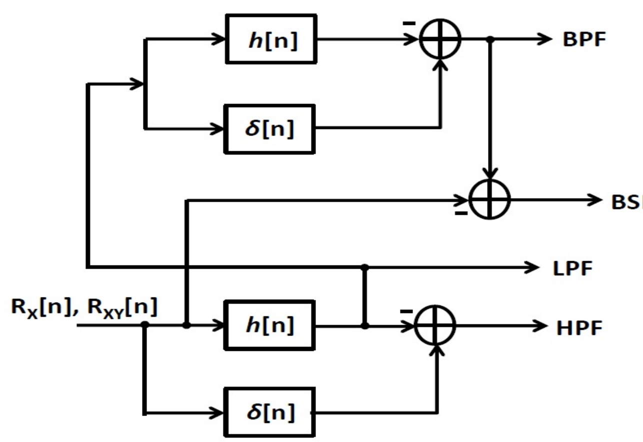

Figure 1: System Block Diagram implementing four types of filtering operations using the Proposed Algorithm[5]

The plot of magnitude responses of the four windows computed in eqs. (1) – (5) are sketched in section 5. It can easily be seen from their sketches (figure 2 – 5) that these graphs resemble very much with the magnitude responses of the low pass filters. It was shown in [5, sections 3, 4 and 5] how cross and auto correlation of the output discrete-time real *WSS* process can be obtained from processing of the input auto correlation sequence having L samples through four types of the windows discussed in the paper. It was also shown in all cases that under the condition when $L >> K$, the worst case time complexity of the proposed algorithm varies linearly with the size of the input auto correlation sequence, *i.e.*, ***O***(L). This shows that low pass filtering (***LPF***) of the input auto and cross correlation sequences can be implemented in linear time using the proposed algorithm [5] as shown in figure 1.

Likewise, we can also perform highpass (***HPF***), bandpass (***BPF***) and bandstop (***BSF***) filtering operations of auto and cross correlation sequences through the same four types of windows by utilizing the facts of [6] as:

$h_{hp}[n] = \delta[n] - h_{lp}[n]$, $h_{bp}[n] = h_{lp}[n] * h_{hp}[n]$ and $h_{bs}[n] = \delta[n] - h_{bp}[n]$, where $\delta[n]$, $*$, $h_{hp}[n]$, $h_{bp}[n]$, $h_{bs}[n]$ and $h_{lp}[n]$ denote the unit sample sequence, discrete convolution operation and impulse responses of highpass, bandpass, bandstop and the four types of windows discussed in the paper respectively.

Figure 1 depicts that highpass (***HPF***) filtering can also be accomplished by subtracting the output sequences obtained after performing lowpass filtering (***LPF***) operation from the input auto and cross correlation sequences. Similarly, bandpass filtering (***BPF***) can also be done conveniently by performing the lowpass filtering of the input auto and cross correlation sequences followed by highpass filtering with the impulse response of the high pass filter as demonstrated in figure 1. Finally, bandstop filtering (***BSF***) can also be implemented by taking the output from an adder shown in figure 1 which does the subtraction of the output sequences obtained after performing bandpass filtering (***BPF***) from the input auto and cross correlation sequences.

4 DETERMINING THE WORST CASE TIME COMPLEXITY OF LOW, HIGH AND BAND PASS FILTERING USING THE ALGORITHM IN [5]

We have shown in [5, section 5] that under the assumption, when the length of input auto correlation sequence becomes very large than the size of the window, (*i.e.*, when ***L >>K***), the ***worst case*** total time complexity of the proposed algorithm according to [10], just reduces to ***O***(***L***) which shows that it ***linearly*** grows with the length of input auto correlation sequence, $R_{XX}(k)$. This shows that lowpass filtering (***LPF***) of input auto and cross correlation sequences can be performed in linear time using the proposed algorithm [5]. Similarly, in order to perform highpass filtering (***HPF***) of the cross and auto correlation sequences, we thus require L more subtraction operations in addition to the worst case time complexity of the proposed algorithm [5]. We know from [10] that ***O***(L) + ***O***(L) = ***O***(L) for large L. This shows that our proposed algorithm in [5] also does perform the highpass filtering (***HPF***) of auto and cross correlation sequences in linear time.

Likewise, bandpass filtering (***BPF***) of the input auto and cross correlation sequences using the proposed algorithm of [5] can be implemented by first performing the lowpass filtering operation followed by the highpass filtering of the input auto and cross correlation sequences. The worst case time complexity of bandpass filtering operation (***BPF***) using the proposed algorithm may be computed by summing the worst case time complexities taken by the proposed algorithm in doing the lowpass and the highpass filtering operations of the input auto and cross correlation sequences. Thus, we compute the worst case time complexity for band pass filtering (***BPF***) = worst case time complexity for low pass filtering (***LPF***) + worst case time complexity for high pass filtering (***HPF***) = ***O***(L) + ***O***(L) = ***O***(L) from [10] in case of large L. Similarly, we require L more subtraction operations in addition to worst case time complexity in order to implement bandstop filtering (***BSF***). This shows that our proposed algorithm described in [5] does indeed perform the lowpass, highpass, bandpass and bandstop filtering operations of the input auto and cross correlation sequences in linear time under the worst case scenario.

5 SKETCHING OF MAGNITUDE RESPONSES OF THE FOUR WINDOWS AS A FUNCTION OF 'K'

In this section, we provide the plots of magnitude responses of the four windows discussed in the paper as a function of its width, K and the constant, a. Figure 2 shows the plot of the magnitude response of an exponential window, $exp[n]$ as a function of the constant, a for sufficient number of samples of the window, *i.e.*, preferably $K \geq 100$. The value of the magnitude response at $w = 0$ changes from 1.33 to 20 (1400%) when the value of the constant, a is increased from 0.25 to 0.95 (280%) progressively. This can easily be inferred from eq. (1) that magnitude of frequency response at $w = 0$ just reduces to $1/(1-a)$ and thus increasing the value of the constant, a towards unity strengthens the value of the magnitude response of this window at $w = 0$.

Figure 3 describes the magnitude of the frequency response in case of rectangular window as a function of its width, K. The value of the magnitude response at $w = 0$ is just equal to K (computed from L Hospital's Rule) and the width of the main lobe is simply equal to ($2\pi / K$). It can easily be seen from the figure that smaller the value of K indicates less height of the main lobe but points towards its wider width as compared to higher values of K which is obviously a trade off in designing all types of low pass filters.

Figure 4 describes the magnitude of the frequency response in case of triangular window as a function of its width, K. The value of the magnitude response at $w = 0$ is simply equal to K^2 (again computed from L Hospital's Rule) and the width of the main lobe is just equal to ($2\pi / K$, same as that of fig. 3). It can again easily be seen from the figure that smaller the value of K indicates less height of the main lobe but points towards its wider width as compared to higher values of K. The comparison of figure 4 with that of figure 3 reveals that its magnitude response is simply equal to the square of the magnitude response of the rectangular window shown in fig. 3.

Figure 5 describes the magnitude response in case of trapezoidal window as a function of its width, $K = K_1 + K_2$. The value of the magnitude response at $w = 0$ is equal to the product of K_1 and K_2 (again computed from L Hospital's Rule) and the width of the main lobe is simply equal to ($2\pi/LCM(K_1, K_2)$), where $LCM(.)$ denote the least common multiple of the two numbers. It can again easily be seen from the figure that smaller the value of K indicates less height of the main lobe but points towards its wider width as compared to higher values of K. The comparison of fig. 5

with fig.4 reveals that its magnitude response simply reduces to that of triangular window for the case when $K_1=K_2$.

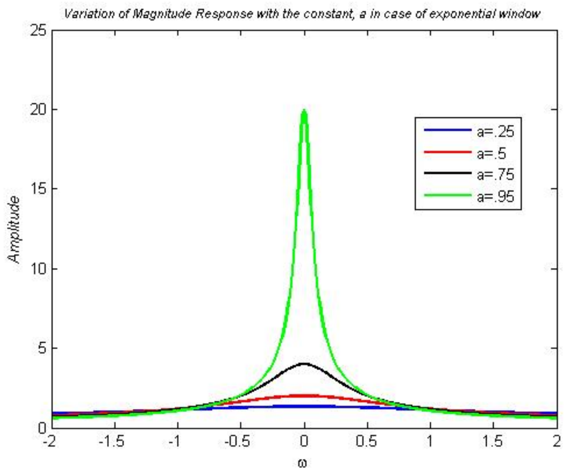

Figure 2: Magnitude Response of an Exponential Window

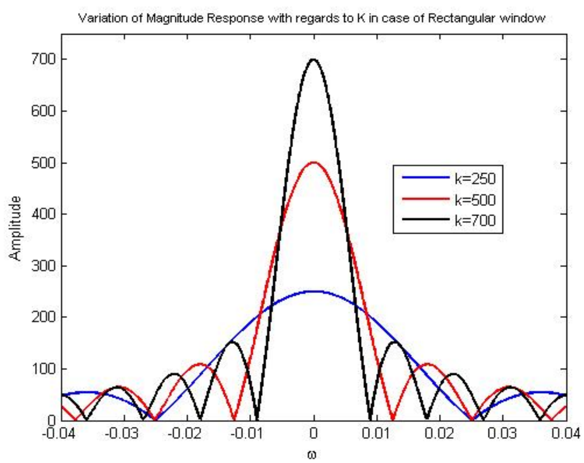

Figure 3: Magnitude Response of a Reactangular Window

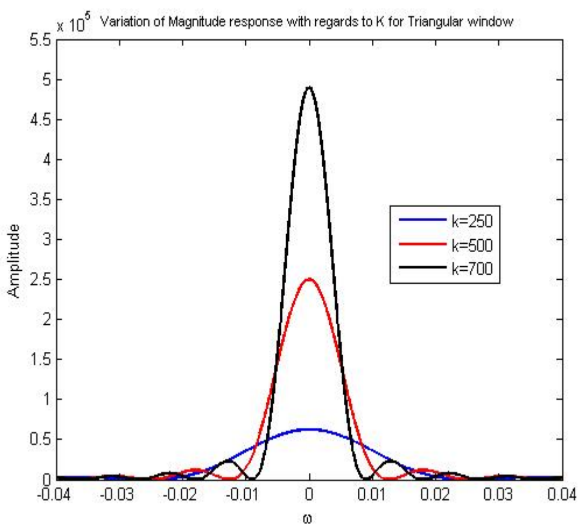

Figure 4: Magnitude response of a Triangular Window

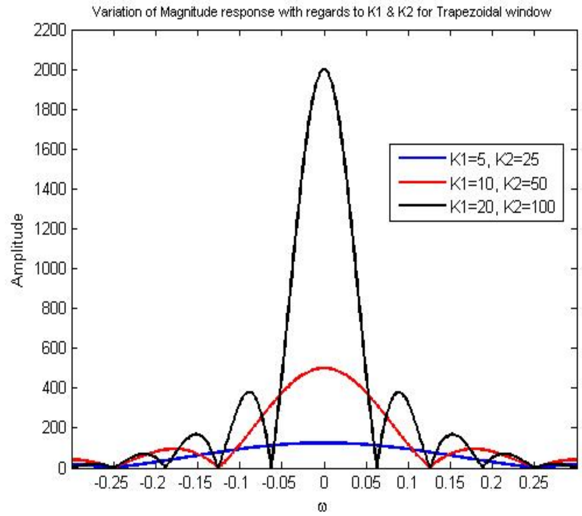

Figure 5: Magnitude Response of a Trapezoidal Window

6 COMPARISON OF FILTERING OPERATION WITH THE EXISTING ONES USING THE PROPOSED ALGORITHM [5]

It was shown in [5, section 6] that the ***worst*** case time complexity of the proposed algorithm for performing the low pass filtering of the input auto and cross correlation sequences under the assumption $L>>K$ is $\boldsymbol{O}(L)$. Likewise, we have also explained in detail in section 4 that worst case time complexity of the proposed algorithm [5] for carrying out high pass, band pass and band stop filtering operations of the input auto and cross correlation sequences through above four types of windows under the same assumption is also $\boldsymbol{O}(L)$. We immediately infer from [10] that $\boldsymbol{O}(\boldsymbol{L})<\boldsymbol{O}(\boldsymbol{L}\log_2\boldsymbol{L}) < \boldsymbol{O}(\boldsymbol{LK})$, where 2^{nd} and 3^{rd} terms denote the worst case time complexities taken by the existing algorithms for performing filtering operations in frequency and time domains. We thus conclude from the above discussion that our proposed algorithm [5] does indeed perform the lowpass (***LPF***), highpass (***HPF***), bandpass (***BPF***) and bandstop (***BSF***) filtering operations of the input auto and cross correlation sequences through different ***FIR LSI*** windows much better than the existing algorithms in both time and frequency domains in terms of the ***worst*** case time complexity. The worst case time complexity for performing all four types of filtering operations grows linearly with the length of the input auto correlation sequence, *i.e.*, $\boldsymbol{O}(\boldsymbol{L})$.

7 CONCULUSIONS

In this paper, we discussed four types of filtering operations on the input auto and cross correlation sequences through various types of ***FIR*** windows using a very ***simple*** and ***efficient*** technique described in [5]. We described lowpass, highpass, bandpass and bandstop filtering operations on the input auto and cross correlation sequences using the proposed algorithm [5]. It was shown that all types of filtering operations can be performed in a much easier and faster manner through four types of windows discussed in the paper in time domain. It was also shown that the worst case time complexity of various filtering operations grows linearly with the size of the input auto correlation sequence and it outperforms all the existing algorithms in both time and frequency domains.

ACKNOWLEDGMENT

We are extremely grateful to the Department of Electrical Engineering of COMSATS Institute of Information Technology (CIIT) for carrying out this work. Moreover, we are also thankful to the anonymous reviewers for their valuable suggestions which really helped us in improving the quality of the paper.

REFERENCES

1. Wang Zongbo, Z., Meiguo, Gao, Xiongjun, Fu and Changyong, Jiang, Design of Real-time Convolution Processor and its Application in Radar Echo Signal Simulator. *International Conference on Computer and Information Technology*, pp. 163-166, 2008.
2. Bhattacharya, B. and Patel, Gopal, A fast implementation of Correlation of long data sequences for Coherent Receivers. *IEEE Transactions on Circuits and Systems II: Analog and Digital Processing,* vol. **49**, pp. 430-433, June, 2002.
3. Jung Kap Kuk, Z. and Ik Cho, Nam, Block Convolution with Arbitrary Delays using Fast Fourier Transform. *Proceedings of 2005 International Symposium on Intelligent Signal Processing and Communication Systems,* pp. 814-816, 2005.
4. Suresh, K. and Sreeniva, T. V., Block Convolution using Discrete Trigonometric Transforms and Discrete Fourier Transform. *IEEE Signal Processing Letters, vol.* **15**, pp. 430-433, 2008.
5. Ejaz A. Ansari, Computing Cross and Auto Correlation Sequences of Output WSS Real Random Process in Linear Time, Engineering and Applied Sciences, *Science International Lahore Journal,, volume* **26**, *issue* 1, pp. 205-209, March 2014.
6. A. V. Oppenheim, R. W. Schafer and J. R. Buck, *Discrete-time Signal Processing, Pearson edition, 2nd edition, 1994.*
7. J. G. Proakis and M. Dimitris, Digital Signal Processing, *(Principles, Algorithms and Applications), Pearson son edition, 4th edition,* 2011.
8. *S. K. Mitra, Digital Signal Processing, Tata McGraw-Hill, 3rd edition, 2006.*
9. A. Papoulis and S. U. Pillai, Prob*ability, Random Variables and Stochastic Processes. 4th Edition, Tata McGraw Hill, India, 2002.*
10. T. H. Cormen, C. E. Leiserson and R. L. Rives, *Introduction to Algorithms. MIT Press, Cambridge, MA,* 1990.