



INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

Frequency Analysis Of Sound Signal Using Fast Fourier Transform With Matlab

¹*TarifLugun and ²*C.S.P.Lugun

¹ Birsa College khunti, ² Retired Professor Department of mathematics, Ranchi, University Ranchi.

Abstract

In this paper we describe the sound wave signal analysis using Fast Fourier Transform application in Matlab. Firstly we know some important application and definition as shown in example i.e. Fourier series, Fourier transform (FT), Discrete Fourier transform (DFT), Fast Fourier transform (FFT) and Wavelet transform (WT) in Filter bank and also how to apply filter bank in Matlab, i.e. Normalized Filter, High-pass Filter, Low-pass Filter and also the Band-pass Filter and Band-stop Filter. That application is very useful in the study of frequency response in a Filter bank and Fast Fourier transform in the signal analysis. It describe the mathematical foundations of digital signal processing, sound, synthesis algorithms, multi-time frequency domain analysis associated with musical sound etc. Our aim is to consider two sound wave signals to analyze the frequency response of sound wave signal or in other words this paper is to compare and analyze two sound wave signals associated with Fast Fourier transform numerically and graphically.

Key word

Fourier Transform, Fourier series, DFT, IDFT, FFT, Matlab, Convolution, sound wave signal, and filter banks, frequency analysis etc.

Nomenclature

T: - period of the function signal (s).

t: - time variable

$x(\omega)$: - time dependent function.

$x(\omega), y(\omega), z(\omega)$: - frequency dependent function.

W: - variable

$a_k : b_k$: - co-efficient of the transform

n: - number of samples

Symbols

ω : – Angular frequency ($rad.s^{-1}$).

K: - co-efficient of the frequency.

1. Introduction

The introduction of Fast Fourier transform to the world helped in making the Fourier analysis as the most important tool for digital signal processing, sound analysis or communication system in the various fields of signal. In this paper we discuss about Fast Fourier Transform and the application of filter bank. In Fourier transform application one can decompose any periodic or non-periodic signal or function into the sum of goniometric function, namely sin and cosine. The decomposition of a complex function to set of simple function is the main advantage of the Fourier method [1]. The main reason for using FFT in engineering sciences is to transform sometime-domain digital signal into the frequency-domain signal. This approach is very useful for determining modal parameters of vibrating systems. If the vibrating system generates noise during its vibration, it is possible to record noise to a digital wave file and use the data for analysis. Then mathematically analysis gives clarity to the digital signal processing [2]. Our main focus to this paper is to compare two sound wave signals mathematically with Fast Fourier transform. In [3], the authors had recorded sound to a digital file and transformed to the data as the FFT using Matlab software and also used the finite element method and the digital analog system using Fourier series and shown its application in communication system. In this paper we define some basic application of Fourier transform and the methodology of Fast Fourier transform associated with Matlab and the important part is the use of digital filter bank to filter the frequency. It is important to see an actual magnitude response graph, using which we can observe detail in pass-band and stop-band in logarithmic scale (dB scale, plotting $M \log_2 N$).

2. Fourier series

Let the Fourier series expression will be, $x(t) = \frac{a_0}{2} + \sum_{k=1}^{\infty} (a_k \cos nT + b_k \sin nT)$. for the interval

$-T \leq t \leq T$. Where a_0, a_k and b_k are the co-efficient of the function. This series decompose any period signal or function into the goniometric sin or cosin function [4].

This paper will describe some of the basic FFT and giving an example how sound wave recorded and filter the signal of sound wave and calculating, comparing with the other method of Fourier transform.

3. Fourier transform

Let be the Fourier transform integral will be,

$$x(\omega) = \int_{-\infty}^{\infty} x(t)e^{-i\omega t} dt \dots\dots\dots (A)$$

Where ω is called the angular frequency and $x(\omega)$ is the function of amplitude spectrum of $x(t)$, there for the equation (A) is called the Fourier transform of the function $x(\omega)$.

4. Convolution

The convolution of two signals 'x' and 'y' in C^N may be defined by ''x * y'' and defined by,

$$(x * y)_n = \sum_{k=0}^{N-1} x(k)y(n - k).$$

Note that this is circular convolution (or 'cyclic' convolution).

∴ It is very important of convolution in linear system theory.

∴ Cyclic convolution can be expressed in terms of previously defined operators as,

$$y(n) = (x * h)_n = \sum_{k=0}^{N-1} x(k)h(n - k)$$

$$= \langle x, SHIFT_n(FLIP(h)) \rangle \text{ Since } (h \rightarrow real)$$

Where $x, y \in C^N$ and $h \in R^N$.

5. Discrete Fourier transform

Let the signal is discrete and periodic, we can't continue Fourier transform. Instead we have to use the discrete Fourier transform (DFT).let us consider our signal is a_n for $n = 0, \dots, N-1$ and $a_n = a_n + jN$ for all n and j . the discrete Fourier transform of 'a', also known as the spectrum of 'a', i.e. -

$$A_k = \sum_{n=0}^{N-1} e^{-i\frac{2\pi}{N}Kn} a_n$$

This equation is also be written as,

$$A_k = \sum_{n=0}^{N-1} W_N^{Kn} a_n \dots\dots\dots (B)$$

Where, $W_N = e^{-i\frac{2\pi}{N}}$ and W_N for $K=0, \dots, N-1$, that is called Nth root of unity.

Again the A_k is the DFT of the sequence a_n , there for each is a complex numbers. Then the sequence a_n is the inverse discrete Fourier transform (IDFT) of the sequence A_k . there for the IDFT is,

$$a_n = \frac{1}{N} \sum_{k=0}^{N-1} W_N^{-kn} A_k \dots\dots\dots(C)$$

6. Discrete signals

The signal field and information processing and transmission, digital technology is used in various applications, there for in binary signal represent only two state contain very little information compared to an analog signals, if a quantity to be represent digitally requires a wider range of value it must be described by several bites show in figure (1),[5][6][7].

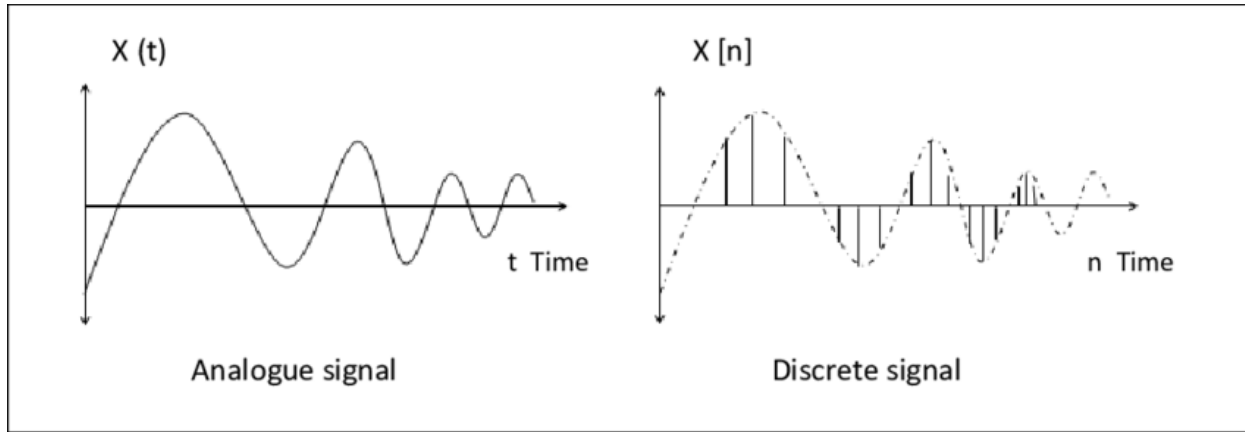


Figure (1). Analogue signal wave and discrete signal wave

7. The Fast Fourier transforms (FFT)

Let the Fast Fourier transform (FFT) is a fast algorithm for computing the discrete Fourier transform (DFT) which is calculating time from $O(N^2)$ to $O(N \log_2 N)$ where N is the number of samples of the discrete signals [8].

i.e. $N > N^2 > N \log_2 N$.

It means that enormous time saving in comparison with DFT. There for the discretization of the time signal needed for DFT is shown in figure (2).

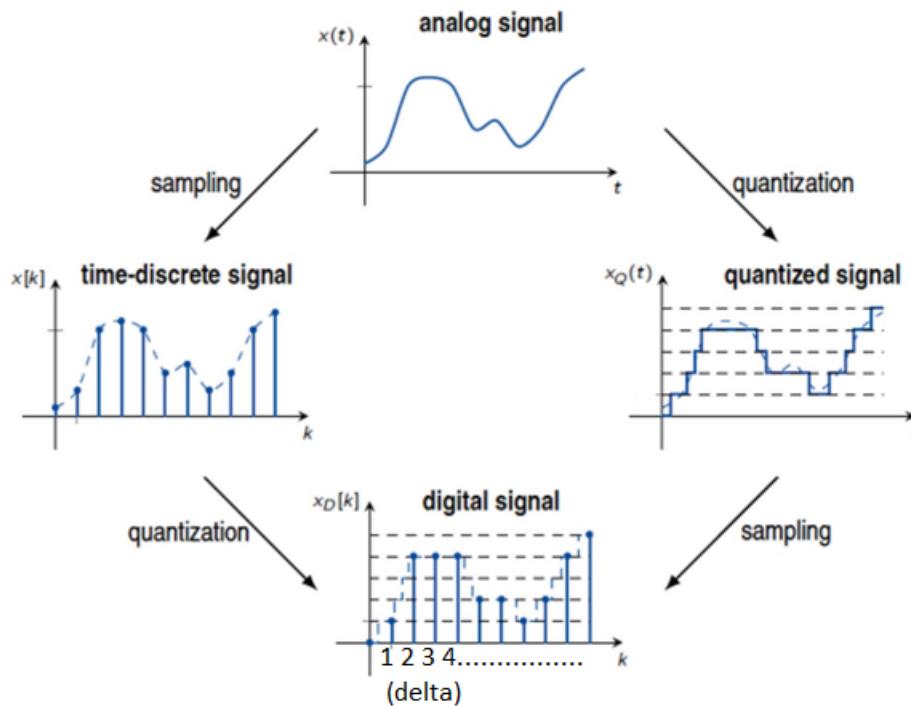
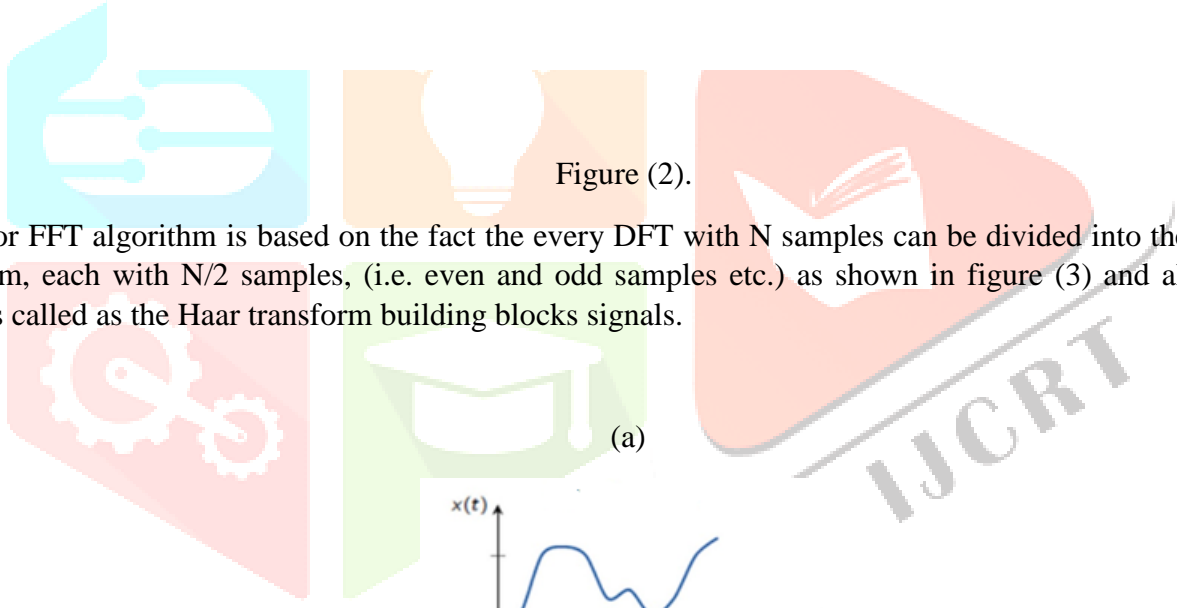
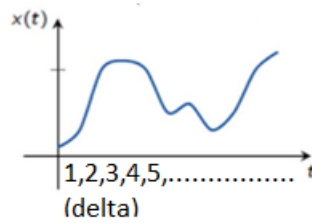


Figure (2).

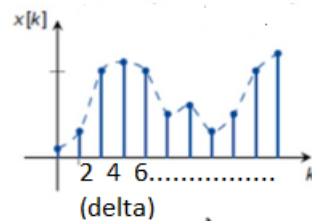
There for FFT algorithm is based on the fact the every DFT with N samples can be divided into the two Fourier transform, each with N/2 samples, (i.e. even and odd samples etc.) as shown in figure (3) and also quantized signal is called as the Haar transform building blocks signals.



(a)



(b)



(c)

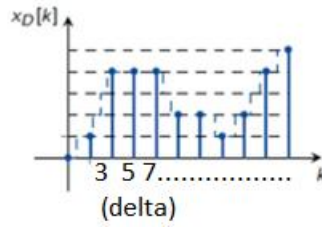


Figure (3)–(a).Normal analogue signal at number 1 t.(b).Divided of even delta sample of the signal at 2, 4, 6.....k. (c).Divided of odd delta sample of the signal at 3,5,7 k.

Now let us consider the Fourier transform is now two new Fourier transform,

$$X_k = \sum_{r=0}^{N-1} \chi_r e^{-i \frac{2\pi kr}{N}}$$

$$= \sum_{r=0}^{\frac{N}{2}-1} \chi_{2r} e^{i \frac{2\pi k(2r)}{N}} + \sum_{r=0}^{\frac{N}{2}-1} \chi_{(2r+1)} e^{i \frac{2\pi k(2r+1)}{N}} \dots\dots\dots (D)$$

$$= \sum_{r=0}^{\frac{N}{2}-1} \chi_{2r} e^{i \frac{2\pi kr}{N/2}} + e^{i \frac{2\pi k}{N}} \sum_{r=0}^{\frac{N}{2}-1} \chi_{(2r+1)} e^{i \frac{2\pi kr}{N/2}}$$

$$Y_k = \sum_{r=0}^{\frac{N}{2}-1} \chi_{2r} e^{i \frac{2\pi kr}{N/2}} \dots\dots\dots (E)$$

Where ‘r’ is the sampled number.

Therefor equation (D) can be defined in real variable.

$$Z_k = \sum_{r=0}^{\frac{N}{2}-1} \chi_{(2r+1)} e^{i \frac{2\pi kr}{N/2}} \dots\dots\dots (F)$$

And there for the complex variable,

$$i.e. \omega = e^{-i \frac{2\pi}{N}} \dots\dots\dots (G)$$

∴ ⇒(D), (E), (F) and (G) we have;

$$X_k = Y_k + W^k * Z_k \dots\dots\dots (H)$$

∴ ⇒Equation (I) is included in most of the digital signal processing [9][10][11][12][13].

8. Determining Sound Frequency in Matlab with Filter Bank

(a) Let us consider two audio files, 1st one is koyal bird sound, and the file length will be 0.75 and dawn -lording bit rate is 48 – kbps and the size of the file is 44.2 – kb. Now coding the program in matlab with respect of Digital filter bank. Firstly we see that the Normalized filter, High-pass, Low-pass, Band-pass and Band-stop. We have taking even value i.e. $n = 10$ and $f = 0.8$ for the Normalized filter, High-pass, Low –pass and taking value $f_2=0.3$ and $f_1=0.8$ coding for band-pass and equal value again taking for Band–stop f_2 and f_1 . And also complete the coding of Digital filter bank program in Matlab. Then we see the following graph with frequency response, [14][15][16].The Matlab code is provided in appendix (8a).

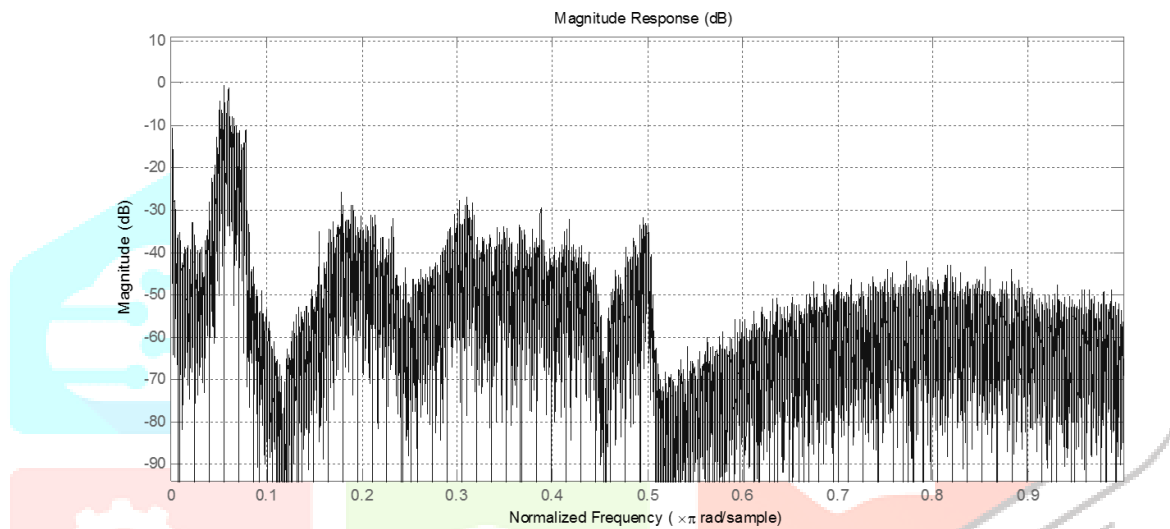


Figure (a). Normalized filter frequency

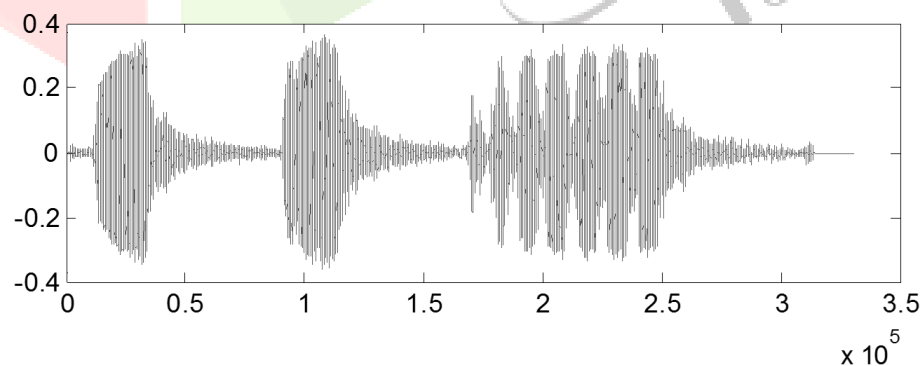
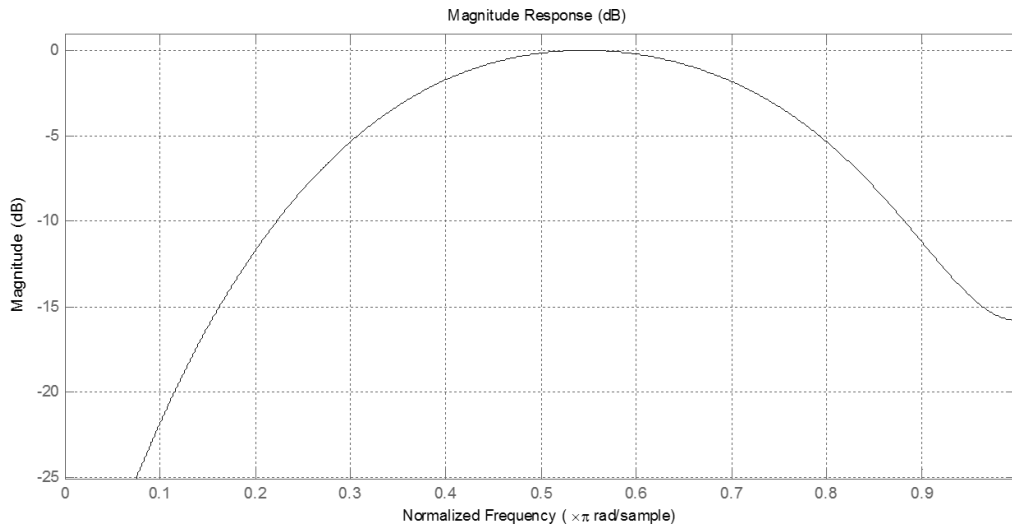


Figure (b). High-pass and low-pass filter



Figure(c).Normalized frequency at Band-pass

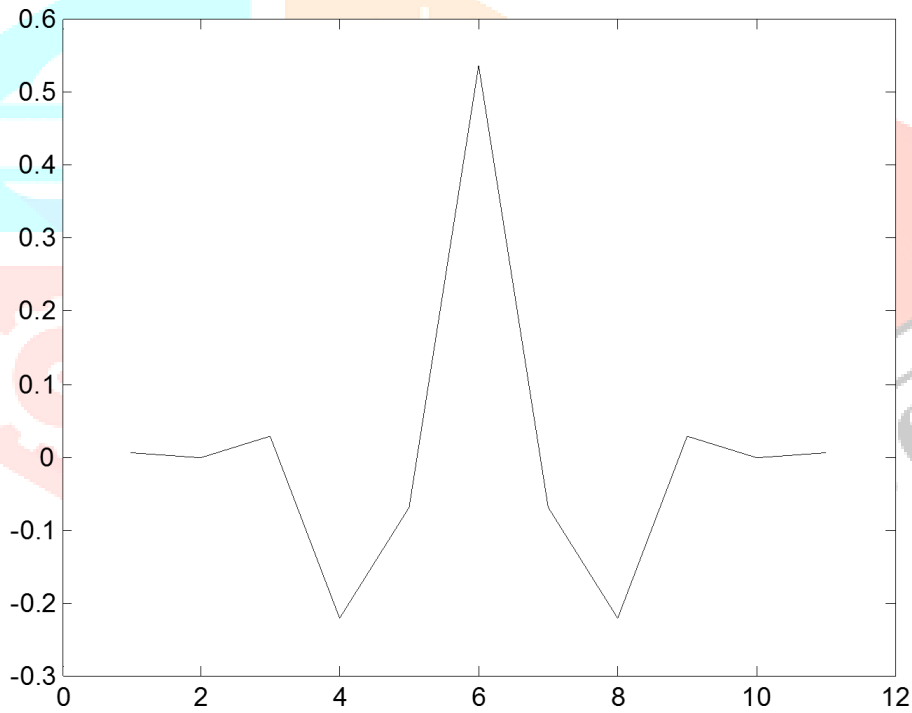


Figure (d). Band-pass filter

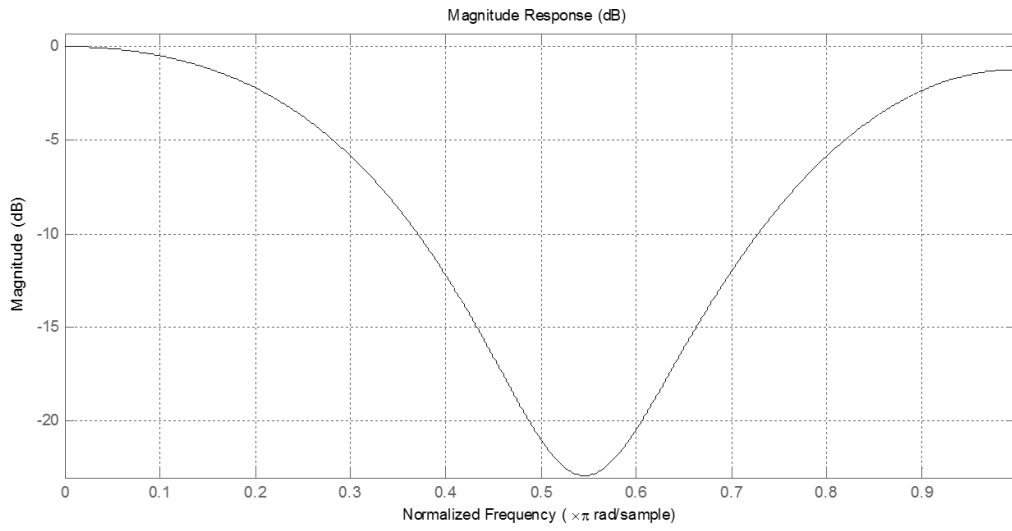


Figure (e).Normalized frequency at Band-stop

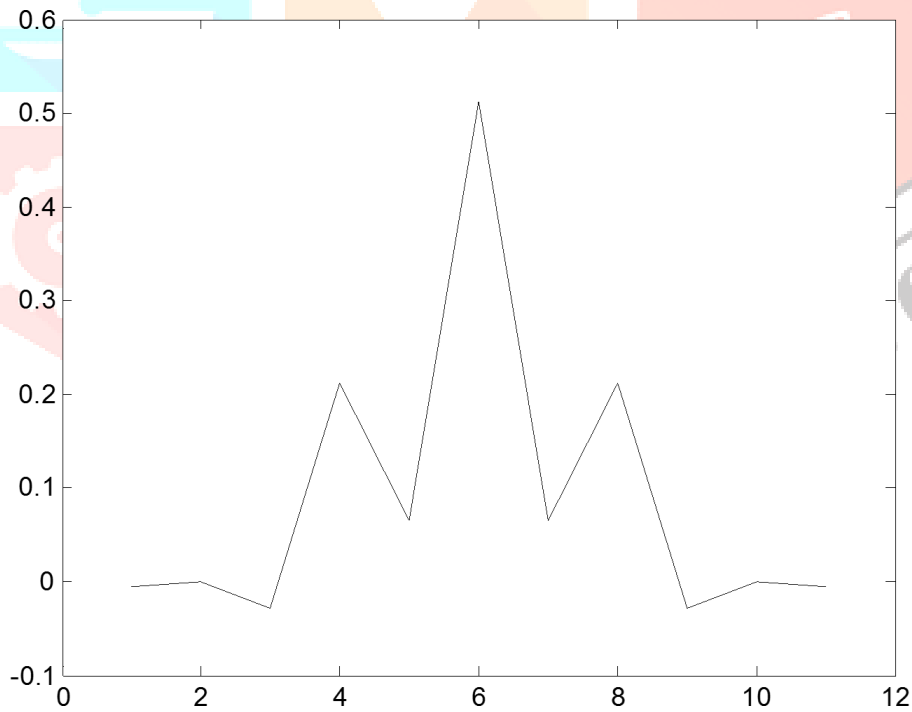


Figure (f).Band-stop filter

(b) On other hand the second sound will be a piano base key sound and the file size will be 862 kb. Now we code the program in Matlab with respect of Digital filter bank. Similarly equal value taken according as 8(a). Firstly we see that the Normalized filter, High-pass, Low –pass, Band-pass and Band-stop. We have taking even value i.e. $n = 10$ and $f = 0.8$ for the Normalized filter, High-pass, Low –pass and taking value $f_2=0.3$ and $f_1=0.8$ coding for band-pass and equal value again taking for Band–stop f_2 and f_1 . And also complete the coding of Digital filter bank program in Matlab. Then we see the following graph with frequency response, [17] [18] [19]. The Matlab code is provided in appendix (8b).

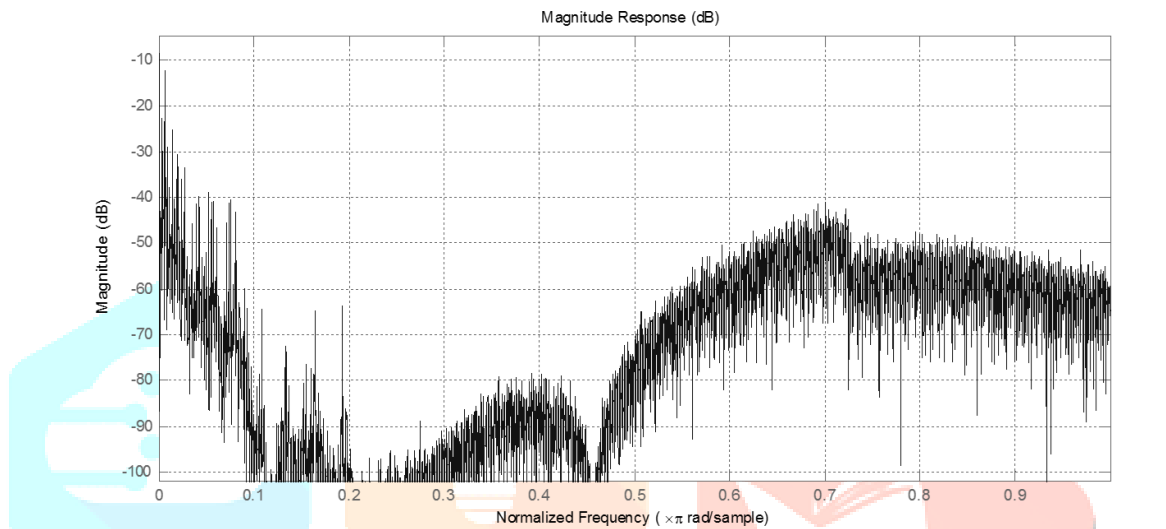


Figure (a). Normalized filter frequency

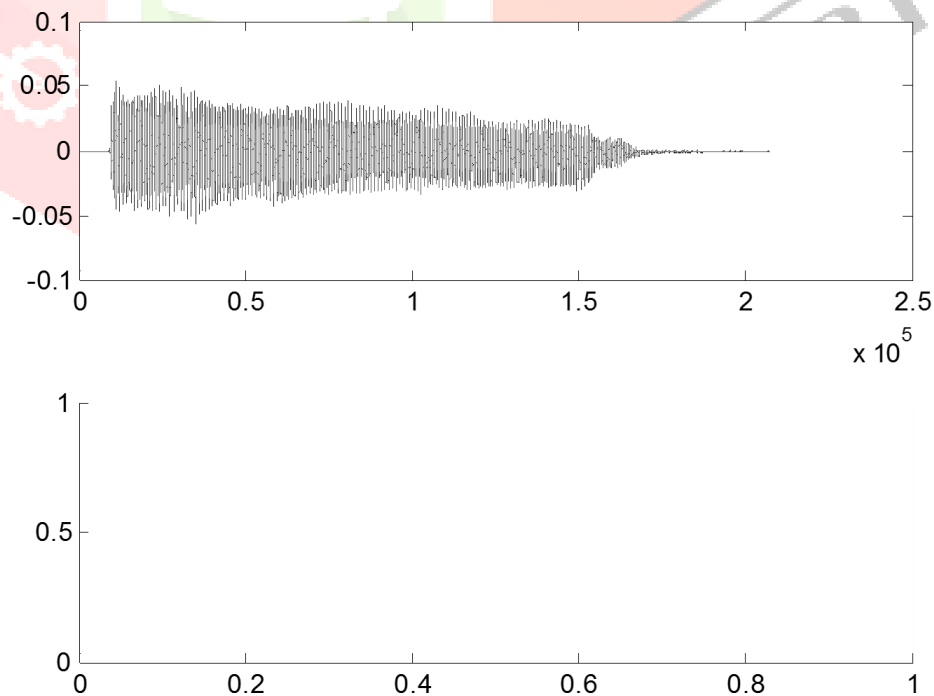
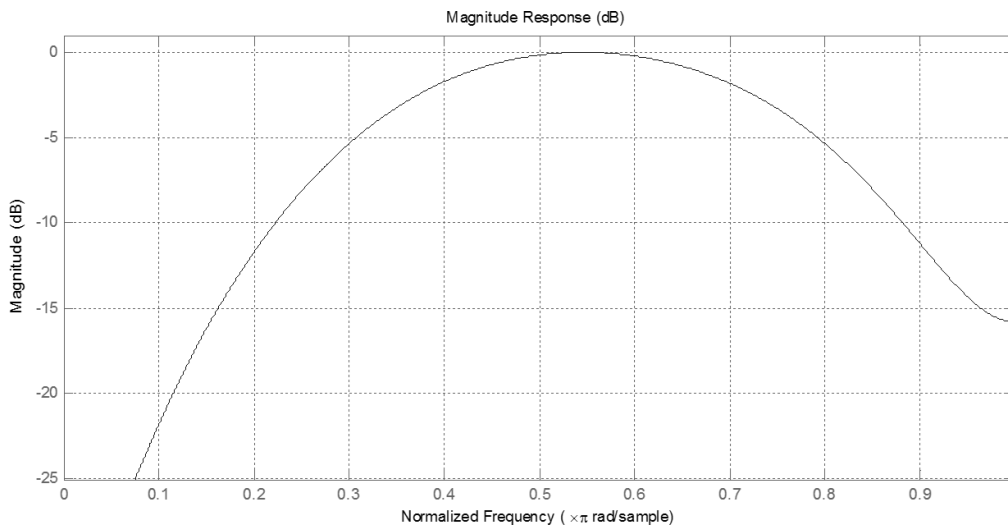


Figure (b). High-pass and low-pass filter frequency



Figure(c). Normalized Band-pass frequency

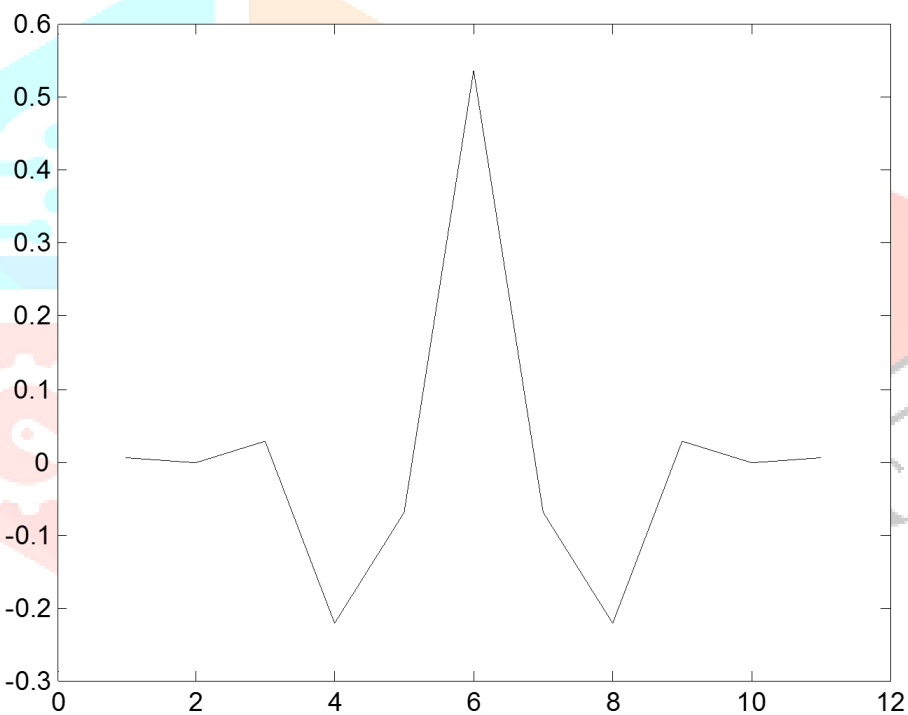


Figure (d).Band-pass filtered frequency

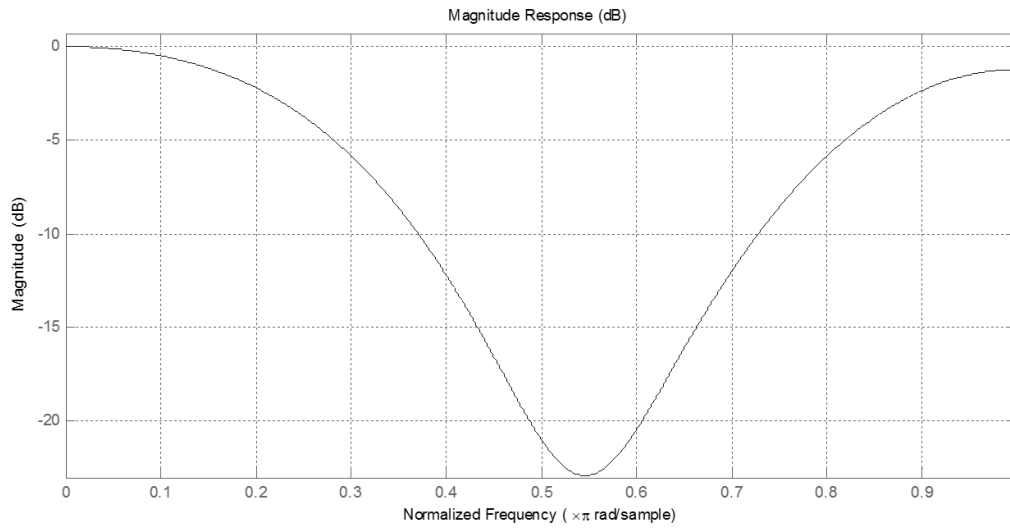


Figure (e).Normalized band-stop frequency

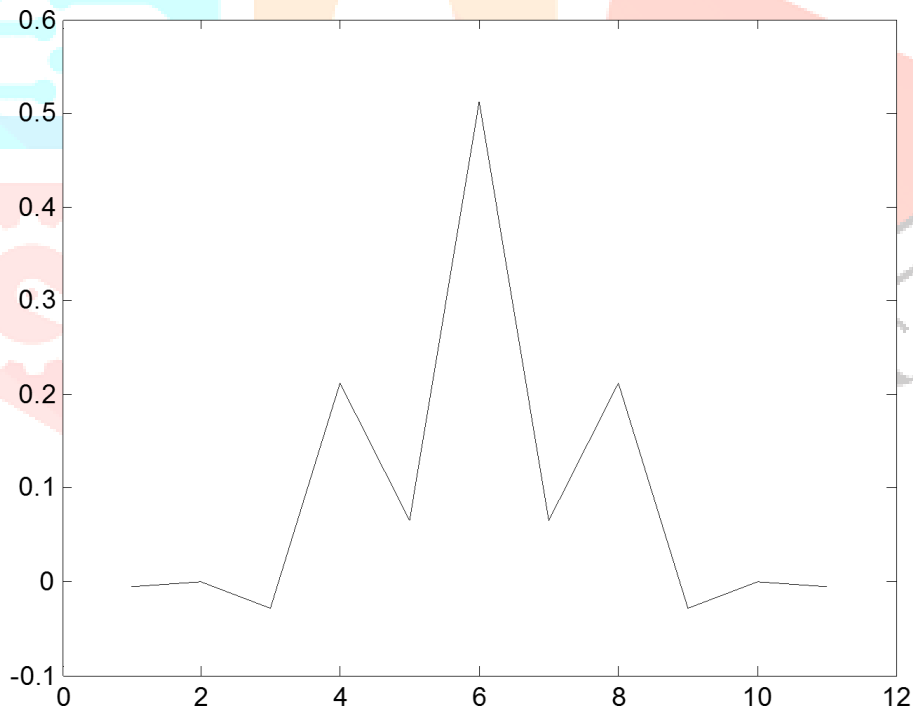


Figure (f).Band-stop filtered frequency

2. Comparison of Filter Bank with Fast Fourier Transforms

Now we have to creating table;

	Samp led value of filter- bank(n)	Func tion vari able of filter- bank(f 1)	Func tion vari able of filter- bank(f 2)	FFT Normal ized Filt er $\times\pi$ rad/sam ple	High fre quency $\times 10^5$	Low fre quency $\times 10^n$	Normali zed Band- pass fre quency	Band- pass fre quency	Normali zed Band- stop fre quency	Band- stop fre quency
1	10	0.8	0.3	0-0.9	0-3	0	0.1-0.9	0-0.5	0.5-0.6	0-0.5
2	10	0.8	0.3	-10 – 0.9	0-2	0	0.1-0.9	0 – 0.5	0.5-0.6	0 – 0.5



3. Discussion on Results for Sound Wave Magnitude Response (dB)

Now the table (1), it is shown 8.(a) in the figure (a) i.e. Normalized frequency must be lie on the interval 0dB to 0.9dB. Then again the figure(b), high pass filter length tends to 0 to 3.5 and the width frequency must be tends to nearly equal to -0.4 to 0.4 and the difference of wavelength is approximately 0.5 at the point 1.5 then the difference was decreasing as the signal goes and the pre-ring formed nearly to 0 and post-ring was formed in nearly to 2.5. As figure (b) we see the low pass frequency approximation it will be zero because there is no graphical structure shown in the figure (b) low-pass filter and there is no pre-ring or post-ring be found. In figure (c) normalized band pass frequency, initial rising point of the graph between 0 to 0.1 and its graphical shape will be a parabolic and it is lie down to middle of the point -20 to -15. There for again in figure (d) the filter frequency of band pass tends to higher at 0.5 to 0.6 and downwards at 4 to 8 and there is 9 breaking point, and the 1st coordinate point is nearly to (2,0), 2nd is (3,0), 3rd is (4,-0.2), 4th is (5,-0.5), 5th is (6,0.5), 6th is (7,-0.5), 7th is (8,-0.2), 8th is (9,0) and 9th is (10,0). Again figure (e) A normalized band-stop frequency. A curve shape like English alphabet 'v' and the top of the crust lie downstream between 0.5 to 0.6 and it will start at 0 and end at 0 to -5 magnitude response. Again in figure (f) the filtered band-stop frequency, the higher response between 0.5 to 0.6 and lie downstream at between 2 to 4 and 8 to 10, it means that band stop graph has 9 beaks point, i.e. 1st coordinate of breaking point is (2,0), 2nd is (3,-0.5), 3rd is (4,0.2), 4th is (5,0.5), 5th is (6,0.5), 6th is (7,0.5), 7th is (8,0.2), 8th is (9,-0.5), and 9th is (10,0).

Again the table (2), we see that 8.(b) in figure (a) the normalized frequency in piano base key was higher magnitude (dB) between 0 to -10 and the graph will continued between 0 to 0.9. And also in figure (b) the high pass frequency length will be 0 to 2 and the width of the frequency between 0.05 to -0.05 and the difference of the wavelength of the frequency is equal to 0.01 and the graph continued between 0 to 2 the wavelength will continued decreases and it is less than 0.01. There is also pre-ring formed in nearly to 0 point and post-ring was formed in after point 1.5. And also the figure(b) shows the low pass frequency approximation will be zero and there is no graphically representation. And there for the figure (c), (d), (e), and (f) in table (2) the graph representing or remain approximately nearly equal to as describe in above, because we taking the equal value in FFT Filter bank matlab programming and he all graphically representation will be one dimensional plots in matlab.

4. Conclusion

Estimating the frequency response (dB) for a physical system commonly involves executing the system with an input sound wave frequency signal and also measuring both input and output time history and comparing the two through a process such as the Fast Fourier transform (FFT). We have two sound frequency one is of bird and other is of piano base key. Bird have a sound frequency sample value of 10, and as per observation the frequency deviation are very high, ranging from 0 to 3 db and the other ,piano base key has also a sound frequency value of 10, and as per observation the frequency deviations is from 0 to 2. So we

conclude that both cases have a common thing in nature and that is low frequency value which is zero. So, we know from the above observation that FFT is the best method to analyze bird type sound frequency. It is the best for transforming some digital signal from time domain to frequency domain and also we can transform different data from sound signal obtained acceleration, displacement, velocity, normalized filter, high pass filter, band pass filter, translation, band stop filter, breaking point and many others.



Appendix 8. (a)

```
1. clc;
2. closeall;
3. clearall;4. f=0.8;
4. n=10;
```

APPENDIXMATLABCODE

```
6. a=fir1(n,f,'high');           %fir high pass filter.
7. b=fir1(n,f,'low');           %fir low pass filter.
8. [y,fs]=audioread('audiofile.wav');           %load audio file.
9. o=filter(a,1,y);           %pas ing audio to deigned high pass filter first.
10. p=filter(b,1,o);           %passing to low pass filter.
11. fvtool(p,1);           %use to display deigned fir filter.
12.subplot(2,1,1);
13. plot(y);           %original signal.
14.subplot(2,1,2);
  %BANDPASS FILTER15.
f2=0.3;
16. f1=0.8;
  %filter3=fir1(n,(f2,f));
17. filter3=fir1(n,[f2,f]);
18. fvtool(filter3,1);
19. figure;
20. plot(filter3);
  %BANDSTOP FILTER21.
f2=0.3;
22. f1=0.8;
23. filter4=fir1(n,[f2,f1],'stop');
24. fvtool(filter4,1);
25. figure;
26. plot(filter4);
27. output
```

Appendix8.(b)

```
1. clc;
2. closeall;
3. clearall;
4.f=0.8;
5. n=10;
6. a=fir1(n,f,'high');           %fir high passfilter.
7. b=fir1(n,f,'low');           %fir low pass filter.
8. [y,fs]=audioread('piano.wav');           %load audio file.
9. o=filter(a,1,y);           %passing audio to deigned high pass filter first.
10. p=filter(b,1,o);           %passing to low pass filter.
11. fvtool(p,1);           %use o display deigned fir filter.
12.subplot(2,1,1);
13. plot(y);           %original signal.
14.subplot(2,1,2);
  %BANDPASS FILTER15.
f2=0.3;
16. f1=0.8;
  %filter3=fir1(n,(f2,f));
17. filter3=fir1(n,[f2,f]);
18. fvtool(filter3,1);
```



```

19. figure;
20. plot(filter3);
% BANDSTOP FILTER 21.
f2=0.3;
22. f1=0.8;
23. filter4=fir1(n,[f2,f1],'stop');
24. fvtool(filter4,1);
25. figure;
26. plot(filter4);
27. output

```

References

- [1] Bracewell.R.N.1991.theFouriertransformanditsapplications,McGraw-Hill.
- [2] Gilbert Strang /Truong Nguyen, Wavelets and Filter Banks,Wellesley-CambridgePress,ISBN0-9614088-7-1.
- [3]FrequencyanalysisofacousticsignalFastFouriertransformonmatlab,byTomas Harcarik, Jozef Bocko, Kristina Masloková(2012)".
- [4] Hanna.J. Ray, Rowland, John,H, 1990. Fourier series, transform and boundary value problems, Braun-Brumfield.U.S.A,ISBN 0-471-61983-3.
- [5]Introduction to sound processing, By Davide RocchessouniversitadiVerona,dipartimento diInformaticaISBN 88-901126-1-1.
- [6] Modern Digital and Analog communication system by B. P Lathiand Zhi Ding.
- [7] Howtouse the Fast Fourier Transform in large finite fields ,Petur BirgirPetersenRungstedKyst–paper[2009],press-arXiv:1101.3001v1[math.NT], pp.1-27.
- [8] AfirstcourseinwaveletswithFourieranalysis2ndEditionISBN978-0-470-43117-7.
- [9] Uses of Integral Transform, by Ian .N. Sneddon, McGraw-Hillbook Company
- [10] An introduction to Wavelet Analysis by David.F.Walnut,ISBN978-81-8489-020-4.
- [11] Shakarchi Ramiand Stein Elias M, Fourier analysis an introduction.(2003),new jersey press Princeton Edu.
- [12] FollandG.B,1992,Fourieranalysisandapplication,A.M.A,Wadsworthhandbook/ColeAdvanced book and software pacific Grove,California.
- [13] Debnath Lokenath, wavelet transform and their application,(2002),USA, Birkhauser Boston.

- [14] Matlab an Introduction with Application by Amos Gilat, 4th Edition, ISBN 978-81-265-3720-4.
- [15] Digital Signal processing using Matlab for Students and researchers. By JOHN.W.LEIS University Of Southern Queensland, ISBN978-0470-88091-3.
- [16] IEEE.SpecialissueonfastFouriertransforms.IEEETran.OnAudioandelectroAcoustics,AU-17:65-186, 1969.
- [17] Communication system by Simon Havkins.
- [18] Communication system (analog todigital)bySanjaySharma.
- [19] Ivanco, V.,kubin K., Kostolny, K., 1994 Metoda Konecnychprvkov I, Kosic: Elfa. ISBN 80-96731-4-0.
- [20] Richard E. Blahut, Algebraic codes for DataTransmission,pp.132-133,CambridgeUniversityPress,Cambridge, 2003.ISBN 0-521-55374-1.
- [21] Duhamel,p.,piron,B.,Etcheto,J.M.,”OncomputingtheinverseDFT,”Acoustics,SpeechAnd Signal Processing, Vol. 36,no.2, pp. 285-286,Feb-1988.
- [22] Trebuna,F.,Simcak,F.,2007.Priruckaexperimentalnejmechaniky.TypoPressKosiceISBN970-80-8073-816-7.