DIGITAL SIGNAL PROCESSING
(ECE 2006)

TITLE- EFFECT OF SIGNAL PROCESSING FILTERS ON MEASUREMENT OF MAXIMUM SOUND PRESSURE LEVELS

MEMBERS-

- MEMBER 1- DIPENDU HAZRA
- REGISTRATION NUMBER- 20BEC0592

- MEMBER 2 – LINGESWAR C
- REGISTRATION NUMBER- 20BEC0579

- MEMBER 3 – PRASANNA B
- REGISTRATION NUMBER- 20BEC0587

BASE PAPER(ELSEVIER)-

Title- Effects of signal processing on the measurement of maximum sound pressure levels

Author- M. Robinson, C.Hopkins.

Robinson, M. & Hopkins, Carl. (2014). Effects of signal processing on the measurement of maximum sound pressure levels. Applied Acoustics. 77. 11–19. 10.1016/j.apacoust.2013.09.017.

Software used-
MATLAB

ABSTRACT-

- Sound pressure, or sound pressure level, is the result of the pressure variations in the air achieved by the sound waves.
- Sound-level meter ,device used for measuring the sound pressure level by measuring the intensity of noise, music, and other sounds .
- A typical meter consists of a microphone for picking up the sound and converting it into an electrical signal, followed by electronic circuitry for operating on this signal so that the desired characteristics can be measured.
- A sound level meter has four main components. These are input filters to remove unwanted frequencies from the signal that is being measured, Constant Percentage Bandwidth (CPB) filters , an A-weighting filter, and a time-weighted level detector to convert an AC signal into a DC signal. The final stages involve statistical, time-based averaging or peak detection components to determine the desired parameter.
- The focus of paper is to see which is better filter CPB or A or C and which must be used in sound level meter for the analysis of sound.

Literature review:

- M.RobinsonC.Hopkins in [1] describes about four different commercially-available sound level meters are used to quantify the variation in measured maximum levels using tone bursts, half-sine pulses, ramped noise and recorded transients.
- Sanja Grubesa, Mia Suhanek in [2] research about noise pollution and did a survey in various urban places and check the range of noise pollution in those areas with a sound meter of a mobile application.
- H. Lin, Q. Tang, J. Li, Z. Teng, H. Chen in [3] presents a new digital impulse-weighting(I-weighting) method for sound level meter constructed by designing a peak detector with attenuator and attending the existing time-weighting model.
- Zhong Bo1 , Xu Huan1 , Sun Qingsheng2 , He Longbiao1 , Niu Feng1 , Bai Ying1 and Yang Ping1 in [4] described the importance and problems related to automatic calibration for sound level meter and then presented a automatic calibration system based on image recognition technology.
- W. E. SCHOLES and A. C. SALVrDGE[5] describes about a simple attachment to a sound level meter that gives values of L10 and of L90 level of sound level meter within 1 dB(A) of the values obtained by the normal method measurements and helps to minimise operator bias.
- D. H. STEPHENS[6] discussed about a practical walls and floors, taken from National Building Studies Research Paper showed that there was good correlation between the sound level difference calculated as if it had been measured directly with a sound level meter.
- Y.T. Kim *, Y.B. Lee, M.J. Jho, S.J. Suh[7] describe the uncertainties in the semi-automatic calibration and that in the full automatic calibration were estimated for the recently developed SLM calibration system.

Methodology:

Any unwanted or distributed noise that effects badly that is called sound pollution. For human ears if the sound is greater than 62 dB then it is assumed as sound pollution. Sound level meter is used to measure the pressure of the noise. There are 2 types of sound level meter in market, one with A-weighting filter and one with C-weighting filter. CPB (Constant Percentage Bandwidth) filter gives the sampled output of the filters. The main target of this paper is to check the analog sampled output of both sound level meter with the help of CPB filter and which sound level meter works better for the measurement of noise lying in the human hearing range frequency, i.e. (20HZ to 20000HZ).

1) First we will be analysing A weighted and C weighted filter, in a frequency analyser to monitor there magnitude response.
2) Then we will take a piece of sound and for each filter A weighted and C weighted we will see how the filtered output behaves with respect to original sound signal.
3) Third we will plot average power vs frequency diagram for octave and 1/3 octave filter.
4) At last ,A weighted and C weighted along with 1/3 octave filter diagram is plotted.
5) Finally according to the diagram we conclude which filter is best for the measurement of sound lying in frequency range of human ears.

A and C weighted filter-

MATLAB CODE-

```
%for A wiegthed filter , graph plotted dB vs frequency
aWeight = weightingFilter('A-weighting','SampleRate',44100);
complianceStatus = isStandardCompliant(aWeight,'class 1')
visualize(aWeight,'class 1’)
%for C wiegthed filter , graph plotted dB vs frequency
cWeight = weightingFilter('C-weighting','SampleRate',44100);
complianceStatus = isStandardCompliant(cWeight,'class 1')
visualize(cWeight,'class 1')
```

OUTPUT-

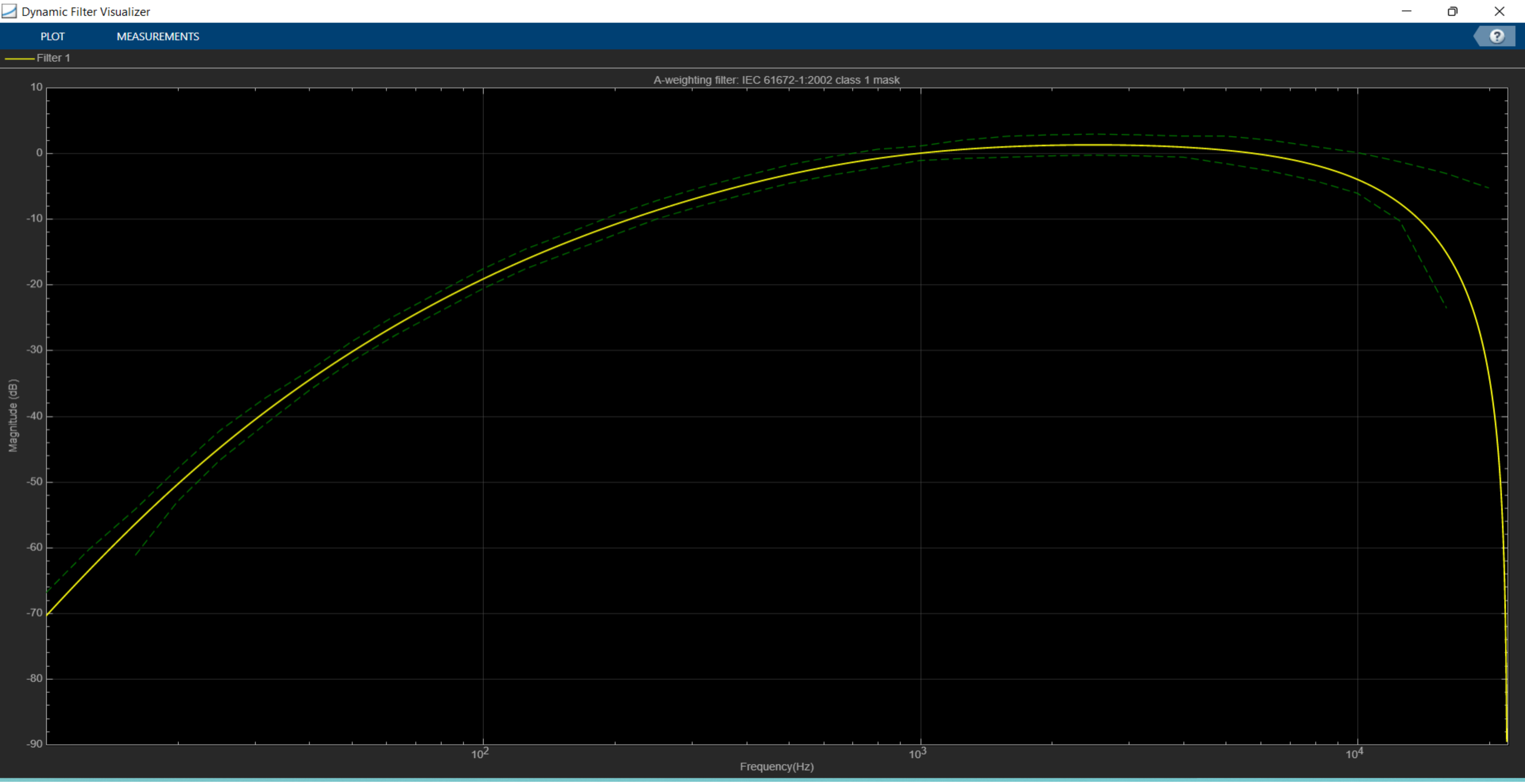

Fig. 1. *Frequency(Hz) vs Magnitude(dB) graph for A weighted filter*

OUTPUT-

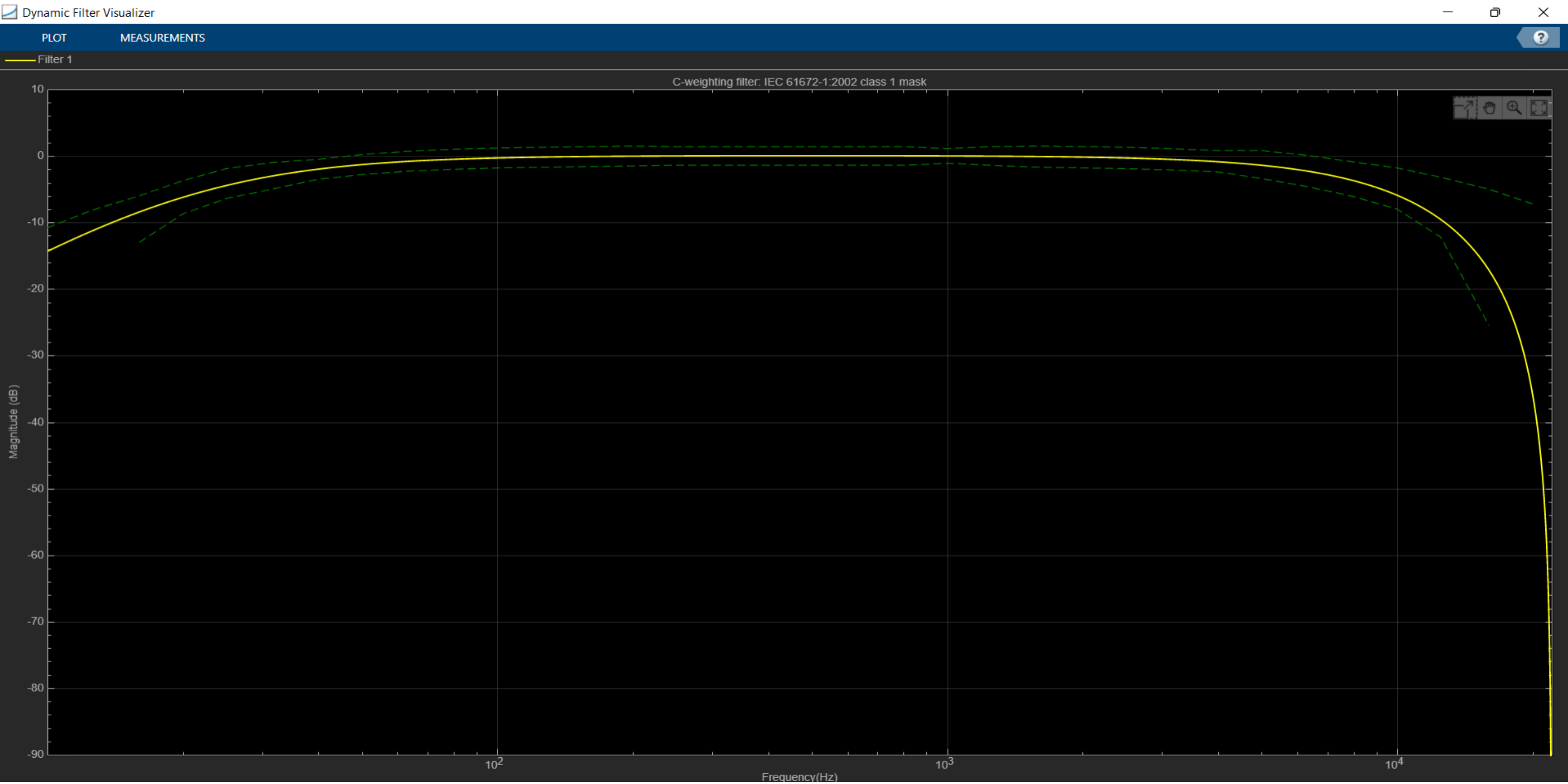

Fig. 2. *Frequency(Hz) vs Magnitude(dB) graph for C weighted filter*

Behavior of filters on a piece of sound-

- A weighted filter-

```
clc;
clear;
close all;
samplesPerFrame = 1024;
reader = dsp.AudioFileReader(Filename="C:\Users\ARPIT BHAGAT\OneDrive\Documents\MATLAB\mixkit-stadium-crowd-light-applause-362.wav", SamplesPerFrame=samplesPerFrame, PlayCount=Inf);
Fs = reader.SampleRate;  %sample íate of the íeadeí used as the sample íate of the weighting filteí.
weightFilt = weightingFilter("A-weighting",Fs);
scope = spectrumAnalyzer( SampleRate=Fs, PlotAsTwoSidedSpectrum=false, FrequencyScale="log", Title="A-Weighted Filtering", ShowLegend=true, ChannelNames=["Original signal","Filtered signal"]);
tic
while toc < 20
x = reader();
y = weightFilt(x);
scope([x(:,1),y(:,1)])
end
release(weightFilt)
release(reader)
release(scope)
```

OUTPUT-

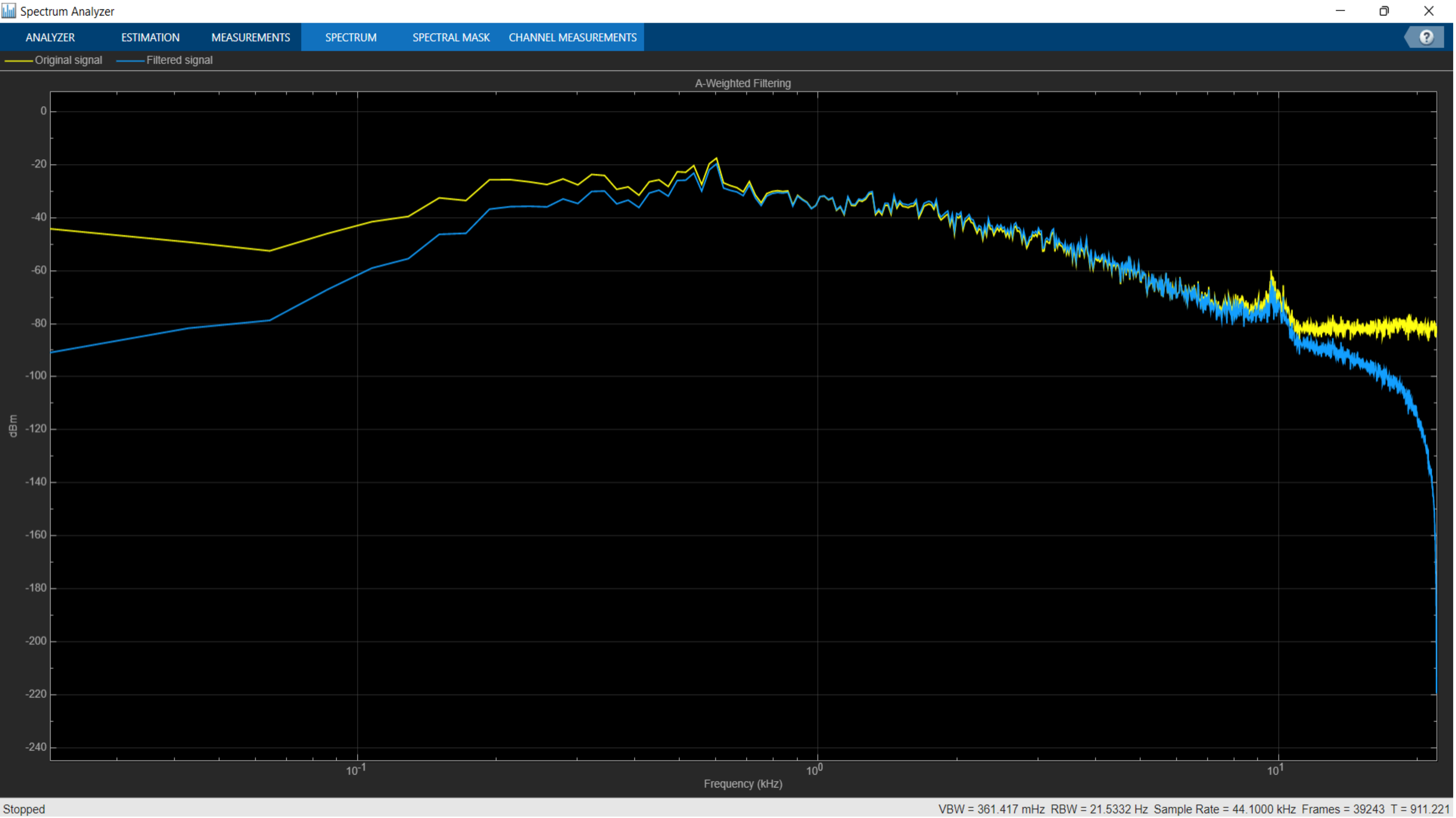

Fig. 3. *Frequency(Hz) vs Magnitude (dB) graph for A weighted filter on a piece of sound.*

C weighted filter-

MATLAB Code-

```
clc;
clear;
close all;
samplesPerFrame = 1024;
reader = dsp.AudioFileReader(Filename="C:\Users\ARPIT
BHAGAT\OneDrive\Documents\MATLAB\mixkit-stadium-crowd-light-applause-362.wav",
SamplesPerFrame=samplesPerFrame, PlayCount=Inf);
Fs = reader.SampleRate;%sample rate of the reader used as the sample rate of the weighting filter.
weightFilt = weightingFilter("C-weighting",Fs);
scope = spectrumAnalyzer( SampleRate=Fs, PlotAsTwoSidedSpectrum=false, FrequencyScale="log",
Title="C-Weighted Filtering", ShowLegend=true, ChannelNames=["Original signal","Filtered signal"]);
tic
while toc < 20
x = reader();
y = weightFilt(x);
scope([x(:,1),y(:,1)])
end
release(weightFilt)
release(reader)
release(scope)
```

OUTPUT-

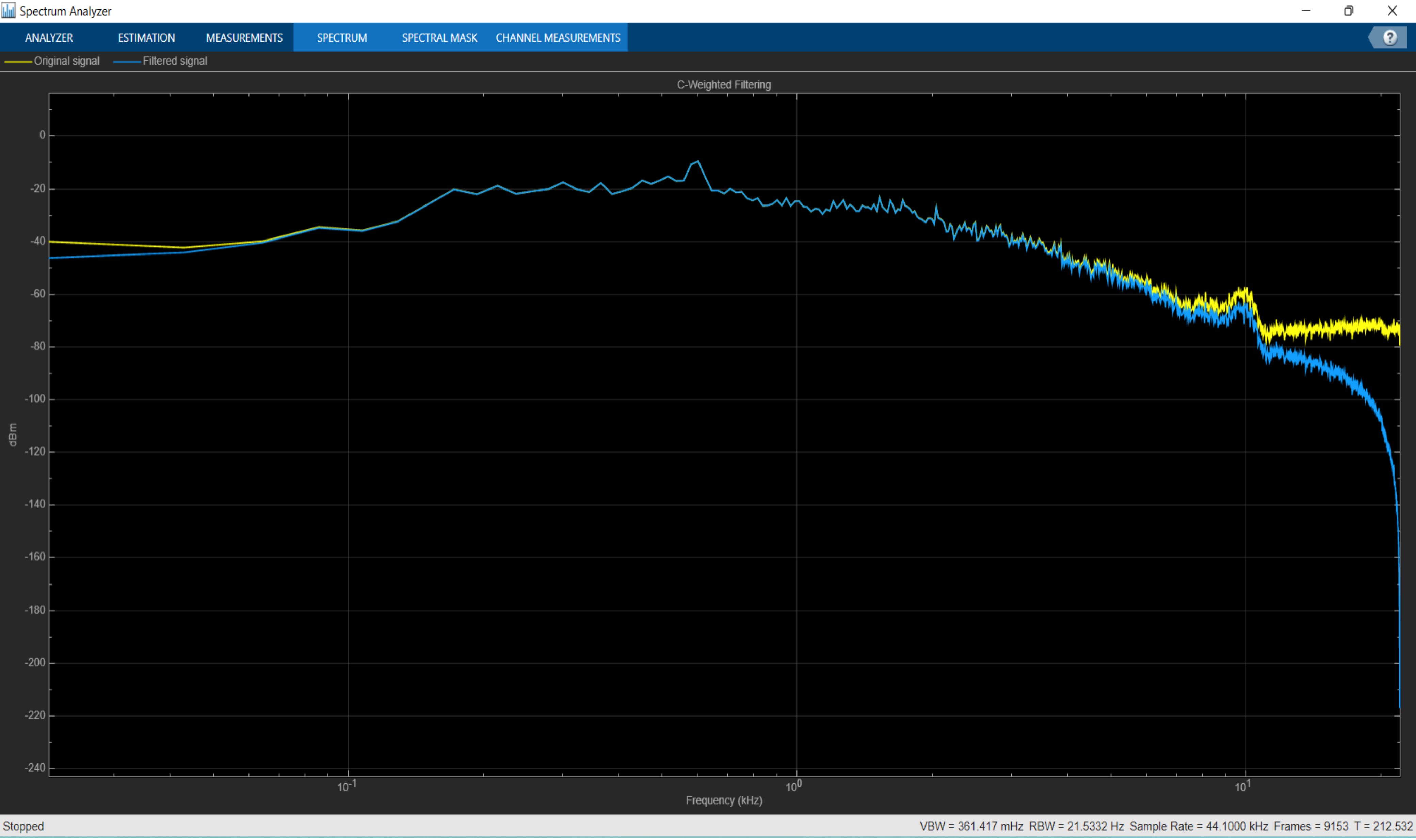

Fig. 4. *Frequency (Hz) vs Magnitude (dB) graph for C weighted filter on a piece of sound.*

Octave band filter-

- The octave spectrum is **the average power over octave ba**
- In octave band upper frequency/lower frequency=2

Uppeí fíequency=Centíal fíequency*$\sqrt{2}$Loweí fíequency=Centíal fíequency/$\sqrt{2}$

Centíal fíequency= $\sqrt{}$uppeí fíequency*loweí fíequency

- In 1/3 octave band upper frequency/lower frequency= $\sqrt[3]{2}$

Uppeí fíequency=Centíal fíequency* $\sqrt[6]{2}$

Loweí fíequency=Centíal fíequency/ $\sqrt[6]{2}$

Centíal fíequency= $\sqrt{}$uppeí fíequency*loweí fíequency

MATLAB CODE-

```
clc;
clear all;
close all;
N = 1e5;
fs = 44.1e3;
wn = randn(N,1);
z = [0.9 0.8 0.1]';
p = [0.9 0.9 0.5]';
[b,a] = zp2tf(z,p,1);% converts a factored transfer function representation
%of a system to a polynomial transfer function representation
pn = filter(b,a,wn);
%filters the input data x using a rational transfer function defined by the
% numerator and denominator coefficients b and a
flims = [200 20e3];
bpo = 1;
opts = {'FrequencyLimits',flims,'BandsPerOctave',bpo};
poctave(pn,fs,opts{:});% returns the octave spectrum of a signal x sampled at a rate fs.
```

OUTPUT-

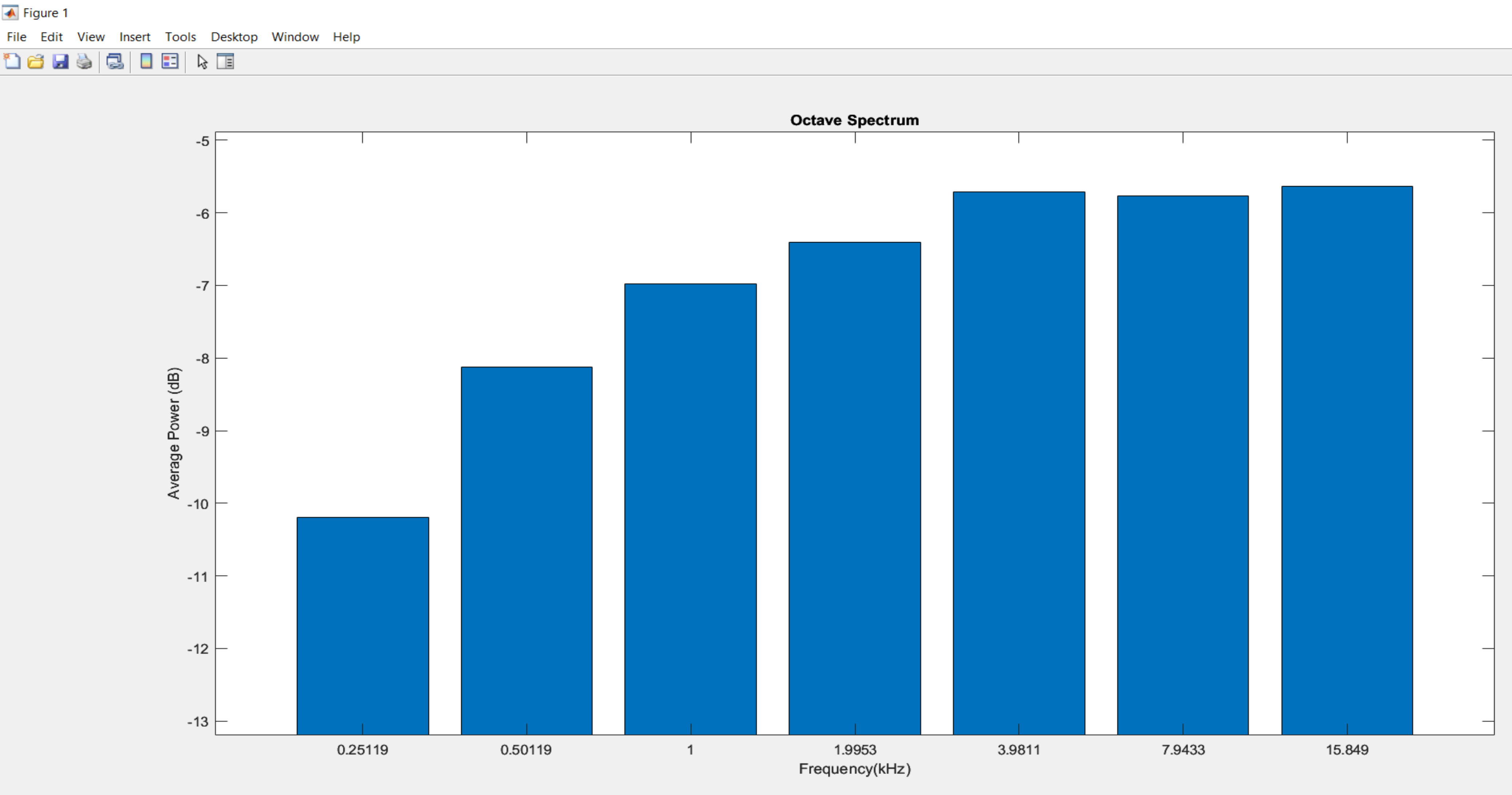

Fig.5 . *Frequency(Hz) vs Average power(dB) graph for octave filter on a piece of sound.*

1/3 octave band filter-

MATLAB CODE-

- clc;
- clear all;
- close all;
- N = 1e5;
- fs = 44.1e3;
- wn = randn(N,1);
- z = [0.9 0.8 0.1]';
- p = [0.9 0.9 0.5]';
- [b,a] = zp2tf(z,p,1);% converts a factored transfer function representation
- %of a system to a polynomial transfer function representation
- pn = filter(b,a,wn);
- %filters the input data x using a rational transfer function defined by the
- % numerator and denominator coefficients b and a
- flims = [200 20e3];
- bpo = 3;
- opts = {'FrequencyLimits',flims,'BandsPerOctave',bpo};
- poctave(pn,fs,opts{:});% returns the octave spectrum of a signal x sampled at a rate fs.

OUTPUT-

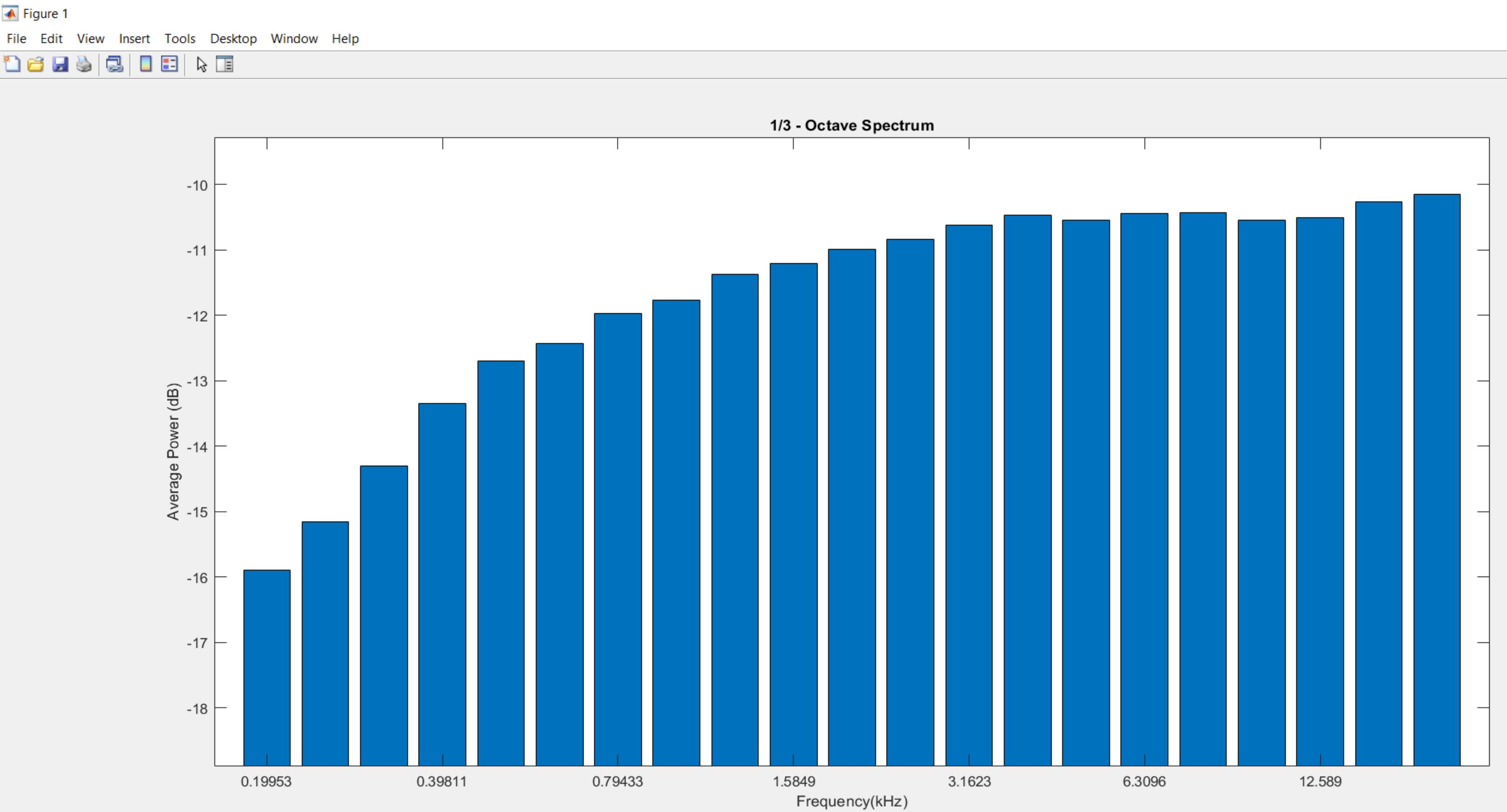

Fig.6 . *Frequency(Hz) vs Average power(dB) graph for 1/3 octave filter on a piece of sound.*

C weighted with 1/3 octave band filter-

MATLAB CODE-

```
clc;
clear all;
close all;
N = 1e5;
fs = 44.1e3;
wn = randn(N,1);
z = [0.9 0.8 0.1]';
p = [0.9 0.9 0.5]';
[b,a] = zp2tf(z,p,1);% converts a factored transfer function representation
%of a system to a polynomial transfer function representation
pn = filter(b,a,wn);
%filters the input data x using a rational transfer function defined by the
% numerator and denominator coefficients b and a
flims = [200 20e3];
bpo = 3;
opts = {'FrequencyLimits',flims,'BandsPerOctave',bpo};
poctave(pn,fs,opts{:});% returns the octave spectrum of a signal x sampled at a rate fs.
hold on
poctave(pn,fs,opts{:},'Weighting','C')
hold off
legend('1/3 ocatve','C-weighted','Location','SouthWest')
```

OUTPUT-

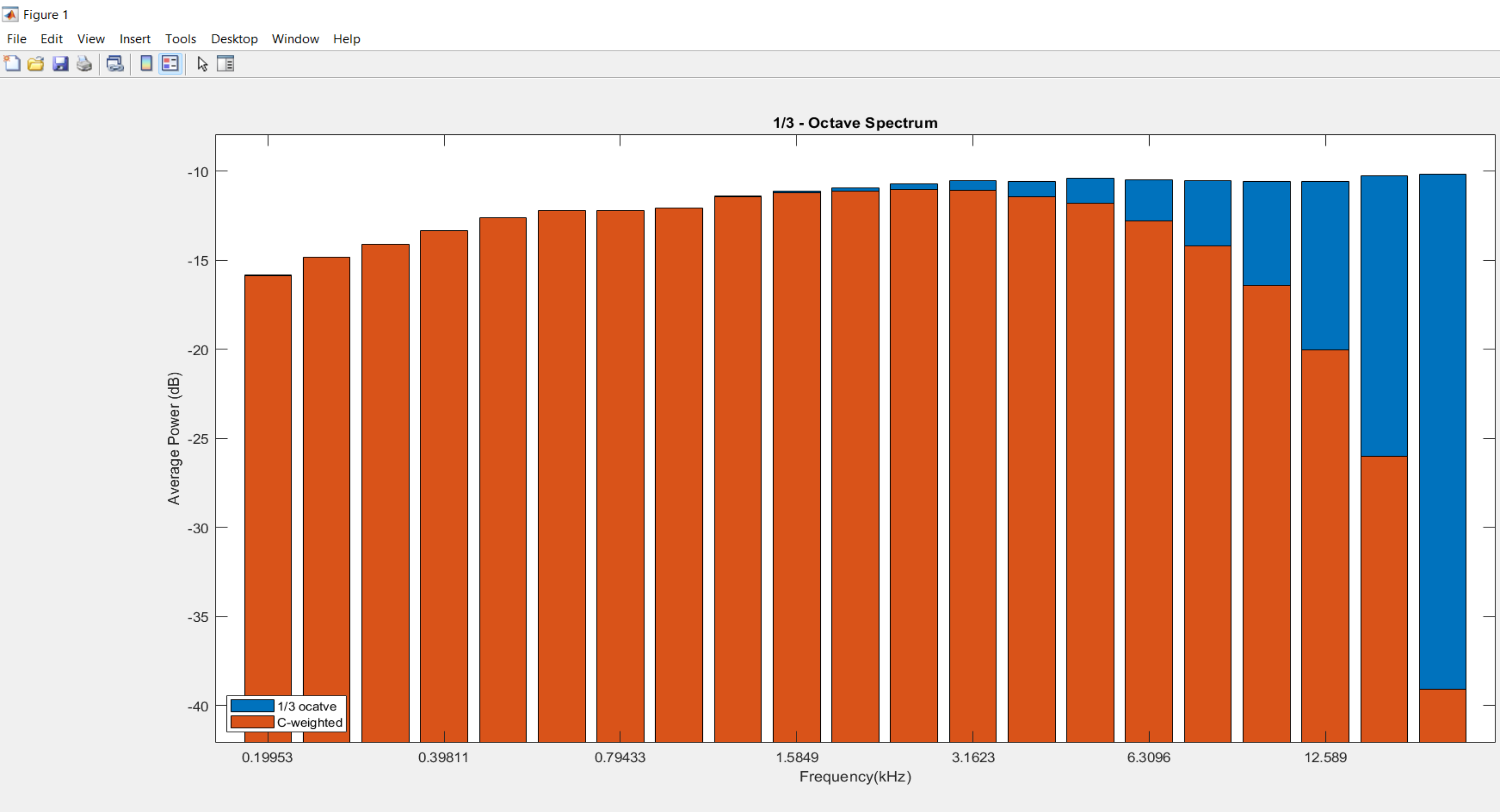

Fig.7. *Frequency(kHz) vs Average power(dB) graph for1/3 Octave CPB filter along with A weighted filter*

A weighted with 1/3 octave band filter-

MATLAB CODE-

- clc;
- clear all;
- close all;
- N = 1e5;
- fs = 44.1e3;
- wn = randn(N,1);
- z = [0.9 0.8 0.1]';
- p = [0.9 0.9 0.5]';
- [b,a] = zp2tf(z,p,1);% converts a factored transfer function representation
- %of a system to a polynomial transfer function representation
- pn = filter(b,a,wn);
- %filters the input data x using a rational transfer function defined by the
- % numerator and denominator coefficients b and a
- flims = [200 20e3];
- bpo = 3;
- opts = {'FrequencyLimits',flims,'BandsPerOctave',bpo};
- poctave(pn,fs,opts{:});% returns the octave spectrum of a signal x sampled at a rate fs.
- hold on
- poctave(pn,fs,opts{:},'Weighting','A')
- hold off
- legend('1/3 ocatve','A-weighted','Location','SouthWest')

OUTPUT-

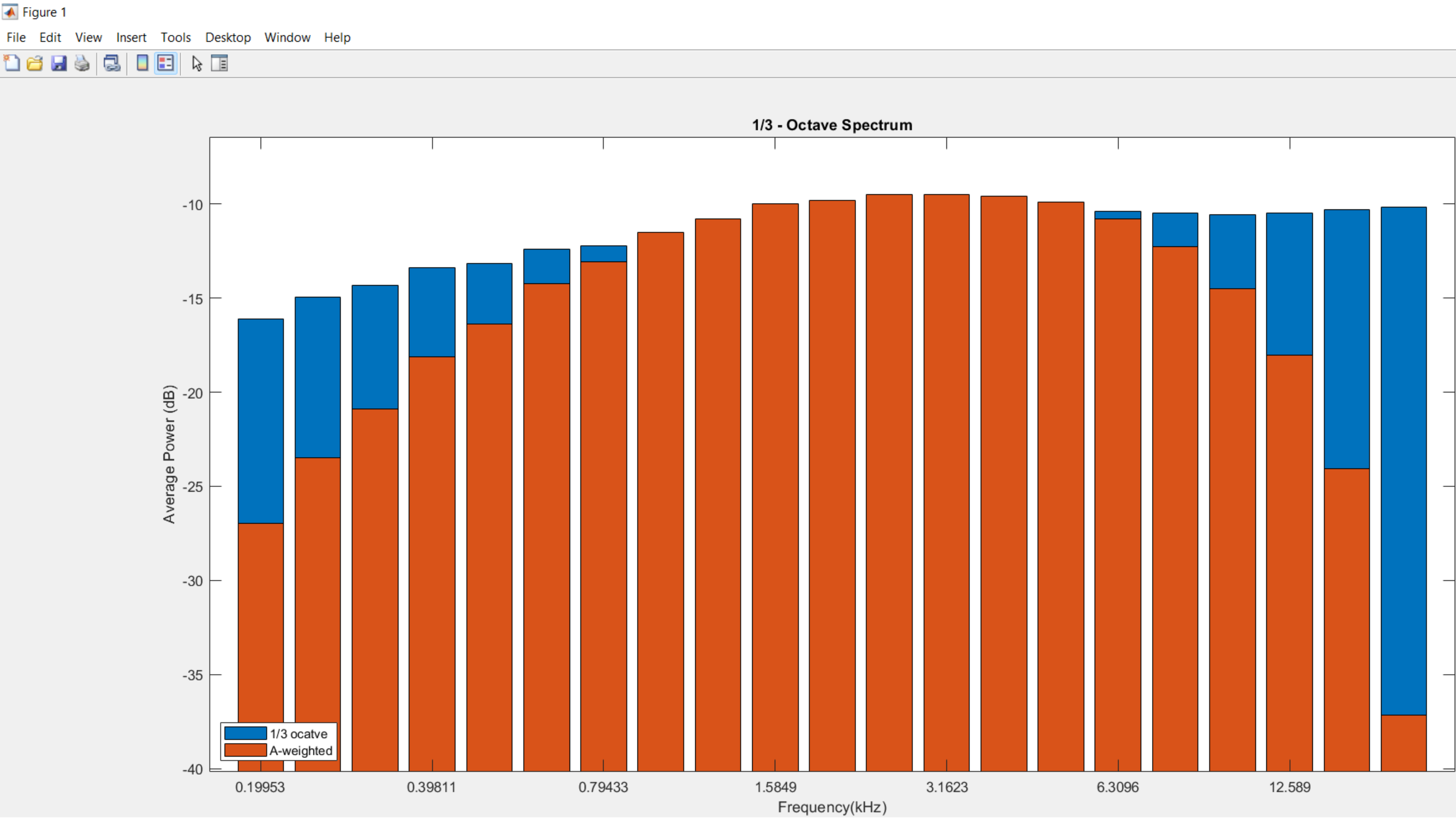

Fig.8. *Frequency(kHz) vs Average power(dB) graph for1/3 Octave CPB filter along with A weighted filter*

Inference-

1) **CPB Octave Analysis - Constant Percentage Bandwidth- Octave filter-**

- CPB filter is a filter whose bandwidth is a fixed percentage of a center frequency. This filter gives a detailed analysis of each frequency level.
- Octave Bands offer a filtering method of splitting the audible spectrum into smaller segments called octaves, allowing you to identify different sound levels across individual frequencies .

2) **1/3 Octave filter**- Each of the Octave Bands is split into three, giving a more detailed description of the frequency content of the sound.

3) **Frequency weighted filters-A-weighting**: Here for low and high frequency level the magnitude decreases.

A-weighting is applied to measured sound levels in an effort to account for the relative loudness perceived by the human ear. The human ear is less sensitive to low and high audio frequencies.

4) **C-weighting:** C-weighting is similar to A as far as the high frequencies are concerned. In the low-frequency range, it hardly provides attenuation. This weighting is used for high-level noise

Therefore the best filter for frequency in between hearing range of human would be A weighted filter used along with 1/3 Octave filter.

REFERENCE-

[1] Robinson, C. Hopkins, Effects of signal processing on the measurement of maximum sound pressure levels, Applied Acoustics, Volume 77, 2014, Pages 11-19, ISSN 0003-682X, https://doi.org/10.1016/j.apacoust.2013.09.017.

[2] Sanja Grubeša, Mia Suhanek, Antonio Petošić, Ivan Djurek, 15 - Monitoring urban noise, Editor(s): Fernando Pacheco-Torgal, Erik Rasmussen, Claes-Goran Granqvist, Volodymyr Ivanov, Arturas Kaklauskas, Stephen Makonin, In Woodhead Publishing Series in Civil and Structural Engineering, Start-Up Creation (Second Edition), Woodhead Publishing, 2020, Pages 391-417, ISBN 9780128199466, https://doi.org/10.1016/B978-0-12-819946-6.00015-1.

[3] Haijun Lin, Qiu Tang, Jing Li, Zhaosheng Teng, Huan Chen, A digital impulse-weighting method for a sound level meter, Measurement, Volume 136, 2019, Pages 511-516, ISSN 0263-2241, https://doi.org/10.1016/j.measurement.2018.12.063.

[4] B. Zhong et al., "An automatic calibration system for frequency weighting functions of sound level meter," 2010 IEEE International Conference on Mechatronics and Automation, 2010, pp. 401-405, doi: 10.1109/ICMA.2010.5589158.

[5] Y.T Kim, Y.B Lee, M.J Jho, S.J Suh, A theoretical model for the evaluation of measurement uncertainty of a sound level meter calibration by comparison method in an anechoic room, Applied Acoustics, Volume 65, Issue 10, 2004, Pages 967-984, ISSN 0003-682X, https://doi.org/10.1016/j.apacoust.2004.03.007.

[6] W.E. Scholes; A.C. Salvidge (1973). *Statistical analysis from a sound level meter. , 6(2), 111–117.* doi:10.1016/0003-682x(73)90020-

[7] D.H. Stephens, Measurement of sound insulation with a sound level meter, Applied Acoustics, Volume 9, Issue 2, 1976,Pages 131-138,ISSN 0003-682X,https://doi.org/10.1016/0003-682X(76)90004-9.