# A MINI PROJECT

## ON

**“AUDIO COMPRESSION”**

SUBMITTED TO SAVITRIBAI PHULE PUNE UNIVERSITY, PUNE, IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE AWARD OF THE DEGREE

**FINAL YEAR OF ENGINEERING (Computer Engineering)**

BY

AFRAZAHMED MOMIN COBC039

RUSHIKESH PATIL COBC046

ARSALAAN ZAIDI COBC065

UNDER THE GUIDANCE OF

**PROF. A. P. Ramdasi**



**SINHGAD ACADEMY OF ENGINEERING**

**CERTIFICATE**

This is to certify that the project report entitles

**“AUDIO COMPRESSION”**

Submitted by

AFRAZAHMED MOMIN COBC039

RUSHIKESH PATIL COBC046

ARSALAAN ZAIDI COBC065

Is a bonafide work carried out by her under the supervision of

**Prof. A. P. Ramdasi** and it is approved for the partial fulfillment of the requirement of final year computer engineering.

**(PROF.A. P. RAMDASI) (Prof.B.B.Gite)**

Guide Head

DepartmentofComputer Department of Computer Engineering Engineering

**(Dr. K.P.Patil)**

Principal,

Sinhgad Academy of Engineering Pune – 48 Place : Pune

Date :

**ACKNOWLEDGEMENT**

We extend our sincere and heartfelt thanks to our esteemed guide for her exemplary guidance, monitoring and constant encouragement throughout the course at crucial junctures and for showing us the rightway.

We would like to extend thanks to our respected ***Head of the Department B.B. Gite Sir*** for allowing us to use the facilities available. We would like to thank other faculty membersalso.

Last but not the least, we would like to thank our friends and family for the support and encouragement they have given us during the course of our work.

#### AFRAZAHMED MOMIN

#### RUSHIKESH PATIL

#### ARSALAAN ZAIDI

**IINTRODUCTION**

Digital audio compression allows the efficient storage and transmission of audio data. The various audio compression techniques offer different levels of complexity, compressed audio quality, and amount of data compression.

Compressors and limiters are used to reduce dynamic range — the span between the softest and loudest sounds. Using compression can make your tracks sound more polished by controlling maximum levels and maintaining higher average loudness.

Digital Signal Processors (DSP) take real-world signals like voice, audio, video, temperature, pressure, or position that have been digitized and then mathematically manipulate them.

Signals need to be processed so that the information that they contain can be displayed, analyzed, or converted to another type of signal that may be of use..

To check how big is the audio data and its bitrate

I Fs frames/second (e.g. 8000 or 44100) x C samples/frame (e.g. 1 or 2 channels) x B bits/sample (e.g. 8 or 16)

→ Fs · C · B bits/second (e.g. 64 Kbps or 1.4 Mbps) bits / frame frames / sec 8000 8 32 44100 CD Audio 1.4 Mbps Telephony 64 Kbps Mobile !13 Kbps

Reducing steps:

1. lower sampling rate → less bandwidth (muffled)
2. lower channel count → no stereo image
3. lower sample size → quantization noise

**Audio And Digital Signal Processing**

Frequency: The frequency is the number of times a sine wave repeats a second. In this project we have used a frequency of 1KHz.

Sampling rate: Most real world signals are analog, while computers are digital. So we need a analog to digital converter to convert our analog signal to digital. The key thing is the sampling rate, which is the number of times a second the converter takes a sample of the analog signal.

The sampling rate doesn’t really matter, as we are doing everything digitally, but it’s needed for our sine wave formula. With the use of a value of 48000, which is the value used in professional audio equipment, the sin wave formula is:

**Coding the filter**

The running mean is a case of the mathematical operation of convolution. For the running mean, you slide a window along the input and compute the mean of the window's contents. For discrete 1D signals, convolution is the same thing, except instead of the mean you compute an arbitrary linear combination, i.e. multiply each element by a corresponding coefficient and add up the results.

Those coefficients, one for each position in the window, are sometimes called the convolution kernel. Now, the arithmetic mean of N values is (x\_1 + x\_2 + ... + x\_N) / N, so the corresponding kernel is (1/N, 1/N, ..., 1/N), and that's exactly what we get by using np.ones((N,))/N.

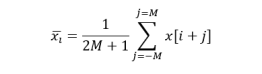
Essentially, convolution is the process of multiplying the frequency spectra of our two audio sources—the input signal and the impulse response. By doing this, frequencies that are shared between the two sources will be accentuated, while frequencies that are not shared will be attenuated. This is what causes the input signal to take on the sonic qualities of the impulse response, as characteristic frequencies from the impulse response common in the input signal are boosted.

In this particular version of the program, we have coded a low pass filter with a set cutoff frequency by using the aforementioned concept of a running mean without actually using the scipy.signals module which contains a function called lfilter which is used to implement a digital low pass filter.

### Moving Average

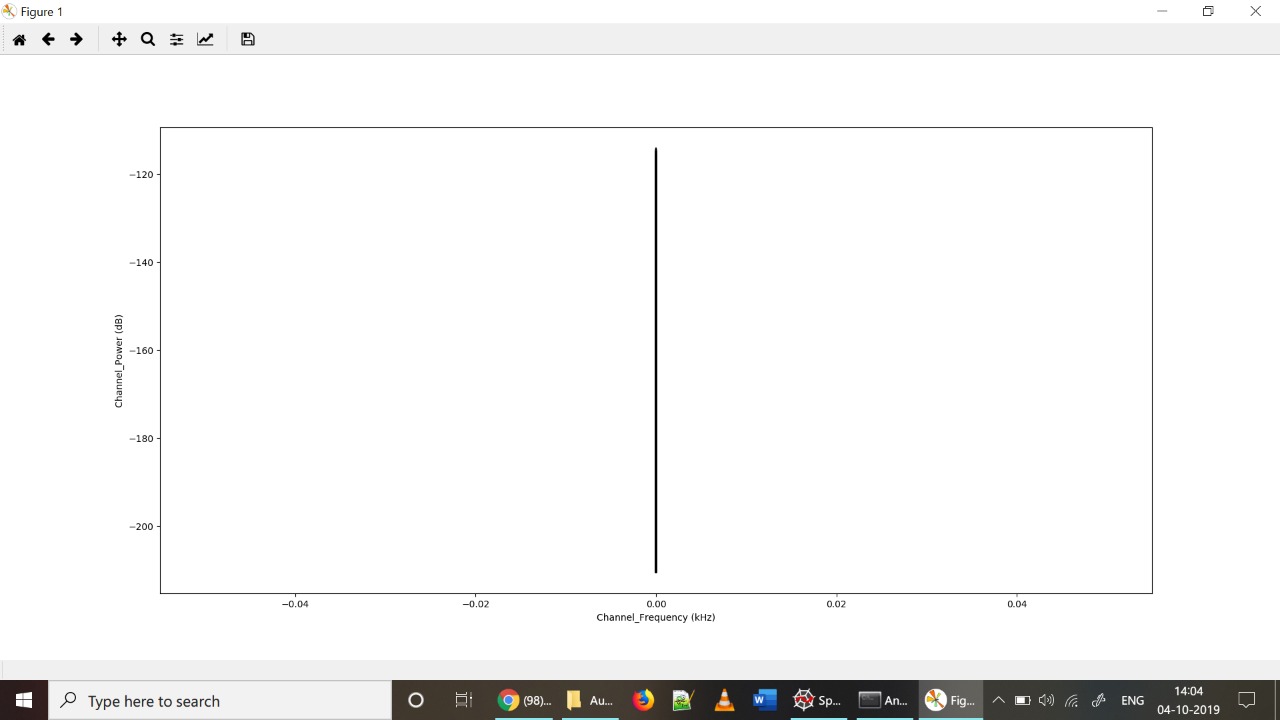
A moving average is a form of a convolution often used in time series analysis to smooth out noise in data by replacing a data point with the average of neighboring values in a moving window. A moving average is essentially a low-pass filter because it removes short-term fluctuations to highlight a deeper underlying trend.

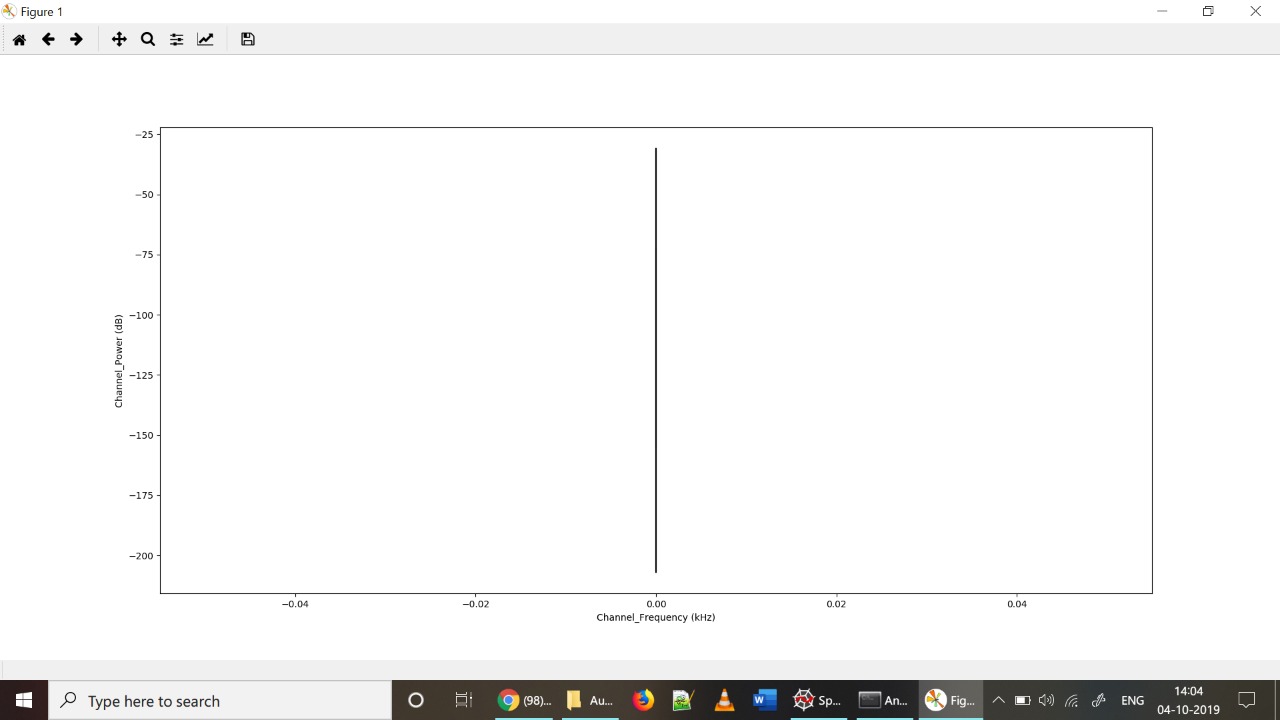
The mathematical definition of a moving average is:



where the average data point is calculated from averaging the points contained in the moving window that is defined as +/- M and centered around xi .

A moving average is commonly referred to as boxcar smoothing because it is implemented by convolving the time series signal with a box-shaped function of width 2M+1,  with an amplitude of 1/(2M+1)

****

****