

PERFORMANCE ANALYSIS AND MODELLING OF IP TELEPHONY TRAFFIC

By

¹James Agajo ²Lazarus Okechukwu Uzoechi, ³Okhaifoh Joseph, ⁴Jibril A. Al-Amin, ⁵Alegeh Prince¹Dept. of Computer Engineering

Federal University of Technology, Minna Nigeria

²Department of Electrical/Electronic,

Federal University of Technology Owerri Imo State Nigeria

³Department of Electrical/Electronic

Federal University of Petroleum Resources Effurun Delta State Nigeria

⁴Department of Electrical/Electronic Engineering

Federal Polytechnic, Auchi

⁵National Open University of Nigeria**ABSTRACT**

The world is increasingly becoming IP Centric, one of the fastest growing internet application is the Internet Protocol Telephony (IPT), which allow voice transmission over Computer Network. This project focuses on IP Telephony (IPT) traffic as one of the recent internet applications. It investigates the characteristics of measured IPT traffic on both call and packet level, with explanations to the major factors that affect voice quality in an IP network. The result shows that internet voice traffic is being affected by the Codec used, Delay by the link, jitter, packet loss, which all contribute to the degradation of voice quality. To boost the quality of service (QoS) in an IP Telephony network, the frame work of this project was built on the modeling of an Adaptive filter using LMS (least mean square) algorithm, to filter noise in an IP Telephony network. The final error e is 0.4081 which shows that noise is totally cancelled from the speech signal. Assuming the final error signal is zero, in this situation the noise signal would be completely cancelled and the far user would not hear any form of unwanted sound in the original speech returned to them.

Keywords: Internet Protocol Telephony (IPT), Delay, traffic, least mean square, IP Centric

INTRODUCTION

The world is increasingly becoming IP-centric, with a large number of devices getting networked every day. Internet Protocol Telephony (IPT), is one of the fastest growing internet applications today. It is a new communication technology that enables voice to be transmitted over a computer network. In the current networking environment, IP Telephony service is the real time delivery of packetized voice traffic across packet switched networks such as Internet and is viewed by many as simply a means to place "free" telephone calls. By the convergence of the telephone network and the Internet, telecommunications networks are gradually

driven to packet-based transmission mode, resulting in the integration of voice and data onto a single network. IPT transmission of digital voice is a logical step, which converts the voice signal from telephone into a digital signal that travels over the internet and converts it back at the other end so you can speak to anyone with a regular phone number. Also, calls can be made directly from a computer using a conventional telephone or a microphone.

The advantages of packet switched networks, such as efficiency and flexibility, make them eventually become the terminator of traditional circuit switch networks, i.e. Public Switch Telephone Network (PSTN).

IPT (Internet Protocol Telephony) is one of the successful stories about applications of packet switched networks. Delay is conventionally characterized as one-way, on the assumption that the paths in each direction

are symmetrical (in real networks, this may not be the case). The total round-trip delay for the examples given is twice the one-way delay in table I[11].

Table I: Round-Trip Delay

| Total one-way delay | Recommendation for use |
|---------------------|---|
| 0 - 150 ms | Acceptable |
| 50 - 400 ms | Acceptable for some applications |
| +400 ms | Unacceptable for general network planning |

IP TELEPHONY BENEFITS AND APPLICATION

These are the benefits of transporting voice via packet networks this are

- a) **Cost savings**
- b) **Flexibility**
- c) **Advanced features:**

Here are some examples of the advanced features provided by current IP Telephony applications. advanced call routing:

- i. unified messaging;
- ii. Integrated information systems

- iii. Long-distance toll bypass;
- iv. Voice security;
- v. Customer relationships
- vi. Telephony application services:

IMPLEMENTATION OF IP TELEPHONY

IP (Internet Protocol) telephony uses the latest innovations with popular and familiar IP protocols to make possible enhanced voice communications throughout the enterprise .It unites an organization’s many locations, including mobile Workers, into a single network.

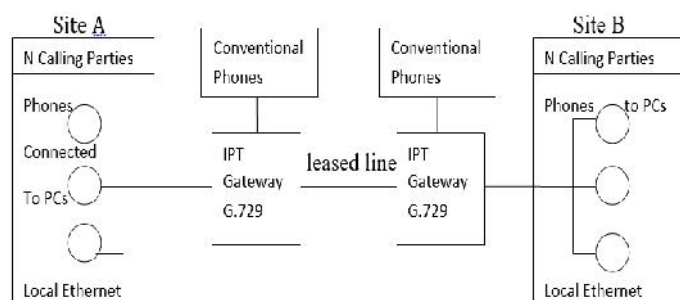


Figure 1: An Architecture for IP Telephony Deployment.

To deploy IP telephony, a server with a MAC (Media access control) address (physical address of a device on the local area network (LAN)) is typically dedicated to load the IPT software that is used to manage all the calls. Servers are just like personal computers except that they have more memory, speed, and capacity. Because the server needs a MAC address, it has a NIC (network interface card) inside to provide the MAC address as well as a physical means of connection to the LAN.

The managing server stores the database that contains all the MAC addresses, corresponding to all of the IP Telephony telephone extensions that will be assigned to end users. Depending on the size of the LAN and the number of users, other servers may be used. Switches are installed around the LAN to form the core

infrastructure of the IP Telephony LAN. These switches are set up in the telecommunication closet and they have a series of ports into which all of the other LAN-addressable devices ultimately connect.

All of the cabling typically runs from the user devices (such as the IP telephony phone, computer, and servers) to the ports on these switches. All of the switches are interconnected, usually with optical fiber.

Existing Traffic Models

To reduce the design complexity, most communication networks are hierarchically organized using a layer concept. A layer has defined interfaces, and offers a number of services to the layer above. An Example is the TCP/IP reference model, illustrated in Figure 2.

OSI REFERENCE TCP/IP REFERENCE

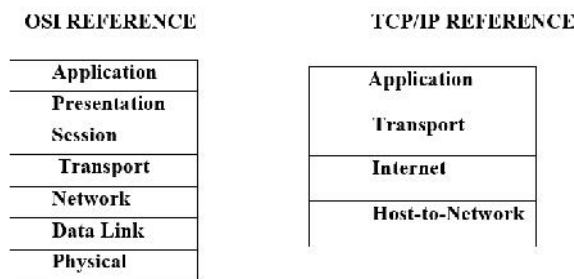


Figure 2: OSI REFERENCE TCP/IP REFERENCE

The layer on top is the **application layer**. It contains the higher level protocols such as the File Transfer Protocol (FTP), the Simple Mail Transfer Protocol (SMTP) and the Hypertext Text Transfer Protocol (HTTP). The **transport layer** offers a reliable end-to-end connection using the transmission control protocol (TCP), embedding flow control, and connection management and congestion control. The second protocol in the **transport layer** is the User Datagram

Protocol (UDP), which offers an unreliable and connectionless service.

The **Internet layer** defines a packet format and protocol called IP (Internet Protocol). Features such as packet routing and congestion avoidance are implemented in this layer. Not defined in greater detail within the TCP/IP reference is the **Host-to-Network layer**. It has to be able to inject and or to accept IP packets to or from the layer above. Due to the complexity of a communication network, it makes sense to use the

layered structure not only for designing but also for modeling a network. In the following, several examples of models dealing with mechanisms on different layers are introduced.

Modelling the Application Layer

An empirical model for traffic originating from Web browsers is proposed by Deng [Den96]. In this work, a simple ON/OFF model is developed, with the ON state describing a user who is downloading content from the Internet and no traffic generated while in the OFF state.

A traffic trace spanning 4.5 hours and containing 293 active users, each connected to a service provider with a T1 line, is used to derive the random variables describing the duration of the ON and OFF states as well as the inter arrival times, i.e. the time difference between two consecutive document downloads.

An ON period is considered a sequence of arrivals with less than 60 seconds between any two

events. If arrival A is separated by more than 60 seconds from arrival B, then A marks the beginning of an OFF period. Two OFF periods divided by a single event are considered a single OFF period. Although the 60 second time period is chosen heuristically, the resulting distributions are, referring to Deng, insensitive to time periods on the order of 30 to 120 seconds. Deng found that the ON period and the inter-arrival time of documents can be described by a Weibull distribution, while the OFF period can be modeled by a Pareto distribution. Sizes of the transferred files are not examined but taken from [18]. The distributions and their parameter are summarized in Table 2

With these parameters, both, ON and OFF times are heavy tailed. Problems arise due to the infinite mean and variance of the OFF distribution: It is, for example, not possible to derive the average time spent in the ON state.

Table 2: Parameter of Deng's Empirical Model

| Period | Distribution | Model (PDF) | Model (CDF) |
|--------------|--------------|--|--|
| ON period | Weibull | $p(x) = x^{\beta-1} e^{-\alpha x^\beta}$ $\beta = 0.88, \alpha = e^{-1.5\beta}$ | $F(x) = 1 - e^{-\alpha x^\beta}$ |
| Interarrival | Weibull | $p(x) = \alpha \beta x^{\beta-1} e^{-\alpha x^\beta}$ $\beta = 0.5, \alpha = e^{-1.5\beta}$ | $F(x) = 1 - e^{-\alpha x^\beta}$ |
| OFF period | Pareto | $p(x) = \frac{\beta \alpha^\beta}{x^{\beta+1}}$ $\beta = 0.5, \alpha = 60$ | $F(x) = 1 - \left(\frac{\alpha}{x}\right)^\beta$ |

A more detailed model for Web traffic is introduced by Choi and Lim in Trang et al, (1998); its basic structure is illustrated in Figure 14. A Web request occurs when a user clicks on a link or enters a URL Trang et al, (1998). The first object transferred is referred to as main object, most likely a HTML document containing references for pictures or a java script applet. These documents, whose download is initiated by the browser, are called in-line objects, each using its

own TCP connection, which can be either back-to-back or in parallel, depending on the HTML protocol version. The authors utilize a traffic trace containing HTML and TCP header information to derive the statistical parameters used to describe the model, such as main and in-line object sizes and inter-arrival times, parsing time, page viewing time and number of Web requests. The state transition diagram for Web traffic generation is given as shown figure 3.

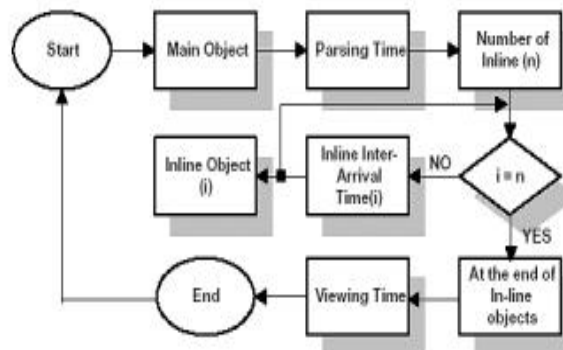


Figure 3: Choi's State Transition Diagram for Web Traffic Generation

When a new Web request is initiated by the user, the main object is loaded, containing the information for the browser on how to proceed (e.g. from where to load the inline objects). After the parsing time (the time needed by the browser to process the main object), the inline objects are loaded. When all downloads are completed, the user usually works with the page for a certain time, introduced as viewing time. The process starts again with a new Web request.

Referring to Choi and Limb, the model generates traffic which closely matches traffic patterns measured on a real network.

NOISE FILTRATION TECHNOLOGIES

It has being found [6] that noise filtration technologies could be categorized into the following groups:

1. Out of phase Noise filtration, which can be acoustic, or electronic (passive or active).
2. Array/Beam steering
3. Multiple band-pass filtering/expansion/gating
4. Adaptive filtering.

The first two belong to category 1 (spatial filter) techniques, and the third is a category 2 (frequency filter) techniques.

(1) **Out of phase Filtration:** This is perhaps the most commonly used noise cancelling technique. Usually two

microphones are used. One microphone picks up primarily the signal of interest, whereas the other picks up primarily environmental noise. Noise microphone signal's phase is inverted and subtracted from the speech microphone signal. However, noise sources with small angular separation (30-45 degrees) cannot be distinguished. The technology is mostly applicable to close talking microphones. Deficiencies and distortions can result from the subtraction and in some cases it increases the noise content.

(2) **Array/ Beam steering:** Although this signal processing technique has been around for quite some time and in use in radar and ultrasound systems, it has recently been introduced for consumer use in microphones. The technique requires many microphones, typically 10-40. The technology uses patterns of sums of delayed microphone signals to create a desired polar response of a directional microphone. Thus, it, too suffers from the same deficiencies as the out-of-phase cancellation technique. Typical noise reduction offered by this technology is around 12 dB. High frequency noise sources with small angular separation (30-45degrees) can be distinguished.

(3) **Multiple band pass:** Multiple band pass processing technology isolates signals based on spectral content. The technology requires a single microphone input. Sophisticated spectral models of the signal of interest

are formulated and the spectra present in the received sound that most closely match that model are isolated, which is used to re-construct the signal. However, portions of the signals that overlap in spectrum and in time cannot be distinguished from one another. This approach results in substantial distortions. Moreover, if the interfering noise source is another speaker's voice, this technique does not provide much improvement.

(4) **Adaptive filtering:** Adaptive filtering can be considered as a process in which the parameters used for the processing of signals changes according to some criterion. Usually the criterion is the estimated mean squared error (how well a system can adapt to a given solution) .The adaptive filters are time-varying since their parameters are continually changing in order to meet a performance requirement Mark A(2003). The aim of an adaptive filter is to calculate the difference between the desired signal and the adaptive filter output, $e(n)$. This error signal is fed back into the adaptive filter and its coefficients adapt to cause the error signal to be a noiseless version of the signal $s(n)$ Trang et al ,(1998). The collection of traffic traces on individual clients is possible with the use of a packet capturing software *tcpdump*. *Tcpdump* is a command-line tool for filtering and capturing network traffic from protocols such as IP, ICMP14, ARP15, ARP16, UDP17 and TCP. It is possible to specify the protocols and ports of interest *tcpdump* is supposed to listen to, as well as the detail of information which is recorded.

The general equation for this algorithm is shown below

$$\mathbf{w}(n+1) = \mathbf{w}(n) + 2\mu e(n)\mathbf{X}(n) \quad [3.0]$$

3.2.2 NORMALISED LEAST MEAN SQUARE (NLMS) ALGORITHM.

One of the primary disadvantages of the LMS algorithm is having a fixed step size parameter for every iteration. This requires an understanding of the

statistics of the input signal prior to commencing the adaptive filtering operation. In practice this is rarely achievable. Even if we assume the only signal to be input to the adaptive noise cancellation system is speech, there are still many factors such as signal input power

But for the interest of this project, traffic traces were collected from a corporate company in Lagos, with an IP Telephony network which consists of approximately 200 IP phones. The analysis was based on 'Call Detail Records' (CDRs), generated by a Cisco Call Manager Release 3.3(2) system. This type of information can be used to post-processing activities such as generating billing records and network analysis. CDRs include 51 fields such as IP address and port number of originating and destination stations, calling and called party numbers, type of used codecs, timestamp of connections, disconnections, and duration of calls. The call arrival and duration information, which are relevant to call-level modeling, were derived from the CDR database and gave the input for my statistical analysis.

LEAST MEAN SQUARE ALGORITHM

The LMS algorithm is one of the most widely used algorithm. It is well known and widely used due to its computational simplicity. It is this simplicity that has made it the benchmark against which all other adaptive filtering algorithms. A parameter called the step size is a small positive constant that controls the change of the filter characteristic. Selection of a suitable value for μ is imperative to the performance of the LMS algorithm, if the value is too small the time the adaptive filter takes to converge on the optimal solution will be too long; if μ is too large the adaptive filter becomes unstable and its output diverges (Micheal H,(2003).

statistics of the input signal prior to commencing the adaptive filtering operation. In practice this is rarely achievable. Even if we assume the only signal to be input to the adaptive noise cancellation system is speech, there are still many factors such as signal input power

and amplitude which will affect its performance. The normalised least mean square algorithm (NLMS) is an extension of the LMS algorithm which bypasses this issue by selecting a different step size value, $\mu(n)$, for each iteration of the algorithm. This step size is

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \frac{1}{\mathbf{x}^T(n)\mathbf{x}(n)} \epsilon(n)\mathbf{x}(n)$$

3.2.3 RECURSIVE LEAST SQUARE (RLS) ALGORITHM

The other class of adaptive filtering techniques studied in this project is known as Recursive Least

proportional to the inverse of the total expected energy of the instantaneous values of the coefficients of the input vector $\mathbf{x}(n)$ [8].

The recursion formula for the NLMS algorithm is stated below
cost of an increased computational complexity and some stability problems.
This algorithm is very costly to implement, and it does not perform well in practical situations.

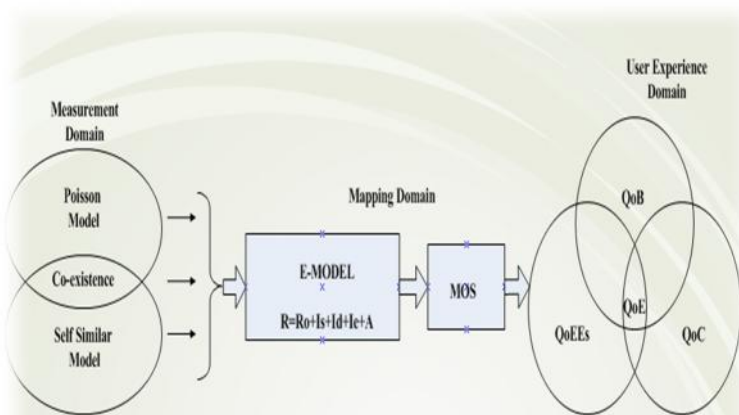


Figure 4: Domains, Models and Realms of IP Telephony

Squares (RLS) algorithms. Unlike the LMS algorithm and its derivatives, the RLS algorithm directly considers the values of previous error estimations. RLS algorithms are known for excellent performance when working in time varying environments... these advantages come with the

E-MODEL

Speech quality measurements can be divided into two categories, subjective measurements and objective measurements. Under the Objective measurement, we have the intrusive and non-intrusive speech quality measurement. The E-model is a non-intrusive objective speech quality measurement, as it is parameter based, and does not require the help of original signal Zhuogun L.(2013)

SYSTEM DESIGN

This Section is divided into two sections, the first, gives an analysis of the performance evaluation of voice quality in IP Telephony systems using E-MODEL with respect to the data table collected from the technical department of Visafone Network at Lagos State. The figure in figure 4. shows Domains, Models and Realms of IP Telephony

The E-Model that is the abbreviation of "European Telecommunications Standards Institute (ESTI) Computational Model" is a computational model. It is different from other methods because it represents also a network simulation tool. It uses transmission parameters to predict the subjective speech quality of packetized voice. E-Model has proven to be useful as a transmission-planning tool, for assessing the combined effects of variations in several transmission parameters

that affect conversational quality of telephony Alcatel,(2004). The primary output from the E-Model is the "Rating Factor" R, and R (which is used for evaluating the quality perceived by the network) can be further transformed to give estimates of customer opinion by mapping it to the MOS (Mean Opinion Score) scale. The MOS provides a numerical measure of the quality of human speech at the receiving end and values range from 1 (poor) to 5 (best) [4] as presented in Table 3.

The E-Model Equation for "Rating Factor" is

$$R=R_0-I_s-I_d-I_e+A \dots\dots\dots(2)$$

- a) **R₀** represents the signal to noise ratio, including the noise generated by the circuit and the background noise;
- b) **I_s** factor is obtained from the combination of all damage factors that almost simultaneously affect the voice signal;
- c) **I_d** coefficient represents the damages caused from delay;

- d) **I_e** coefficient represents the damages caused from codecs with low bit rate or effect of equipment;
- e) **A** represents the advantage factor used to compensate for the allowance users make for

$$MOS = \begin{cases} 1 \\ 1+0.035R+R(R-60)(100-R)7 \times 10^{-6} \\ 4.5 \end{cases}$$

poor quality when given some additional convenience (e.g. 0 for wire line and 10 for GSM).

The R factor result is represented from a scale that goes from 10 to 100, but typically the range used goes from 50 to 90 as it is possible to view in the Table 3 where S represent Satisfied and DS Dissatisfied. The graphical representation is shown in Fig below

Table 3: R factor quality issues

| R-value Range | 100-90 | 90-80 | 80-70 | 70-60 | 60-50 |
|--------------------------------------|--------|-------|---------------|---------------|---------------------|
| Speech Transmission Quality Category | Best | High | Medium | Low | (very) Poor |
| User's Satisfaction | Very S | S | Some Users DS | Many users DS | Nearly all users DS |

MAPPING R FACTOR INTO MOS SCALE

We can map R into MOS scale by the following equations as defined by ITU (international telecommunication union).

For $R \leq 0$

For $0 < R < 100$

For $R \geq 100$

(3)

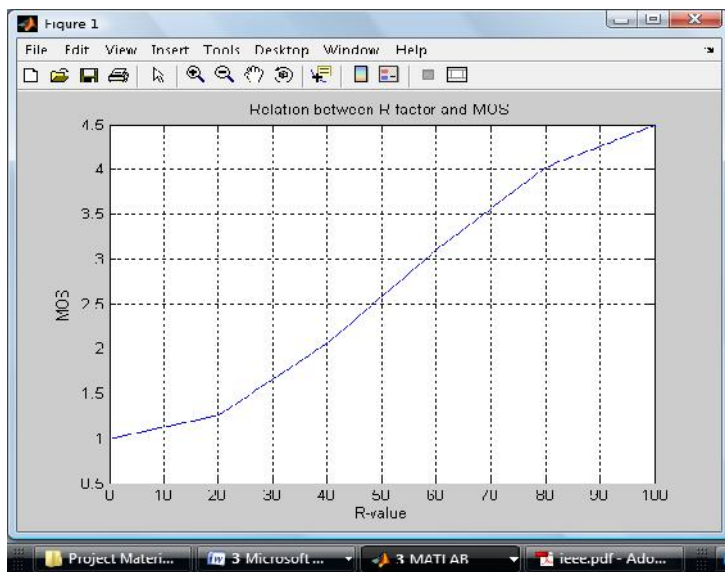


Figure 5: Relationship between R factor and MOS

ADAPTIVE NOISE FILTRATION

In this section, LMS-adapted FIR filter is used to cancel undesired noise from a voice signal. The most common problem in speech processing is the effect of interference noise in speech signals. Interference noise masks the speech signal and reduces its intelligibility Mohammed et al, (2005) Adaptive filtering technique is a method used for speech denoising.

ADAPTIVE FILTERING

Basically filtering is a signal processing technique whose objective is to process a signal in order to manipulate the information contained in the signal. In other words, a filter is a device that maps its input signal into another output signal by extracting only the desired information contained in the input signal. Adaptive filters are dynamic filters which iteratively alter their characteristics in order to achieve an optimal desired output. An adaptive filter algorithmically alters its parameters in order to minimise a function of the

difference between the desired output $d(n)$ and its actual output $y(n)$ Vijaykumar and Vanathi ,(2007.)

Adaptive filtering can be considered as a process in which the parameters used for the processing of signals changes according to some criterion. Usually the criterion is the estimated mean squared error. The adaptive filters are time-varying since their parameters are continually changing in order to meet a performance requirement. Usually the definition of the performance criterion requires the existence of a reference signal that is usually hidden in the approximation step of fixed-filter design.

ADAPTIVE NOISE FILTER

Two input signals, $d(k)$ and $x(k)$, are applied simultaneously to the adaptive filter. The input $x(k)$, a noise source $n(k)$, is compared with a desired signal which consists of a signal $s(k)$ corrupted by another noise $n(k)$. The adaptive filter coefficients adapt to cause the error signal to be a noiseless version of the signal $s(k)$.

Both of the noise signals for this configuration need to be uncorrelated to the signal $s(k)$. The noise sources must be correlated to each other in some way preferably equal, to get the best results.

Due to the nature of the error signal, the error signal will never become zero. The error signal should converge

to the signal $s(k)$, but not converge to the exact signal. In other words, the difference between the signal $s(k)$ and the error signal $e(k)$ will always be greater than zero. The only difference is to reduce the difference between these two signals Trang et al (1998.)

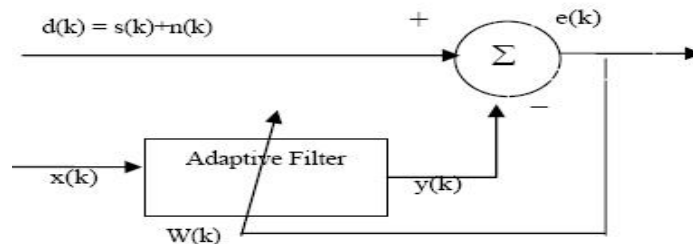


Figure 6 Adaptive Noise Filter

PERFORMANCE MEASURES IN ADAPTIVE SYSTEM

Seven performance measures will be discussed namely Trang et al, (1998): Step size, convergence rate, minimum square error, computational complexity, stability, robustness, and filter length.

Step Size

The step-size directly affects how quickly the adaptive filter will converge toward the unknown system. If it is very small, then the coefficients change only a small amount at each update, and the filter converges slowly.

With a larger step-size, more gradient information is included in each update, and the filter converges more quickly; however, when the step-size is too large, the coefficients may change too quickly and the filter will diverge.

Convergence Rate

The convergence rate determines the rate at which the filter converges to its resultant state. Usually a faster convergence rate is a desired characteristic of an adaptive system. There will be a tradeoff, in other

performance criteria for an improved convergence rate. For instance, if the convergence rate is increased, the stability characteristic will decrease and vice versa. The condition for convergence is step size $< 1/\text{number of weights times reference noise power}$.

Minimum Mean Square Error

The objective of the LMS adaptive algorithm is to reduce the mean square error to minimum. The mean square is a metric indicating how well a system can adapt to a given solution. A small minimum MSE indicates that the adaptive system has accurately converged to a solution for the system. A very large MSE shows that the adaptive filter cannot accurately converge to the solution of the system.

Computational Complexity

The choice of the adaptive algorithm to be applied is always a tradeoff between computational complexity, fast convergence and stability. A high complex algorithm will require much greater hardware resources than a simplistic algorithm.

Stability

This is the one of the most important performance measure, the step size (convergence factor) controls the stability, when the value is too high, the system because instable this can be improved by using adaptive algorithm by averaging (AFA). Generally, the stability is dependent on factors like the convergence rate, step size and transfer function of the system (the poles of the system must not go outside the unit circle).

LEAST MEAN SQUARE ALGORITHMS

The Least Mean Square (LMS) algorithm was first developed by Widrow and Hoff in 1959 through their studies of pattern recognition. The Least-Mean Squares (LMS) adaptive algorithm has been used for over 35 years as the center piece of a wide variety of adaptive algorithms Sriniviasaprasath (2003)

$$y(n) = w^T(n)x(n) \tag{4}$$

and

$$x(n) = [x(n), x(n-1), x(n-2), \dots, x(n-(N-1))] \tag{5}$$

where $w^T(n) = [w_0(n), w_1(n), w_2(n), \dots, w_{N-1}(n)]$ are the time domain coefficients for an N^{th} order FIR filter.

The super script **T** represents the transpose of a real valued vector or matrix.

$$W(n+1) = w(n) - \mu \nabla \xi(n)$$

where

$W(n+1)$ = new coefficient values for the next time interval

μ = convergence factor or step size

$\nabla \xi$ = is the ideal cost function with respect to $w(n)$

Where $\xi(n) = e^2(n)$;

$$e(n) = d(n) - y(n);$$

and $y(n) = x^T(n)w(n)$

The gradient of the cost function, $\nabla \xi(n)$, can alternatively be expressed in

Robustness

The robustness of a system is directly to the stability of a system. Robustness is measure of how well the system can resist both input and quantization noise.

Filter Length

The filter length is dependent on the other performance measures mentioned above like the convergence rate, increasing or decreasing the computational time, stability of the system, at certain step sizes, and the minimum mean square error. If the filter length is increased, the number of computations will increase, decreasing the maximum convergence rate. On the other, if it decreases, the number of computations will decrease, increasing the maximum convergence rate. Agajo et al, (2012)

LEAST MEAN GRADIENT APPROXIMATE METHOD

Given an adaptive filter with an input $x(n)$, an impulse response $w(n)$ and an output $y(n)$ you will get a mathematical relation for the transfer function of the system.

the following form.

$$\begin{aligned} \nabla \xi(n) &= \nabla (e^2(n)) \\ &= \partial e^2(n) / \partial w \\ &= 2e(n) (\partial e(n) / \partial w) \\ &= 2e(n) (\partial (d(n) - y(n)) / \partial w) \\ &= -2e(n) (\partial e(n) / \partial w) \\ &= -2e(n)x(n) \end{aligned}$$

Substituting this into the steepest descent algorithm of equation, we arrive at the recursion for the LMS adaptive algorithm

$$w(n) = w(n) + \mu e(n)x(n) \quad (6)$$

In the above equation μ is sometimes multiplied by 2, but here it will assume it is absorbed by the μ factor.

SYSTEM SPECIFICATION

For this research work, the parameter and values assigned to each value is shown below.

Filter Length = 40

Step size = 0.01

Number of iterations = 1000.

From the values received,

- Delay was set on 0,190,430,690 and 930 ms,
- Packet loss value were 0.2 or 4%.
- The filter convergence factor/Step size = 0.01

Table 5.0 Average values of MOS and MSE for different Delay and Packet loss parameters.

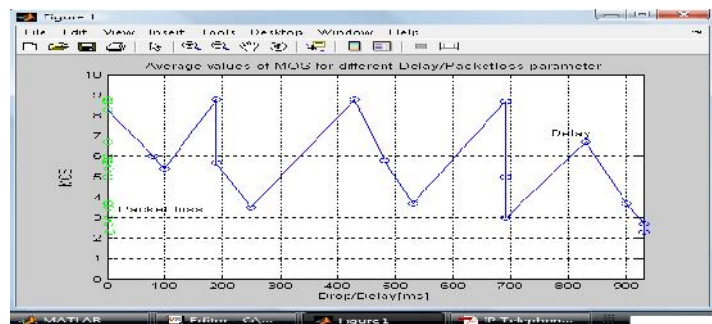


Figure 7: A plot of MOS against Delay/Packet loss.

It is shown that in this type of test, MOS values are decreasing as delay and packet loss parameters are rising.

Each iteration of the LMS algorithm requires 3 distinct steps in this order:

1. The output of the FIR filter, $y(n)$ is calculated using equation

$$y(n) = \sum_{i=0}^{N-1} w(n)x(n-i) = \mathbf{w}^T(n)\mathbf{x}(n) \tag{5}$$

the value of the error estimation is calculated using equation

$$e(n) = d(n) - y(n) \tag{6}$$

The tap weights of the FIR vector are updated in preparation for the next iteration, by equation

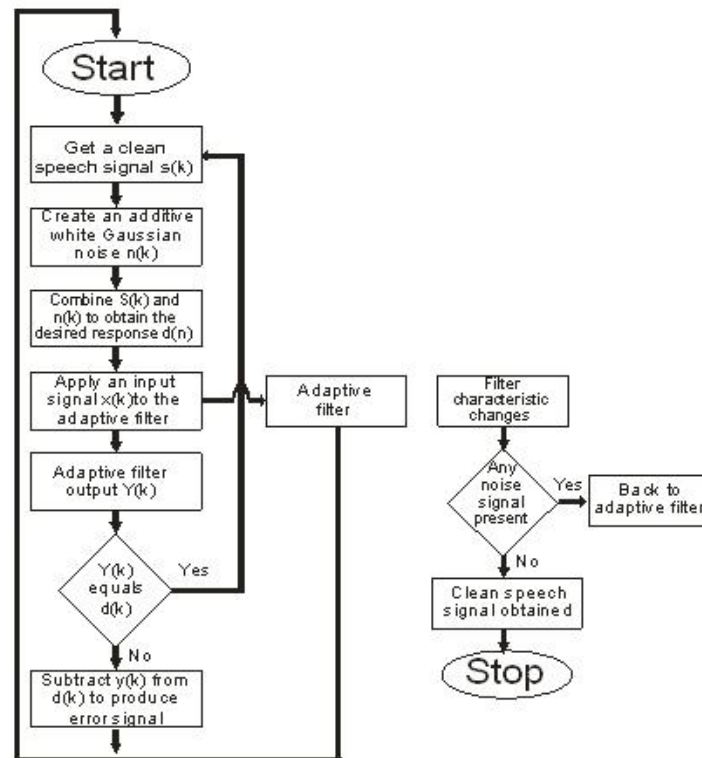
$$\mathbf{w}(n+1) = \mathbf{w}(n) + 2\mu e(n)\mathbf{x}(n) \tag{7}$$

SIMULATION AND RESULT

The previous chapter provides a detailed sketch of an Adaptive Noise filter. In this chapter the flowchart for the software simulation and the results of simulation of the Adaptive noise cancellation algorithm, which was performed in MATLAB are discussed. The idea that drove the simulation was to show that convincing results could be achieved in the software environment.

SYSTEM DEVELOPMENT

This section shows the test plan which is the simulation flowchart and software testing (simulation). The flowchart for the simulation of the noise canceller algorithm is presented in Figure 5.0



1

MATLAB

| S/N | Delay(ms) | Packet Loss(ms) | MOS Values | MSE Values |
|-----|-----------|-----------------|------------|------------|
| 1 | 0 | 0 | 8.3 | 0 |
| 2 | 0 | 2 | 6.0 | 0.17 |
| 3 | 0 | 4 | 5.0 | 0.25 |
| 4 | 190 | 0 | 8.7 | 0.32 |
| 5 | 190 | 2 | 5.7 | 0.42 |
| 6 | 190 | 4 | 3.3 | 0.88 |
| 7 | 430 | 0 | 8.7 | 1.00 |
| 8 | 430 | 2 | 5.3 | 1.35 |
| 9 | 430 | 4 | 3.7 | 1.50 |
| 10 | 690 | 0 | 8.7 | 1.87 |
| 11 | 690 | 2 | 5.0 | 2.20 |
| 12 | 690 | 4 | 3.0 | 2.83 |
| 13 | 930 | 0 | 6.7 | 3.33 |
| 14 | 930 | 2 | 3.7 | 3.53 |
| 15 | 930 | 4 | 2.7 | 3.70 |
| 16 | 930 | 6 | 2.3 | 4.00 |

Figure 8: Flow Chart of MATLAB Simulation

SIMULATION OF ADAPTIVE LMS FILTERING.

Additive white Gaussian noise was generated in MATLAB using the command randn and added to the original speech signal $s(n)$. Random white noise provides signal at all digital frequencies to train the $s(n) = \sin [2\pi n / 10 \sin(\pi (1000n) / 3000)]$ [14]

adaptive filter, the LMS algorithm parameters used for this simulation are convergence factor of 0.01, Filter length = 40 and Number of iterations = 1000. The clean speech signal $s(n)$ equation is given below.

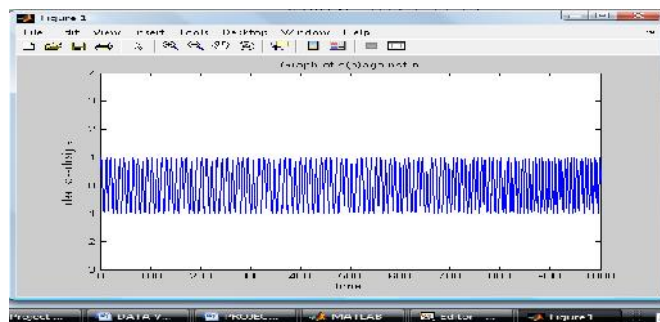


Figure 9: Plot of Clean speech signal against time

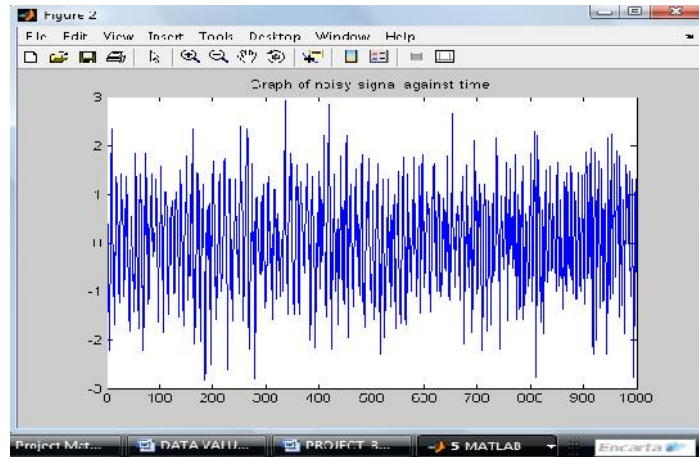


Figure 10: Plot of noisy signal against time

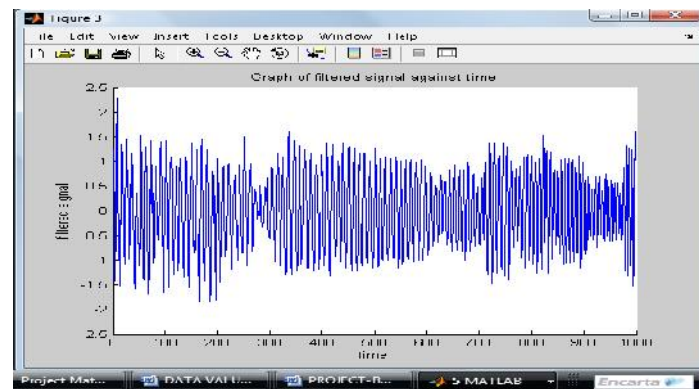


Figure 11: Plot of filtered signal against time

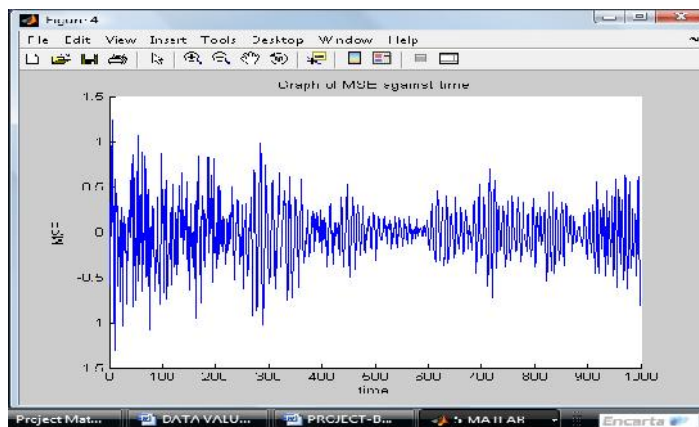


Figure 12: Plot of MSE against time

PERFORMANCE EVALUATION

Figure 9, is a plot of clean speech signal against time, while figure 10 is a plot of noisy signal against time, in this plot, noise has being superimposed

on the clean speech signal. Figure 11 Plot of filtered signal against time, here, noise has being filtered out. Figure 12 is a Plot of MSE against time, in this plot after 1000 iteration the mean square error has being greatly

reduced. From the graph above, figure 10 is the clean speech signal without any form of input noise (white noise).figure 11 shows the graph of the clean speech signal superimposed with an input noisy frequency varying sine wave, after 1000 iterations of the filter, in figure 12, the noise is considerably reduced. The MSE shows that as the algorithm progresses, the average value of the cost function decreases, this corresponds to the LMS filters impulse response converging to the actual impulse response, more accurately emulating the desired signal and thus more effectively canceling the noise signal. The final error e is 0.4081 which shows that noise is totally cancelled from the speech signal. Assuming the final error signal is zero, in this situation the noise signal would be completely cancelled and the far user would not hear any form of unwanted sound in the original speech returned to them. In comparing figure 11 and 12, a conclusion can be drawn that noise can only be reduced to a minimal level.

IMPLEMENTATION OF ADAPTIVE NOISE FILTRATION.

The codec (the coder/decoders that convert analog voice to digital packets) used to stream the voice packet to the network is configured with the noise cancellation algorithm (LMS), to reduce the noise or preferably, we have purpose-built voice enhancement device (VED). The primary advantage of a VED is its integration of multiple voice quality enhancement features, including adaptive noise filtration, acoustic and hybrid echo filtration.

CONCLUSION.

Since the Quality of Service in IP Telephony networks is degraded by the various factors including noise, suitable technique is required to enhance QoS. The Noise Filtration algorithm presented in this research successfully attempted to find a software solution for the problem of noise in the telecommunications environment. The proposed algorithm was completely a

software approach without utilizing any DSP hardware components. The algorithm was capable of running in any Computer with MATLAB software installed.

REFERENCES

- Agajo, J., Idogho, P, Idigo, V. E. (2012). Modelling and Simulation of Voice over Internet Protocol (VOIP), International Journal of Engineering and Innovative Technology (IJEIT) Volume 1, Issue 4,
- Dennis, H. (2003). *Noise and Voice Quality in VoIP Environment*. Agilent Technologies, Inc., US..
- Ejaz Mahfuz; Packet Loss Concealment for Voice Transmission over IP Networks, McGill University, Montreal Canada.
- Floriano, De R., Mauro T., Peppeno F., & Salvatore M. (2006). Subjective and Objective Measurement Methods, International journal of Computer Science and Network Security, January 2006.
- Georgi, L. and Nikola, K. (1995). *Adaptive filtering with Averaging in Noise Cancellation for Voice and Speech Recognition*, Department of Information Science, University of Otago P.O Box 56, Dunedin, New Zealand..
- ictech@ic-tech.com (1999). *Competitive Analysis of Voice Extraction for Speech Signal Quality Enhancement*, IC Tech, Inc., Michigan USA, 1999.
- Mark A. C. (2003). Investigations into using Matlab for Acoustical Adaptive filtering, *Annual Research Journal*1(1).
- Micheal H. (2003). *Acoustic Echo Cancellation using Digital*, John Wiley and sons, New York, November 2003.
- Mohammed A. A, Ali A., Farokh A & Marvasti (2005). *Speech Enhancement by Adaptive Noise Cancellation in wavelet Domain*, IEEE Press, 2005.

MathWorks. (2001). *Introduction to MATLAB*. The Math Works, Inc. Available:

http://www.mathworks.com/access/helpdesk/help/techdoc/learn_MATLAB_chintro.shtml#12671.