Addis Ababa University
Addis Ababa Institute of Technology
School of Electrical and Computer Engineering

Characterizing and Modeling WAN Egress Traffic

by

Rahel Abera Aboset

A Thesis Submitted to
The School of Electrical and Computer Engineering

Presented in Fulfilment of the Requirements for the Degree of Master of Science in
(Computer Engineering)

Addis Ababa University
Addis Ababa, Ethiopia
May 2018

Addis Ababa University

Addis Ababa Institute of Technology

School of Electrical and Computer Engineering

Characterizing and Modeling WAN Egress Traffic

by

Rahel Abera Aboset

Approval by Board of Examiners

Chairman Department of
Graduate Committee _____ signature _____ Date _____

Advisor Dr.Yalemzewd Negash Signature _____ Date _____

Internal Examiner _____ Signature _____ Date _____

External Examiner _____ Signature _____ Date _____

May 2018

Addis Ababa, Ethiopia

Declaration

I, the undersigned, declared that this MSc. Thesis is my original work, has not been presented elsewhere for assessment and all sources and materials used for the thesis have been acknowledged.

Name: Rahel Abera

Signature: _____

Place: Addis Ababa, Ethiopia

Date of Submission: May, 2018

This thesis has been submitted for examination with my approval as a university advisor

Advisor: Dr. Yalemzewd Negash

Signature: _____

Abstract

Computer networks exhibit complex characteristics due to the heterogenous nature of traffic running through the network. This makes the design of reliable networks and network services difficult. To have a design of robust and reliable networks, a detailed understanding of traffic characteristics of the network is needed which will lead to distinguish the traffic model it fits.

In this paper, it is showed that the WAN egress traffic possess self-similar characteristics, using different mathematical techniques. And also, the presence of long memory in WAN egress traffic is shown by the Autocorrelation Function of the trace. Additionally, it is showed that one of the self-similar long memory models, Fractional Auto-Regressive Integrated Moving Average (FARIMA) model, best capture the collected WAN traffic data. To model the traffic data first stationarity was tested using Augmented Dickey Fuller (ADF) test. The AR and MA terms of the model are estimated using the ACF and PACF plot. To test the model, Autocorrelation function is used, and it is found that the Autocorrelation function of the approximated data has a resemblance to the Autocorrelation function of the collected data.

Key words: - WAN, Self-similarity, Long Range Dependence, Traffic models, FARIMA,

Acknowledgment

First of all, I would like to thank God for letting me to start and finish my MSc class. Next my appreciation goes to the female scholarship program which encourage females to be competent on the industry. I would also like to express my gratitude towards my advisor Dr. Yalemzewd Negash, for his guidance and support throughout this work. Also, I would like to Thank Wegagen bank for their willingness to support me. And am so grateful for the IT department of the bank, Daniel A. and Yonas A., for their great cooperation.

I would like to give deep acknowledgement to my dearest friends Muluken S. and Eldana M; they have been very caring and supportive and also they were with me all the time specially during stressful times. My class mates Seni, Samri and Brx, many thanks for you too. At last, I would like to Thank mom, dad, grandma, sisters Elsa and Mekdi, Sol and all my families for their support and encouragement in every step of my life. They let me to achieve this stage of life.

Table of Contents

List of figures

List of tables

Acronyms

Acronym	Description
LAN	Local Area Network
WAN	Wide Area Network
ISP	Internet service provider
LRD	Long Range Dependence
SRD	Short Range Dependence
AR	Auto Regressive
MA	Moving Average
ARMA	Auto Regressive Moving Average
ARIMA	Auto Regressive Integrated Moving Average
FARIMA	Fractional Auto Regressive Integrated Moving Average
ACF	Auto Correlation Function
PACF	Partial Auto Correlation Function
Cisco SDM	Cisco Security Device Manager
ADF	Augmented Dickey Fuller

Chapter 1: Introduction

1.1 Introduction

Networks are everywhere. Now a days, individuals can hardly do anything with data that does not involve a network. Like the human networks that we are all part of, computer networks let us share resources. A computer network consists of two or more computing devices that are interconnected with each other in order to share resources such as printers, servers and to exchange information stored.

Frequently, networks are classified based on the geographical boundaries the network covers. The most common geographical designation for networks are: Local Area Network (LAN) and Wide Area Network (WAN). A third designation, Metropolitan Area Network (MAN), is also used. As the naming convention shows, LANs are for smaller, more localized networking – in a home, business, school, etc. – while WANs cover larger areas, such as cities, and even allow computers in different nations to connect. LANs are typically faster and more secure than WANs, but WANs enable more widespread connectivity. LANs tend to be owned, controlled and managed in-house by the organization where they are deployed. Whereas, WANs typically require two or more of their constituent LANs to be connected over the public Internet or via a private connection established by a third-party telecommunication provider.

One important area in the context of networking focuses on developing traffic models which can be applied to the Internet. Development of robust networks heavily depends on the modeling of the networks. A good traffic model lead to a better understanding of the characteristics of the network traffic itself. Which, in turn can help designing routers and devices which handle network traffic. Also for network simulations, traffic models are needed as input. Thus, it is essential that the assumed models reflect as much as possible the relevant characteristics of the traffic it is supposed to represent. Inaccurate modeling of network can lead to problems in quality of service, performance, loss of money, …

One of the most widely used traffic model is the Poisson model, which is based on the Poisson process. This model is developed in the context of telephony communication by A. K. Erlang [1]. It has been a favorite traffic model for data and voice. The Poisson process has several properties that make it appealing for analysis. In the context of Internet data traffic, for large populations where each user is independently contributing a small portion of the overall traffic, user sessions can be assumed to follow a Poisson arrival process. But regardless of the analytical benefits of Poisson model, researches states that it is not adequate to model the network traffic due to the burstiness nature of Internet traffic.

To represent burstiness through a Poisson model, the compound Poisson process come out in use. In compound Poisson model, the base Poisson model is extended to deliver batches of packets at once. And it shares some of the analytical benefits of the pure Poisson model. Despite the

attractiveness of the Poisson model, its validity for real traffic has been often questioned. Then, after further researches have been done, another process emerges that characterize the burstiness nature of the data traffic, that is the self-similarity process. Self-similarity is a phenomenon where a certain property of an object is preserved with respect to scaling in space and/or time. If an object is self-similar, its parts, when magnified, resemble the shape of the whole [2].

Network traffic with self-similar process can be modeled using one of self-similar models such as autoregressive moving average(ARMA), autoregressive integrated moving average(ARIMA), fractional autoregressive integrated moving average(FARIMA), … based on the traffic being long-range dependent(LRD) or short-range dependent(SRD).

This paper focuses on Wide Area Network and uses statistical techniques to arrive at the characteristics and traffic model of WAN. Thus, on this work first the characteristics of traffic will be studied. Then depending upon the characteristics of the traffic on the network, a traffic model can be chosen for modeling the traffic.

1.2 Problem Statement

In the field of networking, understanding the characteristics of traffic and modeling the traffic is one of the major areas to focus. Different researchers who have done researches on this topic infer the characteristics of network traffic and suggest different traffic models based on the study they conduct.

Among the different mentioned characteristics of network traffic, the Poisson process is one of the most widely used. However, because of the burstiness nature network traffic researches state that it is not good enough to represent Internet traffic. Then again further studies have been conducted and come out with other models like compound Poisson model, Markov model, … to represent burstiness. But still researches put that due to the burstiness nature of network traffic, the Poisson process is not adequate to characterize Internet traffic. Instead the self-similar process more characterizes network traffic.

Therefore here, this research is conducted to answer the question which behavior more characterize Wide Area Network egress traffic. Then depending upon the characteristics of the traffic on the network the research also give answer for the question which model best capture Wide Area Network egress traffic data.

1.3 Objective

1.3.1 General Objective

The general objectives of conducting this research is:

- To analyze the network traffic characteristics in WAN

1.3.2 Specific Objective

The specific objectives of this research are:

- To study traffic modeling in data networks
- To collect traffic data and characterize its behavior
- To fit the traffic data characteristics to different traffic models.

1.4 Scope of the research

Many researches have been done to analyze the characteristics and model internet traffic, both for Local area network and Wide area network. But, this research focuses only on the Wide Aare Network egress traffic.

1.5 Significance of the research

Conducting this research have significance for the field of networking. The research contributes to understand the characteristics of Wide Area Network and to which traffic models it best fits. Understanding what process Wide Area Network exhibits and knowing which traffic models captures Wide Area Network traffic is a significant input for network equipment designers and also for network simulation.

1.6 Organization of the paper

The thesis is divided in to 8 chapters. This first chapter, that is Introduction outlines the work of the paper. Chapter 2 discusses researches done before on this area of topic. Chapter 3 gives an introduction to Networks. It also discusses the classification of networks: LAN and WAN. The next chapter, Chapter 4 present about Self-similar process and Long-Range dependence in detail. Chapter 5 discusses the different network traffic models. For clarity, the chapter describes the traffic models by classifying them in to two: Non-Self-Similar traffic models and Self-Similar traffic models. Chapter 6 presents the experimental setup used on this research and the methods followed. Chapter 7 describes results obtained on conducting this research. Then the last chapter, Chapter 8 concludes the thesis with recommendation for further work.

Chapter 2: Literature Review

In this section, an overview of previous work done on this area of topic will be discussed. So far different researches have been done on exploring the characteristics of network traffic and finding model of network traffic since developing a detailed understanding of traffic characteristics of network helps to design robust and reliable networks and network services.

A group of researchers from Bellcore and Boston University delivered a paper entitled "On the Self-Similar nature of Ethernet traffic" [3]. An extended version of this paper was also published in the IEEE/ACM Transaction on Networking [2]. In these papers, the authors collect a massive amount data of Ethernet traffic measurements from various Ethernet LAN at Bellcore. And after detailed analysis they have demonstrated that the (aggregate) Ethernet traffic is self-similar. On their analysis of the Ethernet data, as the number of traffic sources increases, the aggregate traffic becomes smoother (less bursty) for the Poisson model, which has very little to do with reality. Thus, they suggest that the Poisson model is inadequate as a model for network traffic [3].

Likewise, from Addis Ababa University, a thesis done by Yalemzewd Negash state that ethernet LAN traffic is statistically self-similar after using different mathematical and graphical techniques. And also, the result shows the long-range dependence or the presence of long memory in Ethernet data traffic. Additionally, a FARIMA traffic model is developed [4].

Two researchers from Lawrence Berkeley Laboratory and from California University published a paper entitled "Wide-Area Traffic: The failure of Poisson Modeling" [5]. In this paper, the authors did analysis on WAN traffic (TCP traffic). They collect large amount of TCP packets from different sites. Then after very rigorous statistical analysis they state that Poisson underestimate the burstiness of TCP traffic over a wide range of time scales and self-similarity is present in aggregate WAN traffic. Also, they show that Poisson processes are valid for modeling user sessions, TELNET connection.

Similarly, studies done on World Wide Web traffic have showed that WWW traffic possess self-similar nature. Researchers from Boston University published a paper that express traffic patterns generated by browsers have a self-similar nature and gave some possible causes. Also states that web browser is modeled as an ON-OFF source model and data fits well the Pareto distribution [6].

There are also published papers that have shown the self-similar nature of the digitized Variable Bit Rate (VBR) video traffic as transmitted over ATM and Internet. The results of the paper reported that video transmission exhibits self-similarity and long-range dependence seems to be an inherent feature of VBR video traffic [7].

The issue of self-similarity has also been addressed in various studies from many aspects including its impact on network performance, modeling techniques, and cause of the appearance of self-similarity [8].

After knowing the characteristics of the traffic on the network, a traffic model can be chosen for modeling the traffic, which is difficult to find out easily. The difficulty in finding an accurate yet parsimonious model is due to the heterogeneous and complex nature of traffic running through the network. Different researchers on their paper suggest different models for network traffic.

A group of researchers from Politecnio di Torino, Italy proposed a new MMPP (Markov Modulated Poisson Process) traffic model on their paper. They report that this model accurately approximates the LRD characteristics of Internet traffic traces. On their result, they notice that the queuing behavior of the traffic generated by the MMPP model is coherent with the one produced by real traces collected at edge router of their institution under different traffic loads [9].

Conversely, A.Dainotti et.al researchers propose another traffic model, that is the Hidden Markov Model(HMM) for Internet traffic sources at packet level. They have done analysis on four kinds of traffic sources of different Internet applications. On their result, they show that the proposed HMM approach is able to capture the behavior of marginal distributions, mutual dependencies and the temporal structure of the traffic generated by a heterogenous set of sources [10].

Another paper on Internet traffic modeling presents the Poisson Pareto Burst Process(PPBP) as a simple and accurate model for Internet traffic. On this paper, the formulae relating the parameters of the PPBP to measurable traffic statistics is discussed. Also, the technique for fitting the model to a given traffic stream is described. As a result, the paper authors found that the PPBP model to be very promising Internet traffic model [11].

As discussed above, even most studies agree that Internet traffic exhibits self-similar nature, the papers don't suggest traffic model that best fit the network traffic. Besides, the models mentioned on the researches done on traffic modeling does not support the self-similar nature of network traffic since they present non-self-similar traffic models. So here, this work will concentrate on WAN and answers the questions which behavior more characterize WAN egress traffic and which model best capture the WAN egress traffic data.

Chapter 3: Introduction to Networks

3.1 What is a Network?

The dictionary defines the word network as "a group or system of interconnected people or things". Similarly, in the computer world, the term network means a set of computers connected together in order to share resources such as file server and office machines. The connection between the computers can be done via cabling through the Ethernet cable or wirelessly through radio waves. The best example of a computer network is the Internet, which connects millions of people all over the world.

The first computer network designed was the 'Advanced Research Projects Agency Network (ARPANET)' by the United States Department of Defense. Since then, myriads of new computer networking technologies have been designed. Figure 1 shows an example of a computer network made up of computers, printer and a network device called hub; they share resources such as files and the printer through the network device.

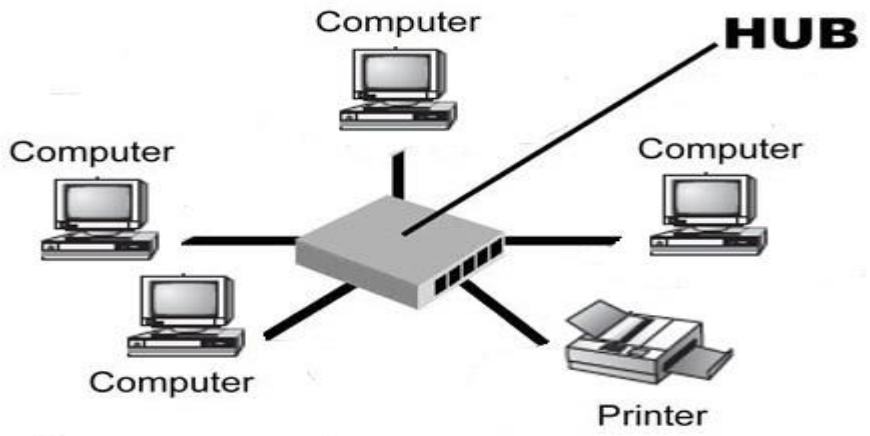

Figure 1. Computer Network

Computer Network have its own advantages and disadvantages. Some are listed below.

Advantage of Network

- Files can be stored on a central computer (the file server) allowing data to be shared throughout an organization
- Expensive devices such as printers or scanners can be shared, that is it saves money
- Files can be backed up more easily when they are all on a central fileserver rather that when they are scattered across a number of independent workstations.

- Networks also allow security to be established, ensuring that the network users only have access to certain files and applications

Disadvantages of Network

- If the file server breaks down the files on the file server become inaccessible.
- Viruses can spread to other computers throughout a computer network.
- There is a danger of hacking, particularly with wide area networks. Security procedures are needed to prevent such abuse, E.g. a firewall

Network traffic, also known as data traffic, refers to the amount of data moving across a computer network at a given point of time. During high traffic periods a computer network become congested if the network is not adequate for the load. A network traffic can be an Egress traffic or Ingress traffic. The dictionary defines the word Egress as "the action of going out of or leaving a place" and Ingress as "the action or fact of going in or entering". An egress traffic is network traffic that begins inside of a network and passes through the edge router of the network to reach the destination node. Whereas, ingress traffic is network traffic that originates from external networks and passes to the destination inside of the network. The term egress is applied to the information from Intranet (a local or restricted communications network) to Internet and the term ingress implies the information from Internet to the Intranet.

3.2 Components of the Network

For a network to provide services and resources, different components are required. These various components work together to make sure that resources are delivered to the correct destination. The network infrastructure contains three categories of network components: End devices, Intermediary devices and Network media.

End devices

End devices, also called hosts are network devices that people are most familiar with. These devices form the interface between users and the underlying communication network. Computers, servers (file servers, web servers) and mobile handheld devices are example of end devices.

Hosts on a network can have different roles. The two most common roles are clients and servers. The role the device plays on the network is determined by the Software installed on it. Clients are hosts that request and consume information obtained from the server. Whereas servers are hosts that provide information and services, like web pages to other hosts on the network. A device can have a client role, a server role or may also be capable of functioning both the client and server role.

In order to distinguish one host from another, each host is identified by a unique address on the network, often an IP address or a Media Access Control(MAC) address. When a host starts communication, it uses the address of the destination host to tell where the message should be sent.

Intermediary devices

Intermediary devices are special purpose network devices used to interconnect end devices. These devices enable the network to run by providing connectivity. They determine the path the message should take through the network to reach at the destination. Some of the intermediary network devices are discussed below.

Network hub: - is a device that broadcasts data to every computer connected to it. A network hub has no intelligence on where to send a message, thus when it receives a packet of data from a connected device, it broadcasts across each port to all connected devices. Also, network bandwidth breaks up to all of the connected computers. Hence, the more computer that are connected, the less bandwidth that is available for each computer, which implies slower connection speeds.

Network switch: - is a device more advanced than hub. A network switch connects computers to each other as a hub, but it does not broadcast the packet of data to all computers connected to it, rather it determines which computer or device the packet is intended for and forwards it only to that computer. Besides, bandwidth is not shared among the connected devices thus makes the network much more efficient.

Network routers: - is a device with a lot more capabilities than a hub or switch. Its function is to route packets of data over a different network, instead of local networks. And also, it determines the best path for a data packet to be forwarded to its destination. In a large network different types of routers can be used, such as core router and edge router. A core router is router in a computer network that routes data packets within a network, but not between networks. An edge router is a router that serve as an entry point to the core of the network and it routes data packets between a self-contained network and other outside networks.

Firewall: - is a hardware device or software that prevents a data to enter or leave a network or computer, just like a security guard that allows to enter/exit a building. It is a security device that helps by blocking unauthorized access to networks.

Network media

Network media is the actual path over which an electrical signal go through as the information moves from one component to another. Data transmission across a network is carried on a medium. The common types of network media include Metallic wires within cables, Glass or plastic fibers (fiber-optic cable) and Wireless transmission.

For each media type, the signal encoding used for the message to be transmitted is different. In case of metallic wires, the information is encoded into electrical impulses. For fiber-optic cables transmission is carried on using pulses of light. Wireless communication uses radio frequency. Not all network media types are applied for the same purpose. Each of the different network media types have their own pros and cons.

There are various points to see for choosing which type of network media to use for specific purpose. Some of them are the distance the media can successfully carry a signal, the environment in which the media is to be installed, the amount of data and the speed at which it must be transmitted and the cost of the media and installation.

3.3 Classification of Network

Computer networks can be classified on the basis of different features. It can be classified based on the type of connection technology used to connect the devices in the network: Ethernet cable, optical fiber and wireless connection. Computer networks may also be classified based on the functional relationships which exists between the nodes, that is if the communicating node is able to accept service, offer service or both accept and offer service: client-server architecture, peer-to-peer architecture.

The network topology upon which the network is based can also classify computer networks. Network topology represent the physical and logical structure of a network. The common topologies include: Bus Network, Star Network, Ring Network, Mesh Network, Star-Bus Network, Tree or Hierarchical Topology Network. Also, computer networks can be classified according to the geographical distance the network spans as Local Area Network(LAN), Metropolitan Area Network (MAN) and Wide Area Network(WAN). LAN is a network infrastructure that covers comparatively small geographical area whereas WAN enables connection among many devices for large geographical area. A MAN is a network infrastructure that spans larger geographical area than LAN but smaller geographical area than WAN. On the next section LAN and WAN will be discussed in detail.

3.3.1 Local Area Network LAN

As the name implies, LAN is a computer network that spans a particular geographical location such as an office building, a single department within a corporate office, or even a home office. Back in a day, there was strict limitations on number of computers to connect to LAN and on the distance the machines could actually be from each other. Now, because of technological advances that’s changed and there is no restriction regard to both LAN size and distance a LAN can span.

Typically, a local area network is a private network owned and administrated by a single organization. To make administration easier it is good practice to split a big LAN in to smaller logical zones. For example, in a typical business environment there could be different LAN

connection for different departments. LANs are considered as the building blocks for creating larger networks. Figure 2 shows two separate LANs.

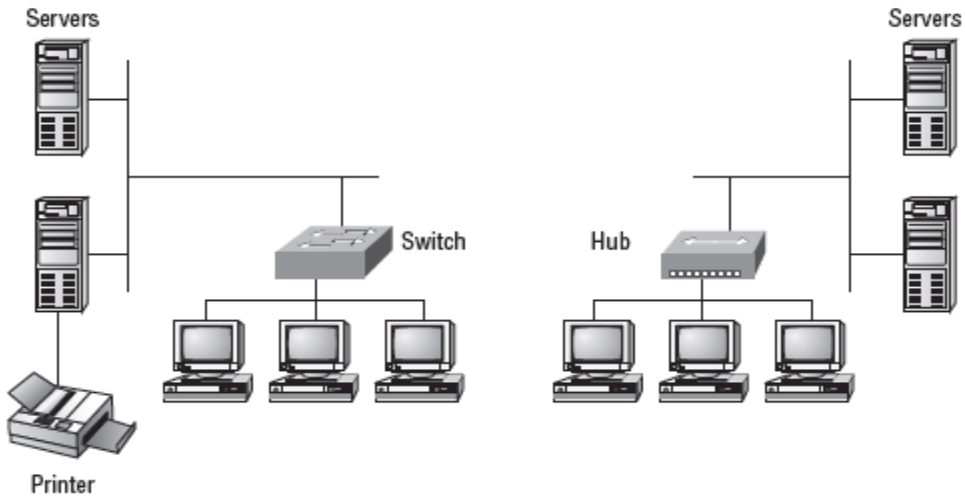

Figure 2. Two separate LANs

The network devices hub and switch allow hosts to physically connect to resources on a LAN. As the figure shows, any device that connects to the LAN can access the resources of that LAN. To get the resources from it, one must be physically connected, via cable or wireless, to the LAN. But it is not possible to get resources of one LAN from the other LAN unless the LANs are connected to each other. One way to make a connection between LANs is to physically connect the two LANs with a cable that would allow the two groups to hook up. But, this way creates only one big group instead of separate groups, which in turn result slower speed since all connected hosts are trying to access the same resource.

The other and better way to make connection between LANs is using a network device called router. When using a router, there will be smaller yet connected groups that are able to share resources. And will result the hosts on each LAN to have much faster response time when accessing resource. Figure 3 shows a router connecting two LANs.

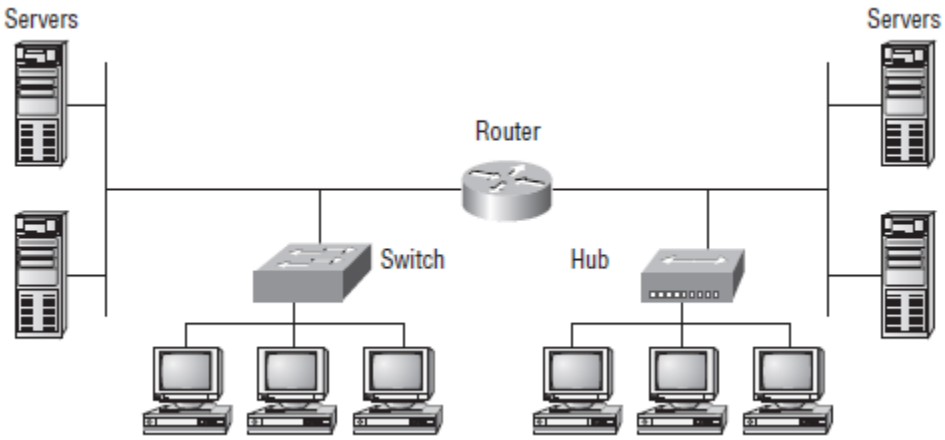

Figure 3. A router connects two LANs

3.3.2 Wide Area Network WAN

As local area networks grew and developed, it became necessary to be able to connect their resources together over long distances—not just locally. At first, this goal is achieved via the phone company network known as the Public Switched Telephone Network (PSTN). And so, the first successful network that could establish voice communications over disparate locations was born. PSTN was a fully operational circuit-switching network, where every phone call established a unique circuit from one endpoint (phone) to another through a path of switches. Then packet switching network come is use as communication delivery method using the existing phone company network. Thereafter, making use of the technological advances it become possible to interconnect and share resources over a long distance i.e. WAN.

WAN is computer network that covers larger geographical area such as cities, states or countries. It is made up of two or more LANs or Metropolitan Area Networks (MANs) that are interconnected with each other, thus users and computers in one location can communicate with users and computers in other locations.

Mostly WANs are private and built for a particular organization. For an organization that have offices at different countries, it is almost impossible to pull network cables between their offices. Hence, WAN provides a solution to companies operating at distant geographical locations who want to communicate and share their central data. And Internet Service Providers(ISP) provide the connectivity among the LANs of the organization. Frequently, WANs are built using leased lines where each end of the line is connected to a router to extend the network capability. Figure 4 shows that different LANs at various location connected via Internet to form a wide area network.

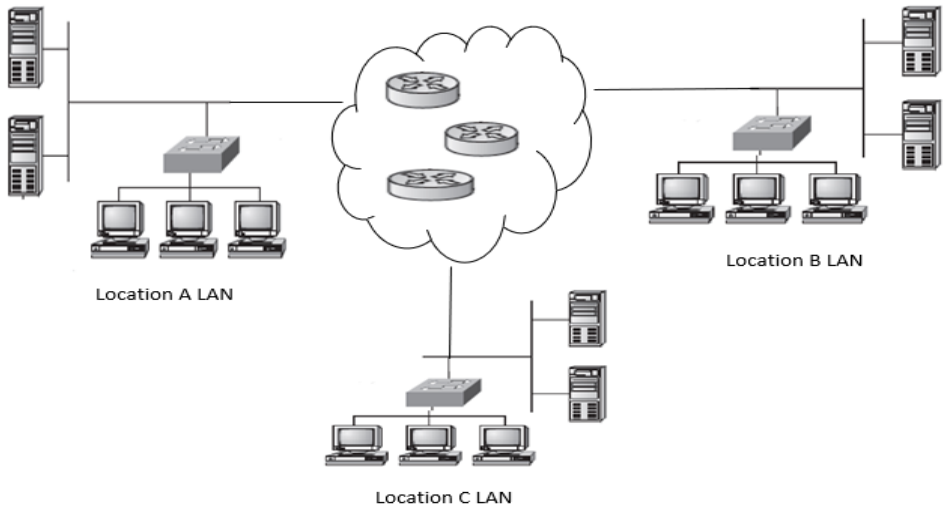

Figure 4. Wide Area Network WAN connection

Wide area networks can be divided in to two: Distributed WAN and Centralized WAN. A distributed WAN is an internetwork that's made up of many interconnected computers located at different places, means which have no central point of control. The primary example of distributed WAN is the Internet which is the largest WAN in the world that helps to connect anyone from any area of the world. The other WAN type, i.e. centralized WAN, is an internetwork that is based on central server or centralized site where all other remote devices are connected to. An example of such WAN is remote offices connected to main corporate office, as shown in figure 5.

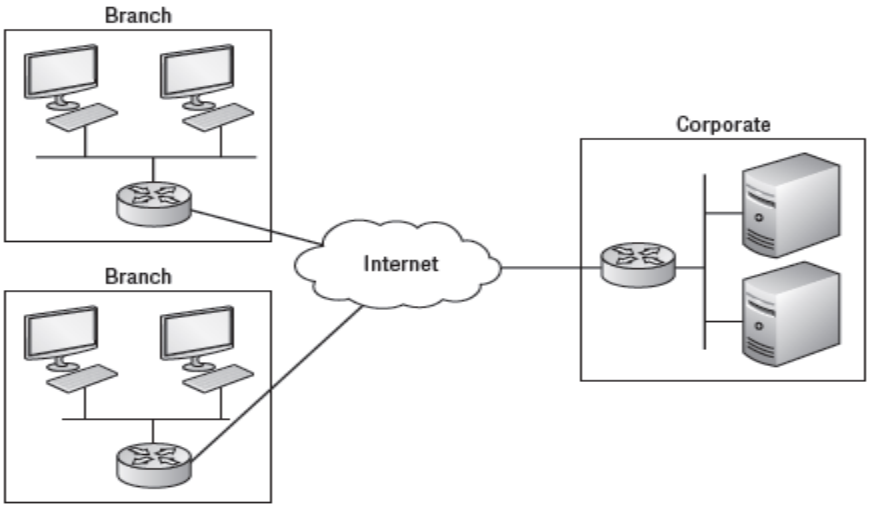

Figure 5. Centralized WAN

When discussing about WAN, there are terms to know that service providers use mostly [12].

Customer premises equipment (CPE**)**: - is equipment that's owned by the service provider but located on the subscriber's premises.

Demarcation point: - is the spot where the responsibility for the connection is exchanged between the customer and the service provider. Physically, it is cabling junction box located on the customer premises, that connects the CPE wiring to the local loop. When problem arise, it is important for troubleshooting because it helps to determine whether the problem is on the customer side or on the service provider side.

Local loop: - is a cable that connects the demark to the closest switching office, known as a central office (CO).

Central office (CO): - also know as a Point of Presence is a service provider building that connects the customer's network to the provider's switching network. It is the entry point to the WAN cloud and the exit point from the WAN.

Toll network: - is a collection of trunks inside a WAN provider's network. This network is a collection of switches and facilities owned by the ISP.

Wide area network differs from local area network in various aspects such as coverage, congestion, bandwidth… Bandwidth is the amount of data that can be transmitted from one point to another in a given period of time. It can be referred as the size of the pipe in which the data can pass through. If the pipe is not enough for all the data to travel through, there becomes congestion. Network bandwidth is often expressed in bits per second(bps). In Wide area network low bandwidth is available for data transmission. Network congestion refers a network state where a network node is carrying more data and becomes too full. It can be think of as a highway traffic. Wide area network is more congested than local area network. The comparison between the two computer networks is presented on Table 1.

Table 1. LAN vs WAN

	LAN	**WAN**
Coverage	Covers small geographical areas (e.g. home, office)	Spans large geographical areas (e.g. cities, states, nations)
Speed	High speed	Less speed
Congestion	Less congestion	More congestion
Bandwidth	Higher bandwidth rate	Lower bandwidth rate
Transmission Errors	Few	More
Set-up costs	Low	High
Maintenance cost	Less	More
Reliability	More reliable and secure than WAN	Less reliable and secure

3.3.2.1 WAN connection types

There are three main types of WAN connections that are used to connect LANs together. These are: Dedicated (leased) lines, circuit switched, and packet switched [12]. The switching function provides communication pathways between two endpoints and manages how data flows between them.

Dedicated (leased) lines: - These are usually referred to as a point-to-point connection. A dedicated leased line is a pre-established WAN communication path through the service providers network to a remote network. It is used in case of specific bandwidth requested thus, provide a reserved connection for the client and due to this it is costly. Leased line connections use synchronous serial lines with speed up to 45 Mbps.

Circuit switching: - Circuit switching requires a dedicated physical connection between the sending and receiving devices. The sending device establishes a physical connection, and the data is transmitted between the two. No other user can use this circuit till this session is completed. When the transmission is complete, the channel is closed, and it is paid only for the time that's actually used. It is just like a phone call. Parties involved in a phone call have a dedicated link between them for the duration of the conversation. When either party disconnects, the circuit is broken, and the data path is lost. Before an end-to-end connection is established, data cannot be transferred. An example illustrating circuit switching during a phone call is shown in figure 6.

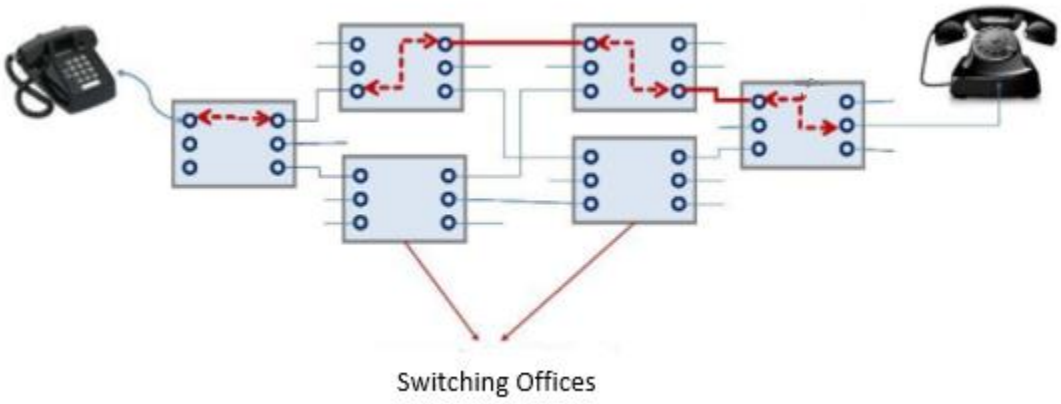

Figure 6. Circuit switching

Packet switching: - Packet switching is the process of transmitting data in small units called packets. In packet switching, messages are broken into smaller pieces: packets. A packet is the unit of data that is routed between an origin and destination on the Internet. Each packet carries the information that will help it get to its destination -- source and destination addresses. Packets are required to have this information because they do not always use the same path or route to get to their intended destination, they are sent off by the best available route. Packets can take an alternative route if a particular route is unavailable for some reason. This makes the network more efficient. When reached at the destination, packets are reassembled in the proper sequence to make up the message. If the amount of data to transmit is large, it's better to break it down into packets. Figure 7 illustrates basic packet switching between a sender and receiver.

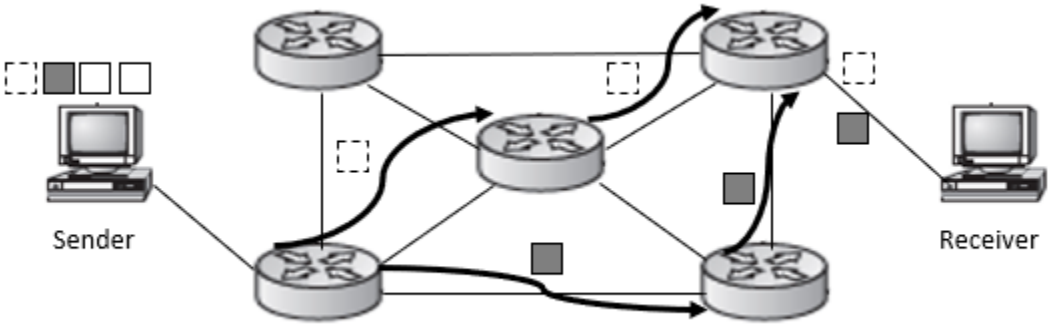

Figure 7. Packet Switching

Chapter 4: Self-Similar Process and Long-Range Dependence

4.1 Introduction

Computer networks possess complex characteristics due to the heterogenous nature of the traffic. So that the design of robust and reliable networks and network services is difficult. The path to achieve this goal is to develop a detailed understanding of the traffic characteristics of the network. Depending upon the type of network and the characteristics of the traffic on the network, a traffic model can be chosen for modeling the traffic. Various stochastic processes such as fractional Gaussian noise, Poisson process have been proposed to model network traffic.

In telephone traffic, the basic characteristic is its limited variability in both time and space. In fact, telephone traffic processes are either independent or have temporal correlations that decay exponentially fast; moreover, the distributions of traffic related quantities have exponentially decaying tails. Low variability implies no or limited burstiness. However, data networks are characterized by high or extreme variability [5] [3]. Large variability in space and time generally causes the corresponding traffic to manifest fractal behavior, that is, to show some statistical properties which repeat themselves at many time scales. Fractal process are stochastic processes that exhibit some kind of fractal like properties such as self-similarity.

One of the objections to self-similar traffic processes was the difficulty in mathematical analysis. Since the seminal study of Leland, et.al [3] which set the ground work for considering self-similarity an important notion in the understanding of network traffic including the modeling and analysis of network performance, an explosion of work has ensued investigating the multifaceted nature of this phenomena. Some studies of real network traffic data have shown that network traffic exhibits self-similar or fractal properties over a wide range of time series but there are also researches that shows network traffic possess a process other than self-similar [10] [11].

The properties of self-similar network traffic are very different from the properties of traditional models based on Poisson, Markov-modulated Poisson and related processes. The use of traditional models in networks characterized by self-similar processes can lead to incorrect conclusions about the performance of the analyzed networks. Traditional models can lead to over-estimation of the network performance, insufficient allocation of communication and data processing resources, and consequently difficulties in ensuring the quality of service [13].

Also, the traditional models such as the Poisson model states that network traffic is memory-less that is the traffic behavior across widely separated times is not correlated. But in real system, data network traffic exhibits long-term memory or Long-Range Dependent(LRD) which implies that similar pattern of traffic persists for a longer span. In this section, self-similarity and long range dependent process is presented in detail.

4.2 What is Self-Similarity?

Self-Similarity and fractals are notions pioneered by Benoit B.Mandelbrot. They describe the phenomenon where a certain property of an object - e.g. a natural image, a time series – is preserved with respect to scaling in space and/or time. “If an object is self-similar or fractal, its parts, when magnified resemble – in a suitable sense - the shape of the whole” [14]. It is something that feels the same regardless of scale. Figure 8 and figure 9 provides a pictorial view of this phenomenon.

Figure 8. Self-Similar in nature

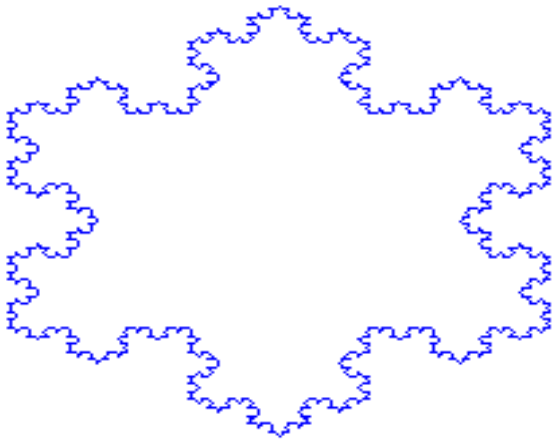

Figure 9. The Koch snowflake fractal

Self-similarity is measured by the Hurst-parameter – H which is developed by Harold Hurst. H represents the burstiness of traffic also considered as a measure of self-similarity and is a value between 0 and 1. As the quantity of data increases the Hurst parameter H also increases, that means traffic becomes more self-similar.

Understanding whether network traffic possess self-similar characteristics or not is important not to model network traffic inaccurately. Traffic models like Poisson do not take into account the self-similar nature of traffic. A Poisson process, when observed on a fine time scales will appear bursty and when aggregated on a coarse time scale will flatten(smooth). However, a Self-similar process when aggregated over wide range of time scales will maintain its bursty characteristics.

Self-similarity can be classified into two types: deterministic and stochastic [13]. In the first type, deterministic self-similarity, a mathematical object is assumed to be self-similar or fractal if it can be decomposed into smaller copies of itself. That is, self-similarity is a property in which the structure of the whole is contained in its parts. In case of the second type, Stochastic self-similarity, probabilistic properties of self-similar processes remain unchanged or invariant when the process is viewed at different time scales. This contrasts with Poisson processes that lose their burstiness and flatten out when time scales are changed. However, the time series of self-similar processes exhibit burstiness over a wide range of time spans. Burstiness implies a quantity reflecting its variability. Burstiness can also be defines as the ratio of peak rate to mean rate.

From the two types of self-similarity, this work focuses on the stochastic self-similarity. So, the next section discusses about Stochastic self-similarity in detail.

4.3 Stochastic self-similar

Stochastic Self-Similarity is a property that can be illustrated visually. Figure 10 shows a traffic trace, where number of packets against time is plotted with different time granularity. That is, a single data point is the aggregated traffic volume over the time interval. The figure depicts the same traffic series by zooming the initial segment further, rescaling it by a factor of 10 i.e. 1 sec, 100msec and 10 msec.

Unlike deterministic self-similarity, the plots corresponding to the below figure do not possess resemblance of their parts with the whole at finer details. Here it is the general impression that remains the same. Indeed, for measured traffic traces, it would be too much to expect to observe exact, deterministic self-similarity given the stochastic nature of many network events that collectively influence actual network traffic. However, if the measure of resemblance is relaxed, like focusing on certain statistics of the rescaled time series, then perhaps it would be possible to expect similarity of the mathematical objects with respect to the relaxed measures. Statistical properties that capture burstiness or variability and the autocorrelation function are measures which scale-invariance can be defined.

A geometric shape is called self-similar in a deterministic way if the same geometric structures are observed. In the context of stochastic processes, self-similarity is defined in terms of the distribution of the process.

Definition: Let Y_t be a stochastic process with continuous time parameter t. Y_t is called self-similar with self-similarity parameter H, if for any positive stretching factor c, the rescaled process with time scale ct, $c^{-H}Y_{ct}$, is equal in distribution to the original process Y_t.

This means that, for any sequence of time points $t_1, \dots, t_k$, and any positive constant c, $c^{-H}(Y_{ct1}, Y_{ct2}, \dots, Y_{ctk})$ has the same distribution as $(Y_{t1}, Y_{t2}, \dots, Y_{tk})$ [15]. There are different classes of self-similarity. The exact and asymptotic self-similarity will be discussed below [2].

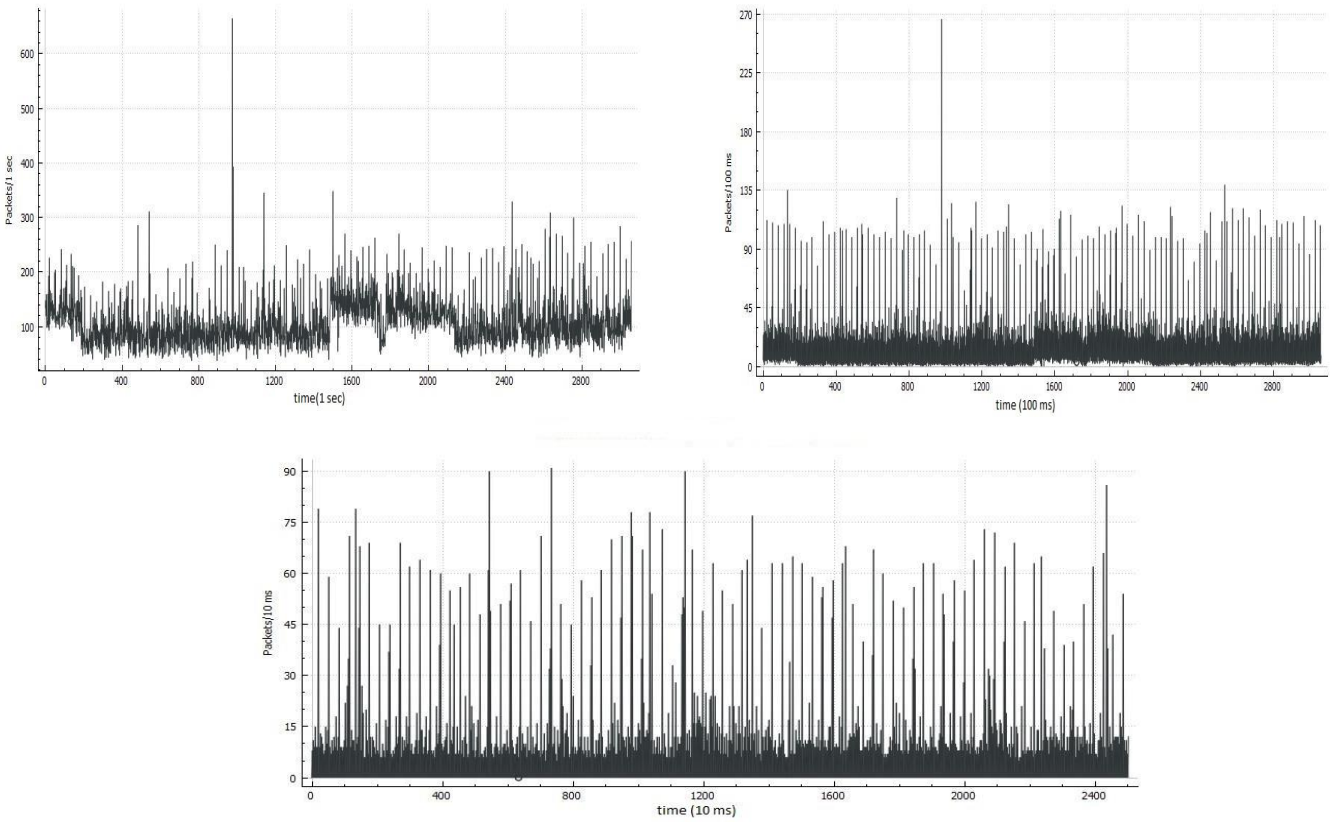

Figure 10. Stochastic self-similarity, burstiness preservation

Let $X = (X_t : t = 0,1,2, \dots)$ be a covariance stationary (sometimes called wide-sense stationary) stochastic process; that is, a process with constant mean $\mu = E[X_t]$, finite variance $\sigma^2 = E[(X_t - \mu)^2]$, and an autocorrelation function $r(k) = E[(X_t - \mu)(X_{t+k} - \mu)]/E[(X_t - \mu)^2]$ $(k = 0,1,2, \dots)$ that depends only on k. Let's assume that X has an autocorrelation function of the form

$$r(k) \sim k^{-\beta} L_1(k), as\ k \to \infty \qquad (1)$$

where $0 < \beta < 1$, k is the lag in time and L_1 is slowly varying at infinity, that is, $\lim_{t\to\infty} L_1(tx)/L_1(t) = 1$ for all $x > 0$ (examples of such slowly varying functions are $L_1(t) = const, L_1(t) = \log(t)$). For each m = 1, 2, 3, . . ., let $X^{(m)} = ({X_k}^{(m)} : k = 1,2,3, \dots)$ denote a new time series obtained by averaging the original series X over nonoverlapping blocks of size m. That is, for each m = 1, 2, 3, . . ., $X^{(m)}$ is given by

$${X_k}^{(m)} = 1/m(X_{km-m+1} + \cdots + X_{km}),\ k = 1,2,3, \dots \qquad (2)$$

Note that for each m, the aggregated time series $X^{(m)}$ defines a covariance stationary process; let $r^{(m)}$denote the corresponding autocorrelation function.

Exact self-similar [2]: - The Process X is called exactly self-similar with self-similarity parameter $H = 1 - \beta/2$, if, for all $m = 1,2,3, \ldots, 1/m^H(X_{km-m+1} + \cdots + X_{km}),\ k = 1,2,3, \ldots$ has the same finite dimensional distributions as X. It is (exactly second order) self-similar with self-similarity parameter $H = 1 - \beta/2$ if for all $m = 1,2,3, \ldots, 1/m^H(X_{km-m+1} + \cdots + X_{km})$ has the same variance and autocorrelation as X. in terms of the aggregated processes $X^{(m)}$, this means that for all m = 1,2,3,…, $var(X^{(m)}) = \sigma^2 m^{-\beta}$and

$$r^{(m)}(k) = r(k) = \frac{1}{2}\delta^2\left(|k|^{2-\beta}\right), k = 0,1,2, \ldots \quad (3)$$

where $\delta^2(f)$ denotes the second central difference operator applied to a function *f*, that is, $\delta^2\big(f(k)\big) = f(k+1) - 2f(k) + f(k-1)$. An example of an exactly self-similar process with self-similarity parameter H is fractional Gaussian noise (FGN) with ½ < H < 1, that is, the increment process of fractional Brownian motion with parameter H.

Asymptotic self-similar [2]: - The Process X is called (asymptotically second order) self-similar with self-similarity parameter $H = 1 - \beta/2$ if

$$r^{(m)}(k) \to \frac{1}{2}\delta^2\left(k^{2-\beta}\right) as\ m\ \to \infty, k = 0,1,2, \ldots \quad (4)$$

Thus, an asymptotically self-similar process has the property that, for large m, the corresponding aggregated time series $X^{(m)}$ have a fixed correlation structure, solely determined by β; moreover, due to the asymptotic equivalence (for large k) of differencing and differentiating, $r^{(m)}$ agrees asymptotically with the correlation structure of X given by (1) [16].

Intuitively, the most striking feature of exactly or asymptotically self-similar processes is that their aggregated processes $X^{(m)}$ possess a nondegenerate correlation structure as m →∞. This behavior is in stark contrast to the more conventional stochastic models, all of which have the property that their aggregated processes $X^{(m)}$ tend to second-order pure noise as m → ∞ that is [16]:

$$r^{(m)}(k) \to 0, as\ m \to \infty, k = 1,2,3, \ldots \quad (5)$$

Stochastic self-similar process has a property of slowly decaying variances. From a statistical point of view, the most salient feature of self-similar processes is that the variance of the arithmetic mean decreases more slowly than the reciprocal of the sample size; that is, it behaves like $n^{-\beta}$ for β ∈ (0, 1), instead of like n^{-1}.

$$var(X^{(m)}) \sim am^{-\beta}, as\ m \to \infty \quad (6)$$

where *a* is a finite positive constant independent of m, and $0 < \beta < 1$

4.4 Statistical Methods for testing Self-similarity

Self-similarity is measured by the Hurst parameter - H. H represents the burstiness or self-similarity of traffic and is a value between 0 and 1. Hence, to test for self-similarity of network traffic, the value of estimated Hurst parameter H is used. If the obtained estimated Hurst parameter approaches 1, that is if H ranges between 0.5 and 1, then this implies that the series exhibits self-similar process.

The problem of testing for and estimating the degree of self-similarity can be approached from two different angles: (1) analysis of the variances of the aggregated processes $X^{(m)}$, and (2) time-domain analysis based on the R/S-statistic. This subsection provides description of the corresponding statistical methods

4.4.1 Variance-time plots

As discussed in stochastic self-similar processes section, for self-similar processes the variances of the aggregated processes $X^{(m)}$, m = 1,2, 3,…, decrease linearly for large m in log-log plots against m with slopes arbitrarily flatter than -1. The so-called variance-time plots are obtained by plotting log(var($X^{(m)}$)) against log(m) and by fitting a simple least squares line through the resulting points in the plane, ignoring the small values for m. If the value of the estimate asymptotic slope β is between -1 and 0, then this suggest self-similarity. And an estimate for the degree of self-similarity is given by the equation H = 1 + β/2. Variance-time plots are not reliable for empirical records with small sample sizes.

Given a sample of *N* observations (X_k: $k = 1, 2, 3, \ldots, N$), one subdivides the whole sample into *K* non-overlapping blocks of size m then to conduct the variance-time plot analysis and to estimate the Hurst exponent the steps to follow are:

Step 1: - calculate the sample means $\bar{X}1, \bar{X}2, \ldots, \bar{X}k$

$$\bar{X} = \frac{1}{n}\sum_{i=1}^{n} x_i \qquad (7)$$

Step 2: - calculate the overall mean

$$\bar{X}(m) = \frac{1}{k}\sum_{i=1}^{k} \bar{X}i \qquad (8)$$

Step 3: - calculate sample variance $S^2(m)$

$$S^2(m) = \frac{1}{k-1}\sum_{i=1}^{k} (\bar{X}i - \bar{X}(m))^2 \qquad (9)$$

Step 4: - plot log $S^2(m)$ against log m and fit a simple least square line through the resulting points then H = 1 + ½(slope)

4.4.2 R/S analysis

R/s analysis (or Rescaled Range analysis) was originally devised by Harold Edwin Hurst in its studies of the Nile discharge. Hurst was an English hydrologist, who worked on the Nile River Dam project. When designing a dam, the yearly changes in water level are of particular concern in order to adapt the dam's storage capacity according to the natural environment. Studying an Egyptian past year record of the Nile River's overflows, Hurst observed that flood occurrences could be characterized as persistent, i.e. heavier floods were accompanied by above average flood occurrences, while below average occurrences were followed by minor floods. In the process of this findings he developed the Rescaled Range (R/S) Analysis.

The R/s analysis is a simple method, easily implemented in a program and that provides a direct estimation of the Hurst Exponent which is a precious indicator of the state of randomness of a time-series. It is especially interesting in revealing the existence of long-term dependence, which prevents, when it exists, the time-series to be reasonably modelized by a random walk.

Here, the objective of the R/S analysis of an empirical record is to infer the degree of self-similarity H (Hurst parameter) for the self-similar process that presumably generated the record under consideration. In practice, R/S analysis is based on a heuristic graphical approach that tries to exploit as fully as possible the information in a given record.

Given a sample of N observations ($Xk : k = 1, 2, 3, \ldots, N$), one subdivides the whole sample into K non-overlapping blocks and computes the rescaled adjusted range $R(t_i, n)/S(t_i, n)$ for each of the new starting points $t_1 = 1$, $t_2 = N/K + 1$, $t_3 = 2N/K + 1$, . . . which satisfy $(t_i - 1) + n \leq N$. Here, the R/S-statistic $R(t_i, n)/S(t_i, n)$ is defined as

$$R(n)/S(n) = 1/S(n)[max \sum_{i=1}^{k}(x_i - \bar{x}) - min \sum_{i=1}^{k}(x_i - \bar{x})] \quad (10)$$

and $S^2(t_i, n)$ is the sample variance of $Xt_i + 1, Xt_i + 2, \ldots, Xt_i + n$. Thus, for a given value (lag) of n, one obtains many samples of R/S, as many as K for small n and as few as one when n is close to the total sample size N. Next, one takes logarithmically spaced values of n, starting with $n \approx 10$. Plotting $log(R(t_i, n)/S(t_i, n))$ versus $log(n)$ results in the rescaled adjusted range plot. In the R/S analysis, an estimate of the self-similarity parameter H is given by the line's asymptotic slope (typically obtained by a simple least square fit) which can take any value between 0.5 and 1.

The steps to conduct R/S analysis and to estimate the Hurst exponent are:

Step 1: - Calculate the mean: $\bar{X}1, \bar{X}2, \ldots, \bar{X}k$

$$\bar{X} = \frac{1}{n}\sum_{i=1}^{n} x_i \quad (11)$$

Step 2: - Create a series of deviations from the mean

$$Y_t = X_t - m \text{ for } t = 1,2, \ldots, n \quad (12)$$

Step 3: - Calculate the cumulative deviate series Z

$$Z_t = \sum_{i=1}^{t} Y_i \text{ for } t = 1,2, \ldots, n \quad (13)$$

Step 4: - Calculate the difference in the series of deviations

$$R_t = max(Z_1, Z_{2,}, \ldots, Z_t) - min(Z_1, Z_{2,}, \ldots, Z_t) \text{ for } t = 1,2, \ldots, n \quad (14)$$

Step 5: - Calculate the standard deviation

$$S_t = \sqrt{\frac{1}{t}\sum_{i=1}^{t}(X_i - \bar{X})^2} \quad (15)$$

Step 6: - Calculate the rescaled range series (R/S)

$$(R/S)_t = \frac{R_t}{S_t} \quad (16)$$

Step 7: - Calculate the logarithmic values for the size and plot $\log(R/S)$ versus $\log n$ then fit a simple least square line through the resulting points. The slope of the line estimates H.

4.5 Long Range Dependence LRD

Long range dependence LRD means that the behavior of a time-dependent process shows statistically significant correlation across large time scales. It can be considered as the phenomenon where current observations are significantly correlated to observations farther away in time. Short-range dependence(SRD) on the other hand, refers to the phenomenon where current observations are not correlated to very old observations. For SRD processes, correlation to previous observations decays to zero very quickly while it remains significant for LRD even for very old observations [17].

An indication of long-range dependence is given by the autocorrelation function. Autocorrelation is a correlation coefficient, however, instead of correlation between two different variables, the correlation is between two values of the same variable at time X_t and X_{t+k}. Mathematically it is defined as [18]:

Definition: Given measurements, $X_1, X_2, \ldots, X_N$ at time $t_1, t_2, \ldots, t_N$, the lag k autocorrelation is defined as

$$r_k = \frac{E[(X_t - \mu)(X_{t+k} - \mu)]}{\sigma^2} \quad (17)$$

Where μ is the mean and σ^2 is standard deviation of *X*.

The autocorrelation function of a long-range dependent decays slowly, that is, has a hyperbolic decay, rather than an exponential as in the case of short-range dependent process. That is the autocorrelation function is persistent for large lags. A process that satisfies equation (18) is said to be Long-range dependent(LRD). On the other hand, the auto correlation function of short-range dependent is summable.

$$\sum_{k=0}^{\infty} r(k) \to \infty \quad (18)$$

Chapter 5: Network Traffic Models

5.1 Introduction

A traffic model is a stochastic process which can be used to predict the behavior of a real traffic stream. Ideally, the traffic model should accurately represent all the relevant statistical properties of the original traffic, but such a model may become overly complex. A major application of traffic models is predicting the behavior of the traffic as it passes through a network. Accurate traffic models are necessary for service providers to properly maintain quality of service. Due to the heterogeneous and complex nature of traffic running through the network it is difficult to find an accurate model.

For modeling traffic, a traffic model can be chosen depending upon the type of network and the characteristics of the traffic on the network. If the underlying traffic models do not efficiently capture the characteristics of the actual traffic, it may result the under estimation or over estimation of the performance of the network. In turn, this would have negative effect on the design of the network. Hence, traffic models are a core component of the performance evaluation of networks and they need to be accurate.

Many traffic models have been developed based on traffic measurement data, that is by collecting and analyzing the traffic measurements. This is not an easy task in traffic modeling since traffic can be highly variable, even between two similar types of sources. Therefore, an attention should be given during traffic measurement. It is good to attempt to collect measurements from many sources or over many time periods. A small sample set could give a misleading representation of the correct traffic behavior.

Traffic models can be categorized into two classes: black box models and structural models [19]. Black box models also known as classical time series models typically provide a good fit to observed time series but unfortunately are not intuitive or amenable to understanding of the dynamics governing the time series. Usually such models have certain parameters that can be adjusted to fit the time series. These parameters are then used for analytical work on performance measures of the network such as queuing behavior. Examples of black box models are fractional Brownian motion and fractional autoregressive integrated moving average (FARIMA). In structural modeling, the objective is to reproduce the observed features of the measurements, using models whose parameters are related to the traffic generating mechanism and the behavior of the main components of the network. Example of structural model is the ON/OFF model. This paper focuses on the black box model.

On this chapter, traffic models are discussed in detail. To make it easily understandable, it is presented by categorizing the different traffic models based on the process the models exhibit: - Non-self-similar traffic models and self-similar traffic models.

5.2 Non-self-similar traffic models

Network traffic that don't possess self-similar characteristics are modeled using one of the Non-self-similar traffic models: Poisson model, Markovian model, … Here some of non-self-similar traffic models are discussed.

5.2.1 Poisson model

The Poisson process, which is introduced by A.K. Erlang, is a very popular model with a long history in the field of telecommunications. The success of the Poisson model in modeling voice traffic over circuit switched networks is unquestionable. The Poisson process is characterized as a renewal process. There are variations of the Poisson distributed process that are widely used. There are for example, the Homogenous Poisson process and Non-Homogenous Poisson Process that are used to represent traffic characteristics.

Definition: - The random process $\{N(t), t \varepsilon [0, \infty)\}$ is called a homogenous Poisson process if the following properties are satisfied:

1) $N(0) = 0$, with probability 1.
2) $\{N(t)\}$ is a process with independent and identically distributed (i.i.d) increments.
3) The number of arrivals in any interval of length $\tau > 0$ has Poisson distribution.

There are several interesting mathematical properties exhibited by Poisson processes. Primarily, superposition of independent Poisson processes results in a new Poisson process whose rate is the sum of the rates of the independent Poisson processes. Further, the steady-state equation is reasonably simple to calculate. The Poisson process also have independent increment property. An independent increment process has the very useful property that its probability density/mass function is obtained by multiplying the probability density/mass function of its increments.

Definition: - A random process $\{N(t)\}$ is called an independent increment process if the random variables $x(k) = N(k + 1) – N(k)$, $k = 0, 1, …, k-1$ are independent for any choice of k and $N(0) = 0$ i.e. the counting begins at time zero and the system is empty.

In a Poisson process the inter-arrival times are exponentially distributed with a rate parameter λ. This exponentially distributed inter arrival process renders a Poisson process memoryless. It is memoryless in the sense that, given the previous arrival occurred *T* time ago, the time to the next arrival will be exponentially distributed with mean $1/\lambda$ regardless of *T*. In other words, the waiting time for the next arrival is independent of the time of the previous arrival. This memoryless property simplifies analysis because future arrivals do not need to take into account the past history of the arrival process.

There are several ways to verify whether a particular arrival process is Poisson. An easy visual way is plotting the histogram of the interarrival times and verifying whether it is an exponentially

decreasing function. Other way to test whether a trace of connection arrivals corresponds to non-homogenous Poisson process for a total of T time units is dividing the entire trace in to N=T/I intervals each of length I. Then separately test each interval to see if the arrivals during the interval are consistent with arrivals from Poisson process with rate fixed so that the expected number of arrivals is the same as the number actually observed. The two key characteristics that a Poisson process have are exponentially distributed, and independent interarrival times. Thus, to test for Poisson process, checking these characteristics for each interval is required. One can test the exponential distribution using the Anderson-Darling (A^2) test recommended by Stephens. And to test the interarrivals for independence, one indication is the absence of significant autocorrelation among interarrivals [5].

The two primary assumptions that the Poisson model makes are (1) The number of sources is infinite (2) The traffic arrival pattern is random. The probability distribution function and density function of the model are given as:

$$F(t) = 1 - e^{-\lambda t} \quad (19)$$

$$f(t) = \lambda e^{-\lambda t} \quad (20)$$

The Poisson process is appropriate if the arrivals are from large number of independent sources, i.e. Poisson sources. For large populations where each user is independently contributing a small portion of the overall traffic, user sessions can be assumed to follow a Poisson arrival process. Based on traces of wide-area TCP traffic, Poisson arrivals appears to be suitable for traffic at the session level when sessions are human initiated, e.g., interactive TELNET and FTP sessions [5].

The major limitation of the Poisson process is that it is unable to model very bursty data. The Poisson model shows that as the number of users and the amount of traffic increase, the traffic becomes smoother and less bursty [2]. Different Poisson models, such as Markov modulated Poisson process(MMPP) can only capture burstiness over a few time scales and short-range correlations [19]. Thus, at large levels of aggregation m(where m > 1), it will smooth out.

5.2.2 Compound Poisson model

One way to represent burstiness through a Poisson model is to use Compound Poisson model. In the compound Poisson model, the base Poisson model is extended to deliver batches of packets at once. The inter-batch arrival times are exponentially distributed, while the batch size is geometric. Mathematically, this model has two parameters, λ, the arrival rate, and ρ in (0,1), the batch parameter. Thus, the mean number of packets in a batch is 1/ ρ, while the mean inter-batch arrival time is 1/ λ. Mean packet arrivals over time period t are tλ/ ρ.

Though the compound Poisson model take batches of packets at once, still it shares the analytical benefits of the pure Poisson model such as:- the model is still memoryless, aggregation of streams

is still (compound) Poisson, and the steady-state equation is still reasonably simple to calculate, although varying batch parameters for differing flows would complicate the derivation.

However, to solve the problem of burstiness in network traffic (even if, it is more captured by self-similar traffic models), there are different non-self-similar traffic models that are preferable than compound Poisson traffic model such as the packet train model, which is discussed below.

5.2.3 Packet Train model

Another model that attempts to capture bursty traffic model is created by Jain and Routhier: the packet train model [20]. Packet train is defined as a burst of packets arriving from the same source and heading to the same destination. This model assumes that groups of packets travel together. Thus, the model minimizes the overhead of the routing decision since the routing decisions can be taken only when the head of a new packet train is detected. However, if packet arrivals are assumed to be independent, then routing decisions for packets traveling between the same endpoints are performed independently on routers, which lead to processing overhead.

The model groups all the packets flowing between two endpoints in to a packet train having the inert-car time (spacing between two packets) smaller than a specified number, called maximum allowed inter-car gap. The inter-car time must be significantly smaller than the time separating two distinct packet trains, the inter-train time. If the spacing between two packets, inter-car time, exceeds inter-train gap, then they belong to different trains. In this model, the inter-train time is a user parameter, dependent on the frequency with which applications use the network. The inter-car time for a train is a system parameter that depends on the network hardware and software.

The packet train model is characterized by the mean inter-car arrival time, mean inter-train arrival time, mean train size,…

5.3 Self-similar traffic models

Network traffic with self-similar characteristics can be estimated using ARMA, ARIMA, FARIMA, … traffic models. And from those traffic models some are capable of capturing short range dependence and others are capable of capturing long range dependence. In both cases the first step for model estimations is to test for the stationarity property of the data.

Stationary processes exhibit statistical properties that are invariant to shift in the time index, that is statistical properties such as mean, variance, etc. are all constant over time. Thus, for example, first order stationarity implies that the statistical properties of X_{t_i} and $X_{t_{i+c}}$ are the same for any c. Similarly, second order stationarity implies that the statistical properties of the pairs $\{X_{t_1}, X_{t_2}\}$ and $\{X_{t_{1+c}}, X_{t_{2+c}}\}$ are the same for any c. There are two important forms of stationarity: Strict Sense Stationary and Wide Sense Stationary.

Definition:- If for $t_1, t_2, \dots, t_n$ the joint distribution of the random vectors $\{X(t_1), X(t_2), \dots, X(t_n)\}$ and $\{X(t_{1+h}), X(t_{2+h}), \dots, X(t_{n+h})\}$ is same $\forall h$ then the stochastic process $\{X(t),\ t \in T\}$ is said to be strict sense stationary of order n. If the above definition holds for every integer n then the process is called a strict sense stationary Process.

Definition: - A Stochastic process is said to be wide sense stationary or weakly stationary or covariance stationary if the following properties hold: -

- m(t) = E $[X(t)]$ is independent of t.
- E $[X(t)]^2 < \infty$
- The covariance depends only on the time difference $t_1 - t_2$

The relation between strict and weak stationarity is that, Strict-sense stationarity implies wide-sense stationarity. But the converse is not true except the Gaussian Process.

Most statistical forecasting methods are based on the assumption that the time series can be rendered approximately stationary i.e. stationarized through the use of mathematical transformations. A stationarized series is relatively easy to predict: it predicts that its statistical properties will be the same in the future as they have been in the past. The predictions for the stationarized series can then be untransformed, by reversing whatever mathematical transformations were previously used, to obtain predictions for the original series. Thus, finding the sequence of transformations needed to stationarize a time series often provides important clues in the search for an appropriate forecasting model.

Another reason to stationarize a time series is to be able to obtain meaningful sample statistics such as means, variances, and correlations with other variables. Such statistics are useful as descriptors of future behavior only if the series is stationary. For example, if the series is consistently increasing over time, the sample mean and variance will grow with the size of the sample, and they will always underestimate the mean and variance in future periods.

To check stationarity in the series, the Augmented Dickey Fuller (ADF) test is mostly used [21]. The ADF test is unit root test for stationarity. Unit roots can cause unpredictable results in time series analysis. The ADF regression equation is given by:

$$\Delta y_t = \mu_0 + \mu_1 t + \emptyset y_{t-1} + \sum_{j=1}^{t} \alpha_j \Delta y_{t-j} + \varepsilon_t \qquad (21)$$

$$t = p+1, p+2, \dots, T$$

where μ_0 is the intercept, $\mu_1 t$ represents the trend in case it is present, $\emptyset$ is the coefficient of the lagged dependent variable, y_{t-1} and p lags of Δy_{t-j} with coefficients a_j are added to account for serial correlation in the residuals.

The hypotheses for this test are:

- The null hypothesis is that the series has unit root
- The alternative hypothesis is that the series is stationary.

To test for stationarity, the ADF test can be applied by using software. When using a software, the result is interpreted as: a p-value less than 5% means the null hypothesis is rejected that there is a unit root. It is also possible to compare the calculated ADF statistic with a tabulated critical value. The ADF test statistic is given by

$$ADF = \frac{\widehat{\emptyset}}{SE(\widehat{\emptyset})} \qquad (22)$$

Where SE ($\widehat{\emptyset}$) is the standard error for $\widehat{\emptyset}$. The null hypothesis of unit root is accepted if the test statistic is greater that the critical values.

After testing for stationarity, the series can be modeled using one of the self-similar traffic models. Self-similar traffic models can be classified as Short-memory model and Long-memory model. Next self-similar traffic models are discussed.

5.3.1 Short-Memory Models

Short-memory models refer traffic models that can capture short-range dependencies in traffic. In this section some Short-memory models are reviewed.

5.3.1.1 ARMA model

Given a time series $X(t)$, the ARMA model is a tool for understanding and predicting future values in the series. As the name implies, the model consists of two parts: an autoregressive (AR) part and a moving average (MA) part. The AR part involves regressing the variable on its own lagged (i.e., past) values. The MA part involves modeling the error term as a linear combination of error terms occurring contemporaneously and at various times in the past. The model is referred to as the ARMA *(p, q)* model where *p* is the order of the autoregressive part and *q* is the order of the moving average part. Before considering the ARMA *(p, q)* model, the AR and MA parts are discussed first.

Autoregressive Model

The autoregressive model is a model where the term X_t is dependent upon the previous terms. The structure of the model is linear, that is the model depends linearly on the previous terms, with coefficients for each term. In this model, as the term "Autoregressive" implies, the variable X_t is regressed on previous values of itself. It is essentially a regression model where the previous terms are the predictors.

The autoregressive model of order *p*, denoted as AR(*p*), is defined by the equation:

$$X_t = \alpha_1 X_{t-1} + \cdots + \alpha_p X_{t-p} + \varepsilon_t \qquad (23)$$
$$= \sum_{i=1}^{p} \alpha_i \, X_{t-i} + \varepsilon_t$$

Where $\alpha_1, \alpha_2, \dots, \alpha_p$ are the model coefficients, p is a non-negative integer and the error term ε_t is white noise, i.e. is uncorrelated over time with a constant variance and mean zero. For $p = 0$, $X_t = \varepsilon_t$ and there is no auto regression term.

Using the backshift operator, denoted by B, which is used to express lagged values of the process i.e. $BX_t = X_{t-1}, B^2X_t = X_{t-2}, \dots B^dX_t = X_{t-d}$, AR($p$) can be written as:

$$\theta(B)X_t = \left(1 - \alpha_1 B - \alpha_2 B^2 - \dots - \alpha_p B^p\right)X_t + \varepsilon_t \qquad (24)$$

$$= 1 - \sum_{i=1}^{p} \alpha_i B^i X_t + \varepsilon_t$$

(The backshift operator is also known as Lag operator and denoted as L for "lag" as B is for "backshift")

Moving Average Model

A moving average model is linear combination of the past white noise terms, unlike the Autoregressive model that is a linear combination of the past time series values. In this model, the observations of a variable at time t are not only affected by the shock at time t, but also the shocks that have taken place before time t. Mathematically, the moving average model of order q, denoted as MA(q), is defined as

$$X_t = \varepsilon_t + \beta_1\varepsilon_{t-1} + \dots + \beta_q\varepsilon_{t-q} \qquad (25)$$

$$= \varepsilon_t + \sum_{j=1}^{q} \beta_j \varepsilon_{t-j}$$

Using backshift operator notation, the MA(q) model is given by the equation:

$$X_t = \Phi(B)\varepsilon_t = (1 + \beta_1 B + \beta_2 B^2 + \dots + \beta_q B^q)\varepsilon_t \qquad (26)$$

$$= 1 + \sum_{j=1}^{q} \beta_j B^j \varepsilon_t$$

The combination of the two models, Autoregressive and Moving average model, results Autoregressive Moving average model ARMA (p, q). The ARMA model of order p and q is defined as:

$$X_t = \alpha_1 X_{t-1} + \dots + \alpha_p X_{t-p} + \varepsilon_t + \beta_1\varepsilon_{t-1} + \dots + \beta_q\varepsilon_{t-q} \qquad (27)$$

$$= \sum_{i=1}^{p} \alpha_i X_{t-i} + \sum_{j=1}^{q} \beta_j \varepsilon_{t-j} + \varepsilon_t$$

In backshift operator notation, the ARMA model is given by

$$\theta(B)X_t = \Phi(B)\varepsilon_t \qquad (28)$$

One of the techniques to estimate p and q parameters of ARMA (p, q) model is to carry out the Autocorrelation Function (ACF) and Partial Autocorrelation Function (PACF) analysis [22]. These statistical measures tell how the observations are related to each other. Plotting the ACF and PACF against consecutive time lags is useful in determining the order of AR and MA terms. ACF

shows the correlation of the series with itself. When computing autocorrelation, the result can range from 1 to -1. An autocorrelation of 1 represent a positive correlation where as an autocorrelation of -1 represent a negative correlation. ACF is also discussed in section (4.5).

Partial Autocorrelation function is the correlation between a variable and a lag of itself that is not explained by correlations at lower order lags. The partial autocorrelation at lag k is the autocorrelation between X_t and X_{t-k} that is not accounted for by lags 1 through $k-1$. It removes the effect of shorter lag autocorrelation from the correlation estimate at longer lags.

When plotting ACF and PACF plot, there is a confidence limit shown as a bar which tells that lags having values outside these limits(bars) should be considered to have significant correlation. So, from the ACF and PACF plot if ACF dies out gradually and PACF cuts off sharply this indicates AR term. Whereas, if ACF plot cuts off sharply after a few lags and PACF plot dies out gradually this indicates MA term. Thus, when PACF is significant till *p* lags, it signifies AR term, and when ACF is significant till *q* lags, it signifies MA term.

5.3.1.2 ARIMA model

The ARMA model, described above can be used for stationary time series data. However, in practice there are time series that are non-stationary. Thus, ARMA models are not good enough to properly describe non-stationary time series. For this reason, the ARIMA model is proposed, which is the generalization of ARMA model to include the case of non-stationarity as well.

The acronym ARIMA stands for Auto-Regressive Integrated Moving Average. It is a class of models for forecasting a time series which can be made to be stationary by differencing. Differencing is a way of transforming non-stationary series to a stationary series. This is done by subtracting the current period observation from the previous one. If the transformation is done only once to the series, then it is "first differenced".

In this model, the Auto-Regressive (AR) terms are lags of the stationarized series and the Moving Average (MA) terms are lags of the forecast errors. The integrated (I) series is a time series which needs to be differenced to be made stationary and it is an integer. ARIMA model can be summarized by three numbers: ARIMA (*p, d, q*) model where

p = the number of autoregressive terms

d = the number of differences needed for stationarity and

q = the number of moving average terms

An ARIMA (*p,0,0*) model is autoregressive model of order *p*, AR (*p*) and ARIMA (*0,0, q*) model is moving average model of order *q*, MA (*q*). An ARMA (*p, q*) process is also an ARIMA(*p, 0, q*) process. Mathematically, ARIMA model can be written as:

$$\grave{X}_t = c + \alpha_1 \grave{X}_{t-1} + \cdots + \alpha_p \grave{X}_{t-p} + \varepsilon_t + \beta_1 \varepsilon_{t-1} + \ldots + \beta_q \varepsilon_{t-q} \qquad (29)$$

Where $\grave{X}$ is the differenced series, c is a constant

Using backshift operator notation, differences can be expressed as $X_t - X_{t-1} = (1-B)X_t$, $(X_t - X_{t-1}) - (X_{t-1} - X_{t-2}) = (1-B)^2 X_t$, … Thus, the equation for ARIMA model is:

$$\theta(B)(1-B)^d X_t = \Phi(B)\varepsilon_t \qquad (30)$$

The first step in fitting an ARIMA model is to check for stationarity. And if the series is not stationary then stationarize the series and determine the order of differencing needed to stationarize the series. After the series has been stationarized, the next step is to identify the numbers of AR and MA terms needed that fit an ARIMA model. This can be done by looking at the ACF and PACF plot of the differenced series, if the data is from an ARIMA (*p, d, 0*) or ARIMA (*0, d, q*) model, as discussed to estimate the AR and MA terms of ARMA model (section 5.3.1.1). If the data have both AR and MA terms, the plots do not help in finding the proper values of *p* and *q* and it is difficult to select the appropriate values. In such cases, functions in statistical languages like R can be used to select the suitable values. Then, the last step is to fit the model that is suggested.

5.3.2 Long-Memory model

5.3.2.1 FARIMA model

The acronym FARIMA stands for Fractional Auto-Regressive Integrated Moving Average. The above discussed models, ARMA and ARIMA, can only capture the short-range dependence property, since *d* is confined in the range of integer order. Therefore, in order to capture the long-range dependence property, the Fractional ARIMA (*p, d, q*) model is proposed by Hosking [2]. The fractional ARIMA model is a generalization of ARIMA model which allows non-integer (real) values of the differencing parameter *d*. For a stationary timeseries X_t, the general form of FARIMA (*p, d, q*) model is defined as:

$$\theta(B)(1-B)^d X_t = \Phi(B)\varepsilon_t \qquad (31)$$

Where B is the backshift operator, $\theta(B)$ is the autoregressive polynomial, $\Phi(B)$ is the moving average polynomial, ε_t is the error term and *d* is the differencing order allowed to be fractional. The value of d is related to the Hurst parameter H by the relationship [23]:

$$d = H - 0.5 \qquad (32)$$

In fractional ARIMA (*p, d, q*) model, the parameter *d* determines the long-term behavior. The value of *d* is in the range $-0.5 < d < 0.5$. The range of interest in the context of long-memory is $0 < d < 0.5$ [2]. When the value of d is greater that 0.5 ($d > 0.5$), the model is nonstationary. And if the value of *d* is in the range $-0.5 < d < 0$, then the process has short-term memory.

Whereas the parameters *p* and *q* determine the short-term correlation, that is these parameters allow for more flexible modeling of short-range properties. By varying the autoregressive and moving average components, short-memory effects can be combined with long-memory effects. Thus,

fractional ARIMA model is capable of describing the long term properties through the parameter *d*, and the short term correlation through the parameters *p* and *q* simultaneously.

To construct a Fractional ARIMA model, the procedure used to fit to the data is as follows:

Step 1. Estimating fractional differencing parameter d

The value of *d*, which measures the strength of long-range dependence, can be estimated using the relationship $d = H - 0.5$, equation (30). The Hurst parameter H can be obtained using several methods such as variance-time plots and R/S analysis as described in chapter 4.

Step 2. Estimate the parameters p and q

This can be done by the estimation method for ARMA models, that is using the Autocorrelation function (ACF) plot and the Partial Autocorrelation function (PACF) plot as described in section 5.3.1.1.

Step 3. Test the model

To check for the model, whether it fits the observed series or not, Autocorrelation Function is used. The Autocorrelation function of the collected data and the fitted model is computed and the resemblance of the ACF of the fitted model to that of the collected data is noticed.

5.4 Traffic model uses

The development of robust networks heavily depends on the modeling of the networks. A good traffic model lead to a better understanding of the characteristics of the network traffic itself. Which, in turn can help designing routers and devices which handle network traffic. Also for network simulations, traffic models are needed as input. Thus, it is essential that the assumed models reflect as much as possible the relevant characteristics of the traffic it is supposed to represent. Inaccurate modeling of network can lead to performance problem, loss of money, …

One may ask why traffic models are needed. Would not traffic measurements be sufficient to design, control and mange networks? Unquestionably, for verifying the actual network performance measurements are useful and essential. However, measurements do not have the level of abstraction that makes traffic models useful. Traffic measurements only reflect current reality whereas traffic models can be used for hypothetical problem solving. In addition, traffic measurements give insight about a particular traffic trace where traffic models give insight about all traffic sources of that type.

Modeling traffic have many advantages. Some of the uses of traffic models are discussed here. One important use of traffic models is to properly dimension network resources for a target level of quality of service. Models of traffic are needed to estimate the bandwidth and buffer resources

to provide acceptable packet delays and packet loss probability. Knowledge of the average traffic rate is not enough. From queueing theory, it is known that queue lengths vary with the variability of traffic. Hence, an understanding of traffic burstiness or variability is needed to determine sufficient buffer sizes at nodes and link capacities [1].

The second use is to verify network performance under specific traffic controls. Modeling of network traffic is an imperative first step towards the formulation of effective algorithms that manage the limited resources in the network. It is important that the formulated algorithm is stable and allows multiple hosts to share bandwidth fairly, while sustaining a high throughput. And without realistic source model, effective evaluation of the stability, fairness and throughput of new algorithm would not be possible. The third use of traffic models is admission control. In particular, connection oriented networks depend on admission control to block new connections to maintain quality of service [1].

Chapter 6: Materials and Methods

Here, on this section the tools used and methods carried out to conduct this research is described. The experimental setup and tools used on this research are listed on Table 2.

Table 2. Experimental Setup

Processor	Intel(R) Core(TM) i5 CPU @ 2.20GHz
Installed memory (RAM)	8.00GB
System Type	64-bit Operating System
Statistical Programming Language	R
Integrated Development Environment (IDE)	RStudio Version 1.0.136
To capture network packets	Cisco Security Device Manager (SDM) and Wireshark

To analyze the WAN egress traffic data, first data is collected and the process that characterize the data is determined. Then knowing the characteristics of the traffic on the network, a traffic model is chosen for modeling it.

6.1 Traffic Measurement

On this work, real traffic measurements taken from Wegagen Bank of Ethiopia (Main branch) is used. The network structure of the bank is shown in figure 11. As it is shown, the edge router is connected to the Internet Service Provider(ISP) in two ways: Fiber Data and Aironet data. And all branches in the country access the database through this edge router. The data is collected using the cisco Security Device Manager (SDM) tool.

Also, real traffic measurements are collected from Bunna Bank of Ethiopia, to make the result more reliable. To collect data from Bunna bank another tool is used, i.e. Wireshark tool. A different tool is used to have data from the two banks for the reason that the two banks employ different application programs to monitor their network traffic. On this work, since WAN is the focus, the data i.e. number of packets at specified time is taken from the edge router of the banks.

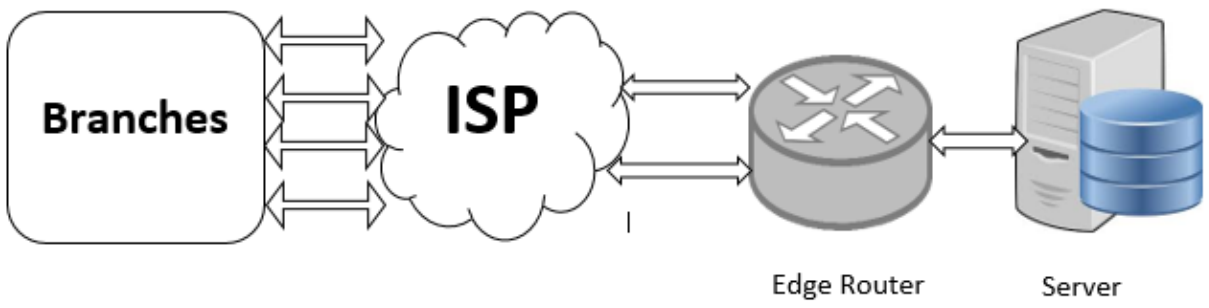

Figure 11. Network structure of Wegagen Bank

The Cisco SDM is a web-based device management tool embedded within Cisco Internetwork Operating System (IOS) access routers. Cisco SDM simplifies router and security configuration through offered multiple smart wizards, enabling customers to deploy, configure and monitor a cisco access router quickly and easily without requiring knowledge of the cisco software command-line interface (CLI).

The Cisco SDM monitor mode provides an option to overview router status, performance metrics such as interface status, CPU and memory usage and other various options to monitor the status of different router features. The interface status screen displays the current status of the various interfaces on the router, and the number of packets, bytes, or errors that have travelled through the selected interface.

The other tool used for this research is Wireshark. Wireshark is an open source tool used to capture data packets to allow a more precise analysis. Such a tool is often referred to as a network analyzer, network protocol analyzer or sniffer. The focus of this tool is observing the data traffic within a network. Wireshark tool have a simple graphics user interface (GUI) and it is user friendly, easy to use tool.

The menu of the Wireshark tool provides options like File: which contains items to open, save, print, or export capture files and to quit the Wireshark application; Capture: which allows to start and stop captures and to edit capture filters; Statistics: which contains items to display various statistic windows including a summary of the packets that have been captured, display user specified graphs (e.g. the number of packets in the course of time) and much more; and other more options.

For analysis more than 3000 data are collected from each bank at different time periods. Since the data is collected from banks which contains secured information, there is limitation to have more data. Then, after gathering data i.e. number of packets at specified time, to understand the process the data exhibits, self-similarity is tested.

6.2 Testing Self-similarity

As discussed in the previous chapters, Hurst parameter H is used to measure self-similarity. Therefore, in order to detect the presence of self-similarity in WAN egress traffic, Hurst parameter is estimated. To Estimate Hurst parameter, the two most popular methods are used i.e. Variance-time plots and R/S plots (as mentioned in chapter 4). If the value of Hurst parameter H is found to be between 0.5 and 1, $0.5 < H < 1$, it implies the traffic possess self-similar process.

For statistical analysis purpose several software programs have been developed, such as MATLAB, SAS, R, … Here, on this work R programming language is used for analyzing the data. R is a free, open source programming language that helps to find solutions for statistical problems easily using the built-in statistical functions and add-on packages available in it. Like other computer languages, R has its own naming conventions i.e. set of rules to denote variables, functions, … To interact with R more readily, there is an Integrated Development Environment (IDE) known as RStudio. RStudio helps to keep R more organized and adds more functionality to it. R has extensive documentation to learn more about the functions.

Although R comes with many common statistical functions and packages, still one may require additional packages. Packages are collection of additional functions that can be loaded when needed. There are world-wide R package repositories or Comprehensive R Archive Network (CRAN) sites that allow packages to be downloaded and installed. Here, for this research packages like tseries, pracma… are used from the CRAN site.

To estimate for Hurst parameter H for testing self-similarity using the variance time plot method, the code is written using R programming language on R studio following the procedure (mentioned in chapter 4). For R/S analysis method, R library is used to estimate H. After estimating H and understanding what characteristics WAN data possess whether it self-similar or non-self-similar, can proceed to modeling the data.

6.3 Modeling

To model a traffic data first the characteristics of the series should be determined. Then depending upon the characteristics of the traffic, traffic model will be chosen to model the data. If the data possess Non-self-similar characteristics, the series will be modeled by one of the Non-self-similar traffic models i.e. Poisson model, compound Poisson model, … But if the data have self-similar characteristics rather than non-self-similar characteristics, self-similar traffic models will be opted to model the series i.e. ARIMA, FARIMA, …

If the WAN egress traffic data have self-similar characteristics, to apply the self-similar models, the series should be tested whether it is stationary or not. If the data is found to be stationary, one can move directly to fit the series to one of the self-similar models. However, when testing stationarity if it is found to be non-stationary, first one should stationarize the series using different

methods like differencing. Then after making the series stationary, it can be fitted to one of the self-similar traffic models.

On this research to check stationarity, the Augmented Dickey Fuller (ADF) test is used. The ADF test examines the null hypothesis that a time series is non-stationary against the alternative hypothesis that it is stationary. This is expressed in terms of a probability (P-value). P-values are an important number in hypothesis tests that quantify the strength of the evidence against the null hypothesis in favor of the alternative. If p-value is greater than 0.05, it implies the null hypothesis is accepted i.e. it is non-stationary. Else if p-value is less than 0.05 then alternate hypothesis is accepted i.e. it is stationary. Hence, stationarity is tested applying ADF test separately to the collected data from the two banks.

Also, since the data being long-range dependent or short-range dependent lets to choose the best traffic model it fits, it is valid to know if the series is long range dependence(LRD) or short-range dependence(SRD) processes. Thus, LRD is tested. And this is done using autocorrelation function (ACF) of the traces. The autocorrelation function of long-range dependence processes has a hyperbolic decay while the autocorrelation function of short-range dependence process has an exponential decay. Accordingly, ACF of the collected data from the two banks is computed individually on RStudio.

The next phase is model identification. That is, if WAN egress traffic data possess self-similar characteristics and it is SRD then the short-memory self-similar traffic model is chosen. Else if the WAN data exhibits self-similar characteristics and it is LRD then the long-memory self-similar traffic model is selected. Here initially stationarity is tested in both cases.

Model identification is followed by parameter estimation for the picked model. ARIMA (p, d, q) and FARIMA (p, d, q) models have three parameters i.e. p, d and q. To estimate the parameters p which stands for Auto regressive term and q which stands for Moving average term, the ACF and PACF plot is used. Thus PACF of the collected data from the two banks is computed. And then the parameters p and q are estimated from the plots. The parameter d in ARIMA model is the number of differencing needed to stationarize the series. And in FARIMA model, the parameter d which is the fractional differencing is calculated applying equation (32).

When estimating parameters p and q in ARIMA and FARIMA model using the ACF and PACF plots, if both p and q are positive, then the plots don't help in finding proper values of p and q. Thus, if this is the case the R functions can be used to select the suitable model parameters automatically. However, if the data are from ($p, d, 0$) or ($0, d, q$) model, that is one of the plots dies out gradually and the other cut off sharply, then the ACF and PACF plots are used in estimating the values of p or q.

At last the model is tested whether it fits the observed series or not. This is done by computing the ACF of the fitted data. Then the resemblance of the ACF of the fitted model to that of the collected data is noticed.

Chapter 7: Results and Discussion

In this section, the results obtained by analyzing the WAN egress traffic data applying the concepts and steps discussed in the previous chapters is shown. Hence, the process that characterize the data is determined first. Next a traffic model is chosen for modeling it after knowing the characteristics of the traffic on the network.

7.1 Traffic measurement

For this work, real traffic measurements are taken from the edge router of Wegagen Bank of Ethiopia and Bunna Bank of Ethiopia. The collected data are number of packets at specified time using SDM and Wireshark tools. The collected data from Wegagen bank is shown in figure 12. Also Figure 13 and Figure 14 shows the collected data from Bunna bank, in two time units. The figure illustrates that there exists a burst like traffic in all time scales. Thus, burstiness is preserved.

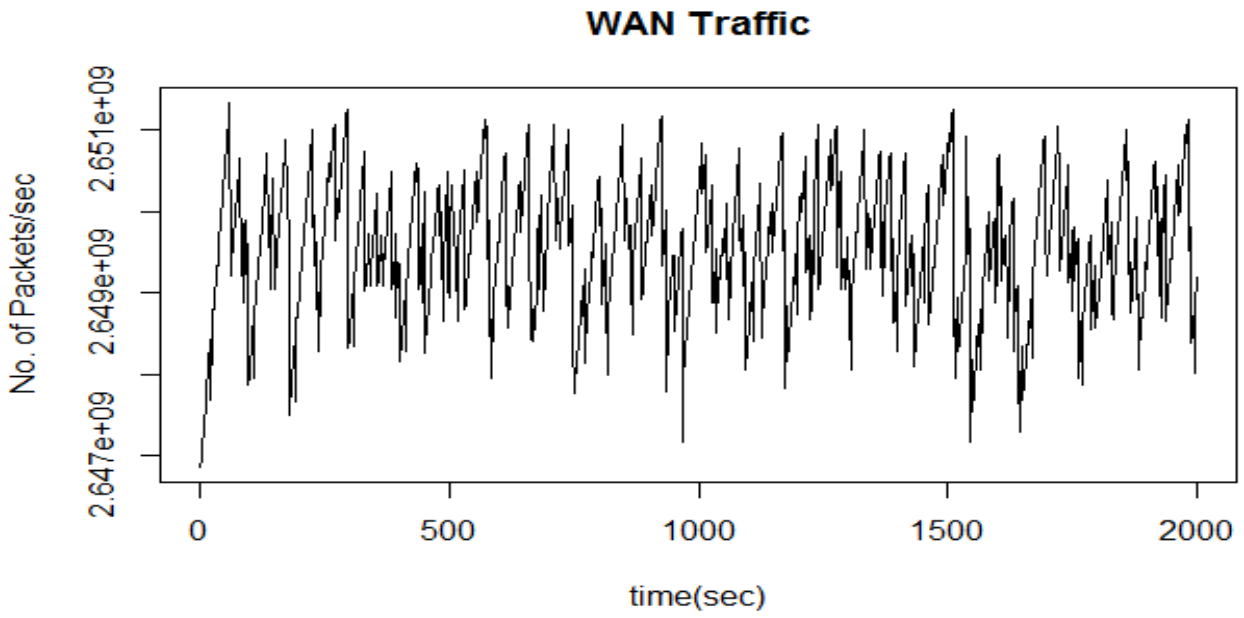

Figure 12. WAN traffic data of Wegagen Bank

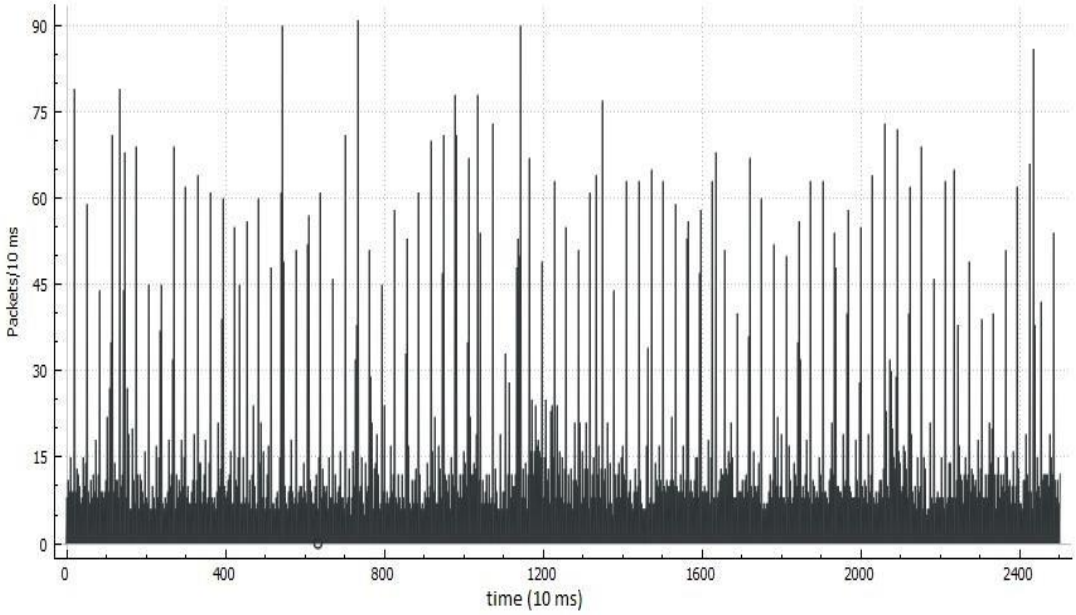

Figure 13. WAN traffic data, 10ms time unit

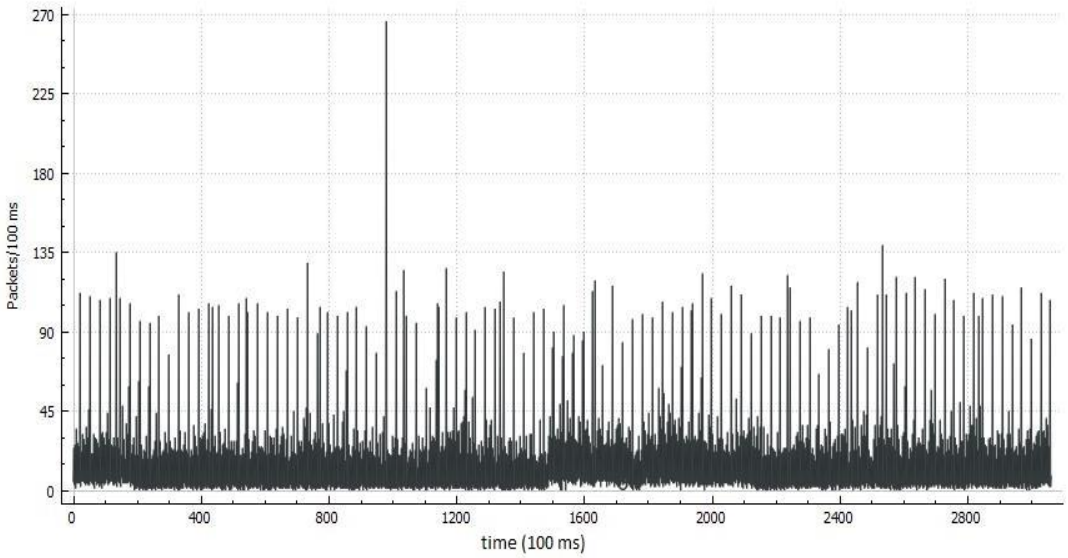

Figure 14. WAN traffic data, 100ms time unit

7.2 Testing Self-similarity

In order to detect the presence of Self-similarity Hurst parameter H is estimated since H is used to measure Self-similarity. And to estimate H, variance-time plot and R/S methods are used.

Analyzing the data collected from Wegagen Bank using the Variance time plot method, the slope is found to be -0.6566, resulting a Hurst parameter estimation of 0.6717 (using the equation, $H = 1 + \beta/2$). And using R/S analysis method the slope is found to be 0.6694, which gives the estimate of Hurst parameter H.

Likewise, applying the Variance Time plot method for the data collected from Buna bank results a slope of -0.4366, which gives a Hurst parameter estimation of 0.7817. And using R/S analysis method yields a slope of 0.7707, which gives the estimation of Hurst parameter H.

The estimated Hurst parameter H is given on Table 3 and 4. And, since the Hurst parameter obtained in both scenarios is $0.5 < H < 1$, it can be concluded that the series have self-similarity characteristics.

Table 3. Hurst parameter H of Wegagen Bank with different methods

Method	Hurst parameter H
Variance time plot method	0.6717
R/S method	0.6694

Table 4. Hurst parameter of Buna Bank with different methods

Method	Hurst parameter H
Variance time plot method	0.7817
R/S method	0.7707

7.3 Modeling

To understand the characteristics the data exhibits, self-similarity is tested on the above sub-section using the variance-time plot and R/S method which leads to choose the right model that fits the data. As shown in the previous sub-section, the WAN egress traffic data have self-similar characteristics. Hence, self-similar traffic models will best fit the series.

Prior to decide on one of the self-similar traffic models, stationarity of the data is tested using ADF test. In addition, LRD is also tested. In the next sub-sections, the result found in testing stationarity and testing long range dependence will be described.

7.3.1 Testing for Stationarity

To apply the self-similar models, first stationarity of the variable should be examined. The summary of the Augmented Dickey Fuller test applied to both collected traffic data is shown in figure 15 and figure 16. The P-value obtained, as shown in the figures, is 0.01 that is less than 5%. Therefore, it can be concluded that the series is stationary. Hence no need to stationarize.

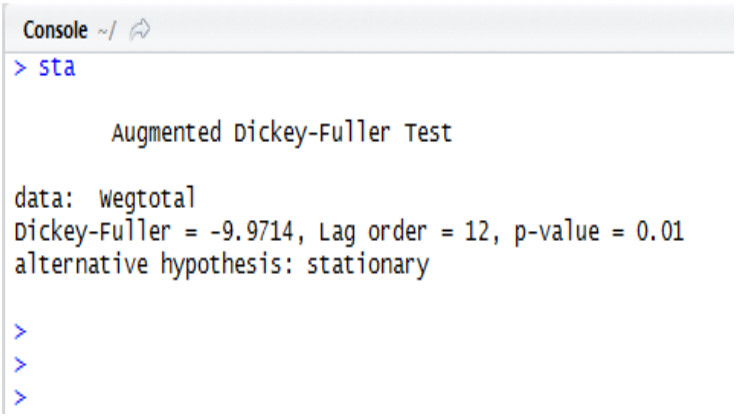

Figure 15. ADF Test of Wegagen Bank data

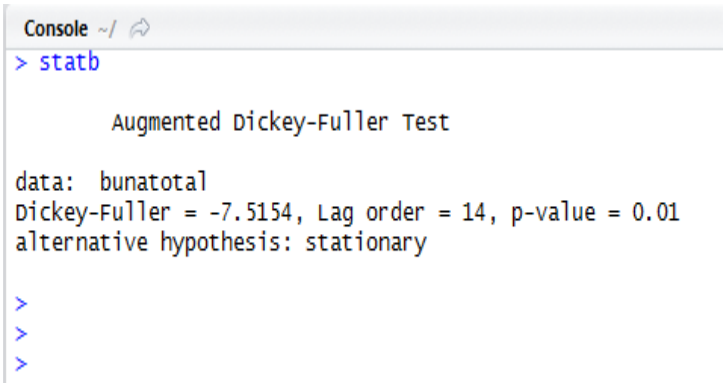

Figure 16. ADF test of Bunna bank data

7.3.2 Testing for Long Range Dependence

Indication of long-range dependence is given by the autocorrelation functions (ACF) of the traces. Using R programming language, the autocorrelation function is computed and found as shown in figure 17 and figure 18. On the figures, the horizontal broken lines show the confidence interval. From the figure, it can be seen that the autocorrelation function is decaying slowly, which implies the traffic data is Long-range dependent or has Long Memory.

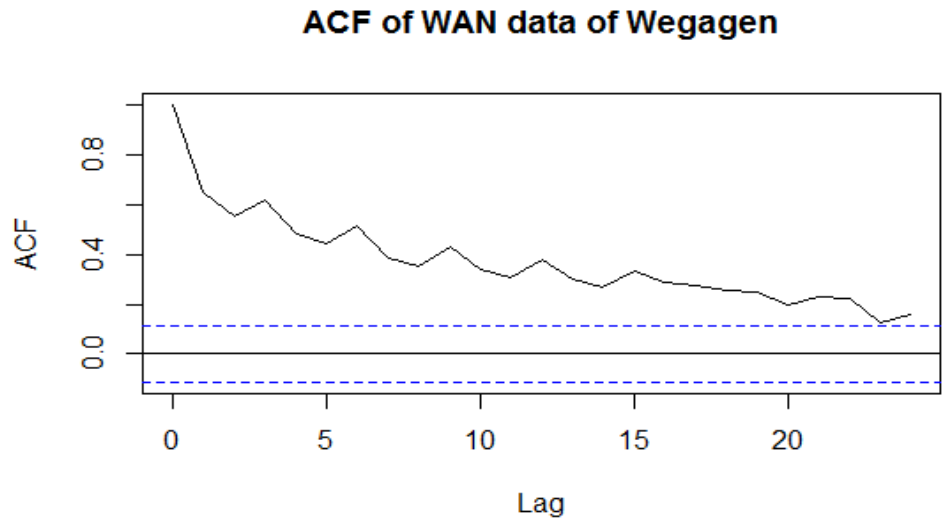

Figure 17. Autocorrelation Function of WAN data of Wegagen Bank

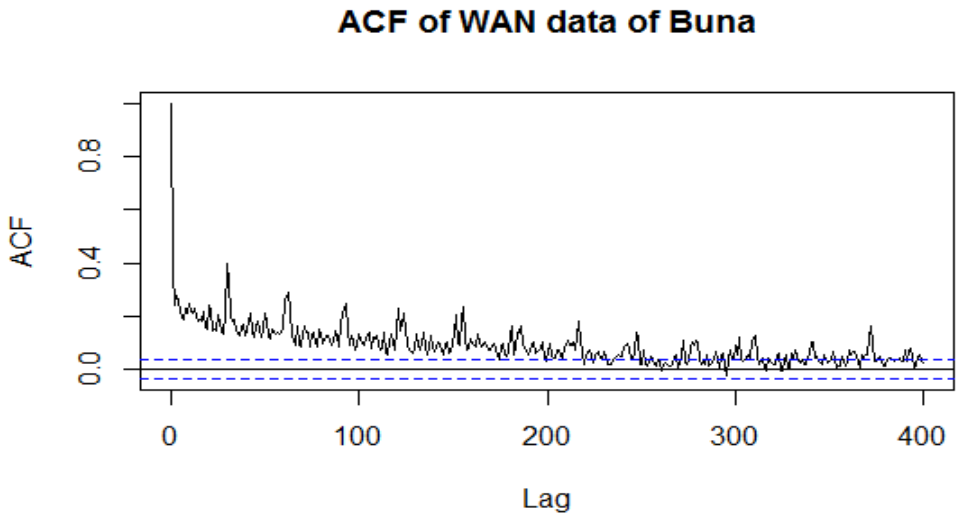

Figure 18. Autocorrelation Function of WAN data of Buna Bank

7.3.3 Model Identification and Parameter estimation

Thus, because the data is long-range dependent and also has self-similar characteristics, the long-memory self-similar traffic model is selected that is Fractional Auto Regressive Moving Average FARIMA (p, d, q) model. Next parameters of FARIMA model is estimated. As discussed on section (5.4.2) the parameters of FARIMA (p, d, q) model stands for Auto regressive term(AR), fractional differencing and Moving average term(MA) respectively. To construct FARIMA model, the first step is estimating the fractional differencing parameter d applying equation (32). And it is found to be 0.17 for the data from Wegagen Bank and 0.2 for the data from Buna Bank, which

suggests the series displays a long-term memory in both scenarios since it is in the range $0 < d < 0.5$.

The next step is to estimate the parameters p and q. To estimate the values, in addition to the autocorrelation function the partial autocorrelation function is also required. So, the partial autocorrelation function is computed using R programming language and it is found as shown in figure 19 and 20.

Figure 19 show that the PACF plot has significant spike till lag 3 then it cuts off and on figure 20 the PACF plot has significant spike till lag 4 then it cuts off. This indicates that all the higher order autocorrelations are explained by the lag 3 and lag 4 autocorrelation effectively. Thus, the PACF determines how many AR terms are needed to use to demonstrate the autocorrelation pattern in a time series. And also, figure 17 and figure 18 show that the ACF dies out gradually. Therefore, since when ACF dies out gradually and PACF cuts off sharply indicates AR term, the value of parameter p can be estimated to be $p = 3$ for the data from Wegagen Bank and $p = 4$ for the data from Buna Bank.

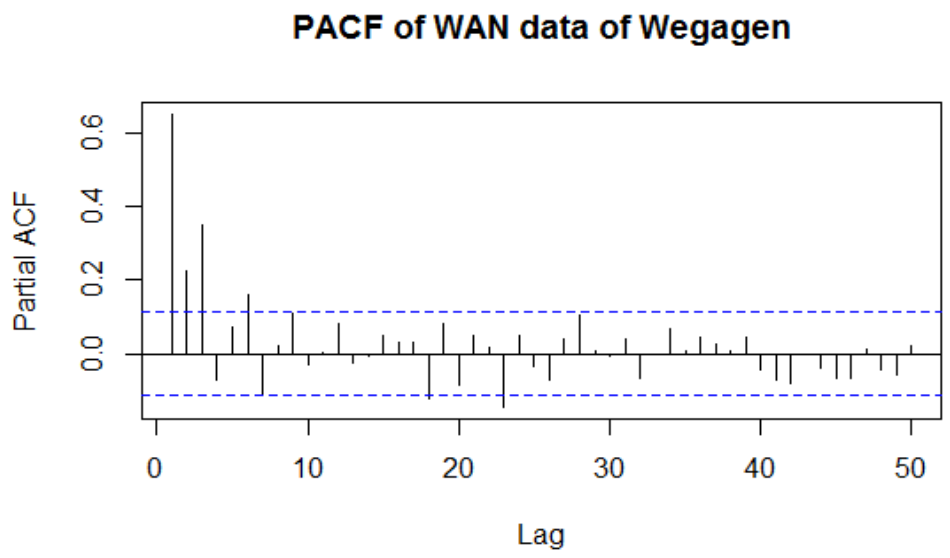

Figure 19. Partial ACF of WAN data of Wegagen Bank

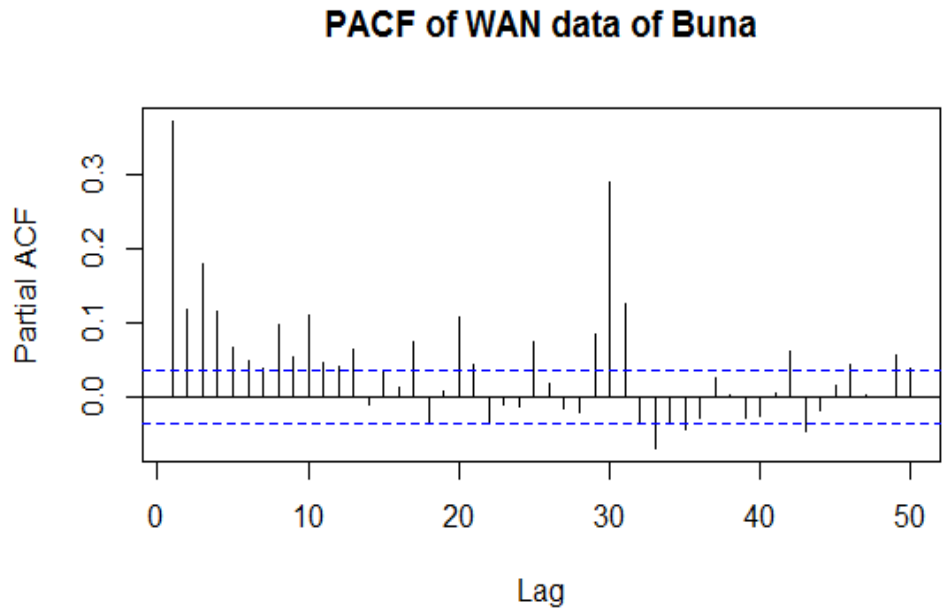

Figure 20. Partial ACF of WAN data of Buna Bank

And in both scenarios, there is no MA term. So, the model for the data from Wegagen Bank is estimated as FARIMA (3,0.17,0) and for the data from Buna bank FARIMA (4.0.2,0).

Now lets check if the model fits the collected data. And this is done by computing the autocorrelation function of the collected data and fitted data. Figure 21 and 22 shows the obtained result.

As the figure shows, there is resemblance of the autocorrelation function plot between the collected data and the fitted data. Therefore, it can be concluded that FARIMA model is capable of capturing Wide Area Network traffic.

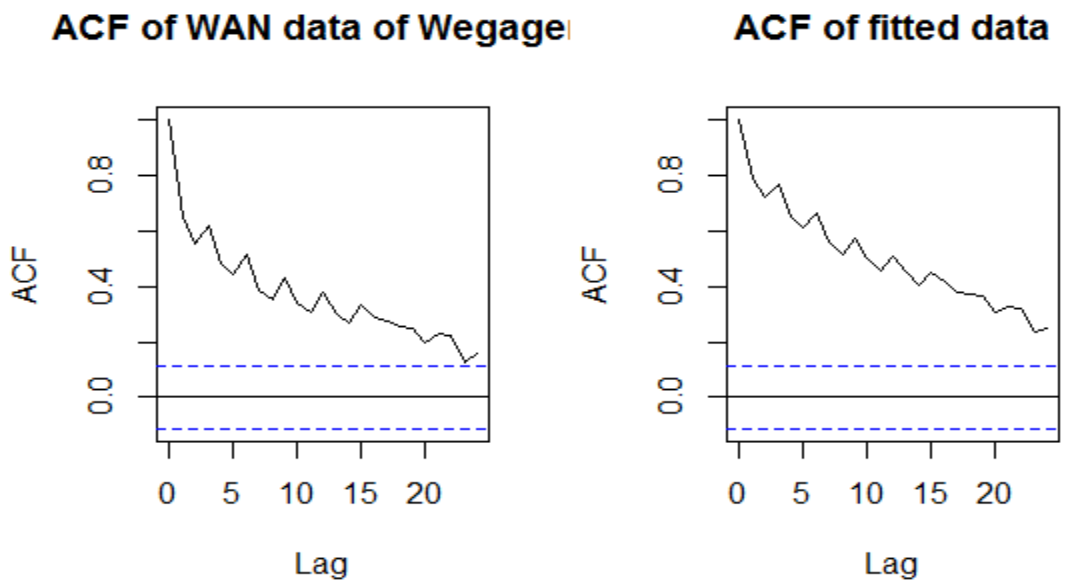

Figure 21. ACF of collected data Vs fitted data of Wegagen Bank

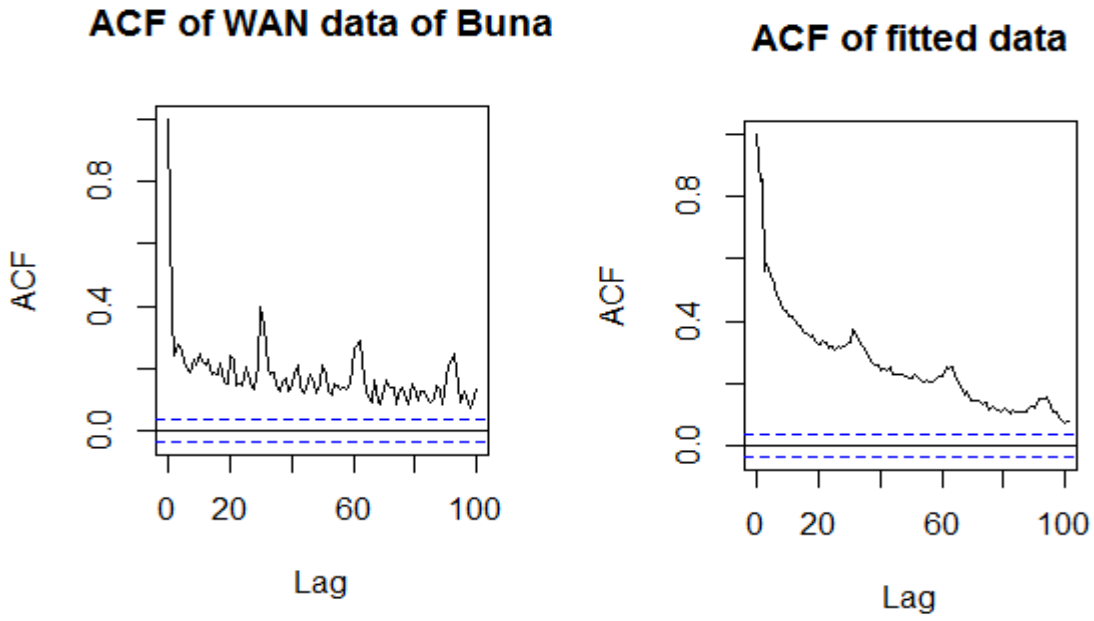

Figure 22. ACF of collected data Vs fitted data of Buna Bank

Chapter 8: Conclusion and Recommendation

A computer network is an interconnection of different networking devices located at different places for the purpose of sharing resources and information. A computer network can be classified as Local Area Network (LAN) and Wide Area Network (WAN) based on the geographical distance the network covers. WAN covers large geographical areas than LAN. And WAN has more congested traffic flow than LAN. In this research work, WAN traffic data from two banks of Ethiopia is analyzed to understand the characteristics of WAN egress traffic data and to determine traffic model that captures WAN egress traffic.

To determine the characteristics that WAN egress traffic possess, the Variance Time plot and R/S method are used, and the results of the analysis show that WAN egress traffic possess Self-similar characteristics. The degree of Self-similarity is measured in terms of the Hurst parameter H which also represents the burstiness of the traffic. In this work, the Hurst parameter is found to be around 0.7. Also, from Autocorrelation Function (ACF) plot of the data, it is showed that WAN traffic is Long-Rang dependence rather than Short -range dependence. That is, it has long memory.

After understanding the characteristics of the data, a traffic model that can capture WAN traffic is determined. From the result it is observed that WAN traffic data possess Self-similar characteristics and has long memory, thus the Self-Similar Long memory traffic model, that is Fractional Auto Regressive Integrated Moving Average Model (FARIMA) model, is found to best captures WAN traffic. And it is shown that ACF plot of the approximated FARIMA model fits nicely with the ACF of the collected data. FARIMA model is a good model for WAN traffic since it can capture both the long memory and short memory property of the data.

Even though modeling network by gathering traffic data is difficult, understanding the model of the network is important for designing network devices, … On this research, data is collected from two banks of Ethiopia and it is demonstrated that WAN traffic data exhibit Self-Similar characteristics and has long memory. Also, it is showed that FARIMA model captures the traffic data. For future work it is recommended to test the characteristics and model network traffic of other industries traffic.

References

[1] T. M. Chen, "Network Traffic Modeling," in *The Handbook of Computer Networks*, Hossein Bidgoli(ed), wiley, 2007.

[2] W.Leland, M.Taqqu, W.Willinger and D.Wilson, "On the self-similar nature of Etherent traffic(Extended version)," IEEE/ACM Transactions on Networking, Vol. 2, No. 1, February, 1994.

[3] W.Leland, M.Taqqu, W.Willinger and D.Wilson, "On the self-similar nature of Ethernet Traffic," Proceedings of ACM SIGCOMM'93, 1993.

[4] Y. Negash, "Analysis of Ethernet Traffic," *Journal of EAEA,* vol. 17, 2000.

[5] Vern Paxson and Sally Floyd, "Wide-Area Traffic: The falilure of Poisson Modeling," IEEE/ACM Transaction on Networking, 1995.

[6] M.E.Crovella and A.Bestavros , "Self-Similarity in World Wide Web Traffic: Evidence and Possible causes," IEE/ACM Transaction on Netowrking, Vol 5, No 6, 1997.

[7] J.Beran, R.Sherman, M.S.Taqqu and W.Willinger, "Long-Range Dependence in Variable-Bit Rate video Traffic," IEEE Transactions on Communications,Vol 43, No. 2, 1995.

[8] K.Park, G.Kim and M.Crovella, "On the effect of self-similar on network performance," In Proc. of the SPIE International Conf. on Performance and control of network system, 1997.

[9] L.Muscariello, M.Mellia, M.Meo, M.Marsan, R.Cigno, "Markov models of internet traffic and a new hierarchical MMPP model," Computer Communications, 2005.

[10] A.Dainotti, A.Pescape, P.Rossi, F.Palmieri, G.Ventre, "Internet traffic modeling by means of Hidden Markov Models," Computer Networks, 2008.

[11] Moshe Zukerman, Timothy D. Neame and Ronald G.Addie, "Internet Traffic Modeling and Future Technology Implications," IEEE INFOCOM, 2003.

[12] T. Lammle, CompTIA Network+ Study Guide, Sybex.

[13] Dimitar Randev and Izabella Lokshina, "Advanced Models and Algorithms for Self-Similar IP Network Traffic Simulation and Performance Analysis," Journal of Electrical Engineering, Vol. 61, No. 6, 2010.

[14] Kihong Park and Walter Willinger, Self-Similar Network Traffic: An Overview.

[15] J. Beran, Statistics for Long-Memory Processes, US of America: CHAPMAN & HALL, 1994.

[16] W.Willinger, Murad S.Taqqu, Will.E.Leland and Daniel.V.Wilson, "Self-Similarity in High-Speed Packet Traffic: Analysis and Modeling of Ethernet Traffic Measurements," Statistical Science, 1995.

[17] M. Ghaderi, "On the Relevance of Self-Similarity in Network Traffic Prediction," University of Waterloo, Waterloo, Ont N2L 3G1, Canada.

[18] Thomas Karaglannis,Mart Molle, and Michalls Faloutsos, "Long-Range Dependence," *IEEE Computer Society,* 2004.

[19] Ian W.C.Lee, Abraham O.Fapojuwo, "Stochastic processes for computer network traffic modeling," Computer Communication, 2005.

[20] Raj Jain and Shawn A.Routhier, "Packet Trains- Measurement and a new model for Computer Network Traffic," *IEEE Journal on Selected Areas in Communications,* vol. 4, no. 6, pp. 986-995.

[21] Omekara C.O. , Okereke O.E. , Ukaegeu L.U., "Forecasting Liquididty Ratio Of Commercial Banks in Nigeria," 2016.

[22] Ratnadip Adhikari, R.K.Agrawal, An Introductory Study on Time Series Modeling and Forecasring.

[23] Kai Liu, YangQuan Chen and Xi Zhang, "An Evaluation of ARFIMA (Autoregressive Fractional Integral Moving Average) Programs," *Axioms,* 2017.