PERFORMANCE OF VOICE OVER IP UNDER VARIOUS UNICAST QUALITY ROUTING ALGORITHMS

By

Ala' Fathi Mohammed Khalifah

Supervisor
Dr. Jamil Ayoub, Prof.

Co-Supervisor
Dr. Ali Elmousa

This Thesis was Submitted in Partial Fulfillment of the Requirement for the Master's Degree of Science in Electrical Engineering / Communications

Faculty of Graduate Studies
University of Jordan

Jan, 2004

This thesis was successfully defended and approved on 8/1/2004

Examination Committee	**Signature**
1) Prof. Jamil Ayoub	____________
2) Dr. Ali El Mousa	____________
3) Dr. Androws Sweedan	____________
4) Dr. Mohamed Hawa	____________
5) Dr. Abid Al Rahman Al zubidi	____________

Dedication

To my dear parents.
To my dear brothers and sister.
To everyone who looks desirably for my success.

Acknowledgement

I am thankful and grateful to Allah, who supported me in my thesis. My great respect and thanks to my supervisors; Prof. Jameel Ayoub and Dr. Ali Elmousa, for their endless support and encouragement. My thanks to Dr. Peng Zhang from Helsinki University of Technology, for his support and guidance in completing the simulation part of this thesis.

My great gratitude and respect to my family for their continuous encouragement and patience during my study days, special thanks to my dear sister, Hanan, who helped me a lot in printing this thesis and applying the final touches to it.

Finally, I would like to thank my dear friends, Qusai Marashdeh, Amjed Al Farran, Mousa Ayyash , Bilal Hamzeh , and Jamal Al Karaky for their help , in providing me with some papers and references, which I used in my thesis.

Table of Contents

List of Tables

List of Figures

Abbreviations

BE	Best-Effort
CBQ	Class Based Queue
DiffServ	Differentiated Services
DSCP	Differentiated Services CodePonit
DV	Distance Vector
FCFS	First-Come-First-Served
FDM	Frequency Division Multiplexing
FTP	File Transfer Protocol
HTTP	Hyper Text Transfer Protocol
IETF	Internet Engineering Task Force
IntServ	Integrated Services
ISPs	Internet Service Providers
ITGs	Internet Telephony Gateways
ITSPs	Internet Telephony Service Providers
ITU	International Telecommunication Union
LS	Link State.
MaRS	Maryland Routing Simulator
MOS	Mean Opinion Score
NSFNET	National Science Foundation Network
OSPF	Open Shortest Path First
PBX	Private Public Exchange
PCM	Pulse Coded Modulation
PSTN	Public Switched Telephone Network

QoS	Quality of Service
QOSPF	Quality of Service Open Shortest Path First
QRS	Quality of Service Router Simulator
RIP	Routing Information Protocol
RM	resources Manager
RSVP	Real-Time Reservation Protocol
RT	Real-Time
SMTP	Simple Mail Transfer Protocol
SPF	Shortest Path First
SWP	Shortest Widest Path
TDM	Time Division Multiplexing
UDP	User Data Protocol
UTP	Unshielded Twisted Pairs
VoIP	Voice over IP
WLC	Widest Least Cost
WSP	Widest Shortest Path

Introduction

The future Internet is expected to accommodate a large number of applications with diverse service requirements. The Internet, however was designed for non-real time data communications, such as the current Hyper Text Transfer Protocol (HTTP) service, File Transfer Protocol (FTP) service, and Simple Mail Transfer Protocol (SMTP) service. All of these non real-time services do not have any quality of service requirements or assurance mechanisms, such as the degree of sensitivity to time delay and packet loss. But in the last few years, the Internet has witnessed a tremendous growth, (Li et al., 2000) and it has a great potential for providing a wide variety of services, of a real-time nature, in which the delivery time, is one of the most crucial requirements. The major real-time applications are: Voice over IP (VoIP), and video streaming (Durkin, James.2003).

Voice over IP

Using the Internet to carry phone conversations, also known as Internet telephony or voice over IP, is taking the telecommunications industry by storm, since it represents the best opportunity so far for companies and individuals to facilitate voice and data convergence.

It allows the building and administration of only one data network, which is capable of carrying both voice and data, instead of establishing two separate infrastructures, one for voice and one for data. This is one advantage of implementing VoIP. Moreover, it also promises to deliver cheap long and short distance telephone calls.

Video Streaming

Streaming refers to an application that generates a constant stream of data to send either audio or video information across the IP network. Video streaming can be in the form of video on demand, where it is carried, point-to-point from the source to a receiver. Video conferencing is another important service that is expected to be widely implemented in the next generation Internet (Li et al., 2000).

Implementation Challenges

Many challenges face the process of implementing VoIP and video streaming applications, and prevent them from being widely used and extensively deployed (Crawley et al., 1998). This issue can be better illustrated, if we have an overview of the Internet infrastructure and how it was built. Originally the idea behind the Internet infrastructure was to exchange data and information between the universities and the Ministry of Defense in the United States of America. This service was available to a limited number of people and not for the public. Later on, this service started to spread among the public and the organizations. Some new services such as the mail service and the news service were introduced, but all of these services do not require any kind of real-time delivery, and do not take into account the end-to-end delay, for example, or the quality of service. The most important thing was to receive an acknowledgment message from the receiver's side, to inform the sender that the sent data was successfully received.

The Internet consists of many routers connected together via different communication media and links. When the data packet reaches the router, a routing algorithm is used to calculate the best path for that packet, and the packet is routed to its destination using that path. The problem is that the current Internet does not differentiate between real-time traffic, such as that of voice and the video traffic, and the regular Internet traffic, which is called best-effort traffic. So when real-time traffic reaches a router, it has to wait in the traditional queue until the best-effort traffic is served, then the router can handle and route the real-time traffic using the same routing algorithm used in calculating the best path. This algorithm is called Shortest Path First (SPF), for the best-effort traffic. In fact, the current queuing strategy and routing algorithm may not be suitable to real-time applications, such as voice over IP and video conferencing, and may result in excessive delay, which may be more than the acceptable delay limit determined by different standardization authorities and organizations. Of course, that will lead to unacceptable voice or video quality. To solve this problem, different QoS frameworks were suggested (Braden et al., 1994). Many working groups have been established; their duty was to suggest new quality of service architectures and models that can be implemented on the traditional data network, which must take the real-time traffic into consideration, and must have special treatment for real-time flows, in which the end-to-end delay, and the lost packets are very important factors to be handled and investigated so, to achieve that, special a queuing scheme was suggested, which will differentiate between different traffic types, so the real-time traffic will have special queue, different from the best-effort traffic queue .

A new routing strategy called quality of service routing was designed, which takes more than one metric into consideration when determining the best path for the real-time traffic. This is because shortest path isn't always the best path for real-time traffic (Chen, 1999). Moreover, a new reservation mechanism was suggested, which will use a reservation protocol called Real-time Reservation Protocol (RSVP) (Zhang et al., 1993). RSVP checks the available links for the requested bandwidth by the real-time applications, and once that path which meets the real-time bandwidth requirements is available, it will be reserved for the whole session, until the real-time source stops sending packets or some fault occurs to that link.

Thesis Purpose and Scope

We are investigating methods to improve the performance of VoIP through finding an appropriate QoS routing algorithm that can be easily deployed with minimum computational complexity.

This algorithm is supposed to route voice packets to non-congested links, thus reducing the end-to-end delay, and improving the voice performance. Also different metrics accompanied with the chosen QoS routing algorithm will be investigated. Moreover, the impact of deploying real-time services such as VoIP, and video streaming on the current best-effort Internet traffic will be taken into consideration. We hope to find the appropriate algorithm, which will improve the performance of real-time services, and reduce the effect of running these services on the current best-effort traffic networks.

The well known, unicast, intra-domain Open Shortest Path First (OSPF) routing protocol, will be used in our simulation, after introducing some QoS extensions to it, making it able of handling real-time flows. Our simulation model will adopt the Integrated Services (IntServ) QoS architecture model, suggested by the Internet Engineering Task Force (IETF).

Thesis Structure and Layout

In Voice over IP part, we will define voice over IP; study the benefits of it, and its advantages over the traditional Public Switched Telephone Network (PSTN). The architecture of voice over IP networks will be discussed, mentioning the major components required to enhance the current Internet, to make it capable of supporting this service. After that, the challenges and the problems that exist, that might limit the spread of this service are discussed with the proposed solutions suggested by the Internet research community.

Quality of service part is dedicated to study the concept of quality of service, which is essential to have an acceptable voice over IP quality. Two main frameworks suggested by the IETF for supporting QoS are discussed; the Integrated Services (IntServ) and the Differentiated Services (DiffServ) architecture. Then, the concept of QoS routing is mentioned, depicting its role in improving the performance of real-time services. Then the simulation scenarios and models, used in this thesis are discussed. Results and discussion of the simulation will be mentioned. Finally, the conclusion and future work are summarized.

Voice over IP

1 Preview

In the last few years, there has been a significant change in telecommunications technology, with emphasis on the growth of data communications market, demanding cheap and efficient telephony service. Voice over IP is expected to be extensively deployed in the near future, because of its potential for delivering cheap and efficient telephony services with accepted quality of service (Held, 2000). This section will discuss this technology, studying the major components of VoIP network, and studying the challenges that may affect the performance of this service.

2 VoIP Benefits and Merits

Voice over IP, from its name, can be defined as the process of transporting voice signals over an IP network (Held, 2000), in fact, interest in voice over IP has been significantly increased over the past few years. Enterprises, Internet Service Providers (ISPs), Internet Telephony Service Providers (ITSPs), and carriers view VoIP as a viable way to implement packet voice. Although the Internet was designed to handle non-real time data traffic (Wang, Crowcroft, 1996), that is being used increasingly to carry voice and video. One important real-time service, which utilizes the Internet infrastructure, is VoIP. It has the capability of interfacing with the existing telephone network, but needs the existence of an Internet Telephony Gateways (ITGs), which perform protocol translation between an IP network and the public switched telephone network. In order for the Internet to constitute an attractive alternative to the traditional PSTN, it must provide high quality VoIP service. This means that service providers must maintain certain a level of quality of service that attracts the user and make him use it instead of the current PSTN. Many reasons exist for implementing VoIP such as: toll bypass for the PSTN network, network consolidation, and service convergence (Held, 2000). Toll-bypass allows long-distance calls to be placed without incurring the usual toll charges, since this technology will no longer relay on the international telephone lines in conducting these calls. Instead, it will depend on the Internet infrastructure for carrying these voice calls in the form of voice packets, similar to the data packets.

Through network consolidation, voice, video, and data can be carried over a single network infrastructure. That simplifies network management and reduces cost, through the use of common equipment capable of handling all of these traffic types. However, designing a VoIP network requires careful planning to ensure that voice quality can be properly maintained. In this section, we will examine the factors that affect the voice quality, and the factors, which degrade the performance of such a service.

Typical Voice and Data Network

Figure 1 illustrates the classical method for integrating voice and data via time division multiplexing.

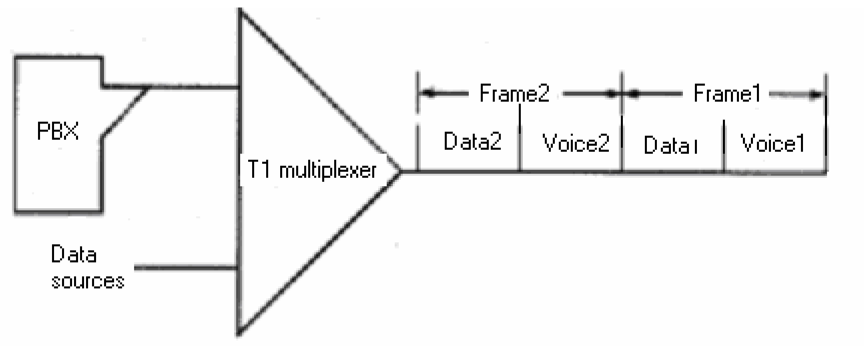


Figure 1 Data / Voice Multiplexing (Held, 2000)

It can be seen that the multiplexed frame is configured to provide a static allocation of bandwidth between voice and data sources. For example, if the private public exchange (PBX) was configured to provide 20 pulse coded modulation (PCM) voice conversations, each operating at 64 Kbps, then, each frame would consist of 1.28 Mbps (64 Kbps x 20) of bandwidth allocated to the PBX and the remaining bandwidth of 256 Kbps to the data sources excluding signaling control bits. Since the slots in the frame are fixed by time, if the data sources became inactive, the PBX could not take advantage of this available bandwidth.

In implementing voice over IP, more bandwidth utilization can be achieved, and there are many advantages over the traditional voice multiplexes and networks. Below are some of these advantages and characteristics gained in implementing VoIP.

Advantages of Voice over Data Networking

Bandwidth Allocation

IP represents packet-shared networks, for which bandwidth is consumed only when transmission occurs. This mechanism removes the fixed bandwidth allocation associated with the traditional multiplexers described above; which leads to more bandwidth utilization and consumption.

Modern Voice-Compression Techniques

A PCM-digitized voice conversation normally requires 64Kbps-operating rate (MINO, Minoli, 1998). However, many new compression/decompression techniques have been introduced. Using these techniques, the voice rate may be reduced to 4 or 8 Kbps, This means it is possible to transmit between 8 and 16 voice conversations on one 64Kbps link, while using the traditional multiplexers, a PCM-digitized voice conversation requires a 64Kbps operating rate, that means on the same bandwidth only one conversation can take place using the traditional multiplexers, while a larger number of calls with a lowered bit-rate can be obtained using the IP infrastructure, using the same 64 kbps bandwidth

Reliability

Reliability is another added value for implementing VoIP. This can be illustrated if we investigate a typical backbone infrastructure of packet networks, which is commonly constructed of mesh topology. This topology is illustrated in Figure 2.

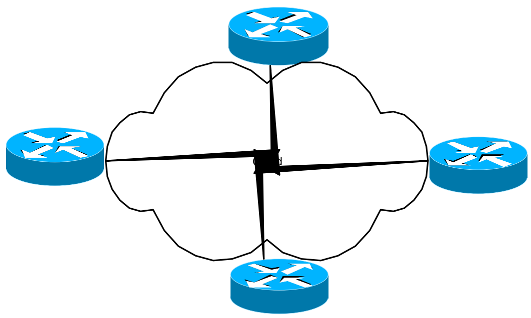


Figure 2 Redundancies in IP Network

The mesh structure can provide a large number of alternative routes between network nodes. For example, if the direct line connection between one of the two routers, shown above become inactive, transmission between those two locations could continue on another path. Thus, the backbone mesh structure of packet networks provides a built-in alternative routing capability that leads to high reliability.

Economics of Use

Finally, the tremendous economical saving which is achieved by implementing this service is one of the most attractive reasons which make people use it,, and push them to enhance their data network to make it able to a accommodate voice services over the same data network. Transmitting voice packets over the data network will cost the organization or the individual the same cost for sending an email or requesting an html page. On the other hand, since voice packets will be transmitted using the regular data network, almost the same cost will be encountered, other than the one-time payments for equipments to support voice digitization and packetization. In addition, payment may be needed for the VoIP gateway, which you will be used when conducting calls.

3 Voice over IP Definition and Architecture

An easy approach to understand how to build VoIP network is to study the current public switched telephone network. It simply consists of the following components:

- Telephone handset: which allows the user to connect to the telephony network, and convert speech into electrical signals ready to be transmitted to the exchange where the telephone is connected.
- The Exchange, which is the core of the telephony system. It has many functions including call setup and termination, call status monitoring, billing, and a multiplexing process, which multiplexes/demultiplexes different calls using either Time Division Multiplexing (TDM) or Frequency Division Multiplexing (FDM).
- Signaling protocol: in order to achieve the above functions, the exchange uses signaling protocol such as the SS7 protocol.

Similar to the PSTN network, VoIP network, consists of the following parts:

- IP telephone or usual telephone handset. This part is similar to the typical phone handset, used in the PSTN, which acts as the user terminal or end point. In VoIP, this end point can have different forms; it may be no more than a microphone with a headphone connected to a computer, which is running special software to act as soft phone. This can be seen in the PC-to-PC scenario described later. Another form, which might exist, is to have a special device, called an IP phone. This device has a similar interface to the typical telephone device, but this device must be connected to a usual switch using the Unshielded Twisted Pair (UTP) cables, similar to the cables used in connecting the computers together in the data network. Finally, VoIP can use a typical telephone handset, provided that this device is connected with proper interfaces to the VoIP network, this scenario is described in the PSTN to PSTN scenario.

- Gateway / Server. The gateway and the server provide the user with similar functionality that the exchange provides, such as the ring tone, call setup and termination, and billing. Moreover, one of the most important functions of the server, called sometimes proxy or gatekeeper, (Reynolds, Rix, 2001) is the mapping between the telephone number that the user dials, when issuing a telephone call, to the corresponding destination IP address, which represents another VoIP gateway. To allow the typical telephone handset to conduct a VoIP call, a special kind of interface device must be installed in the gateway. Finally, the switching and multiplexing process which is done in the exchange is mapped to IP forwarding and routing, for the voice packets.
- Signaling protocol. Similar to the SS7 signaling protocol that exist in the PSTN network, and is used by the gateway and the server, there are two main signaling protocols used in VoIP: the H.323, and the SIP protocol (Huitema et al., 1999). Figures 3, and 4 show these basic components for VoIP network.

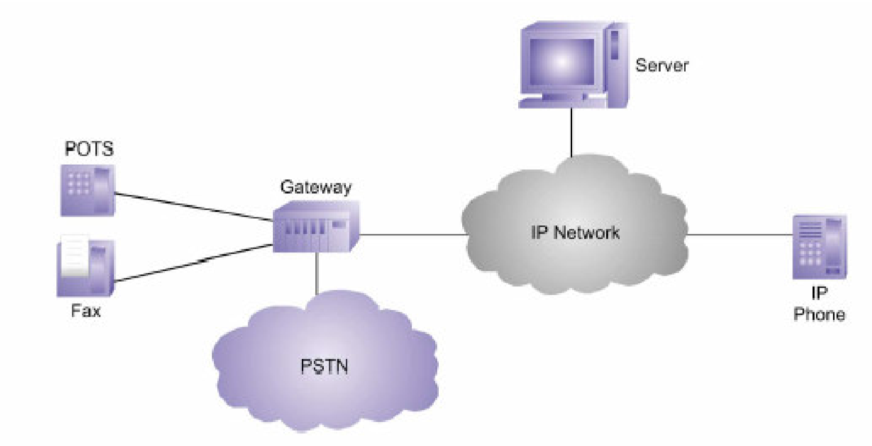


Figure 3 Major VoIP Components

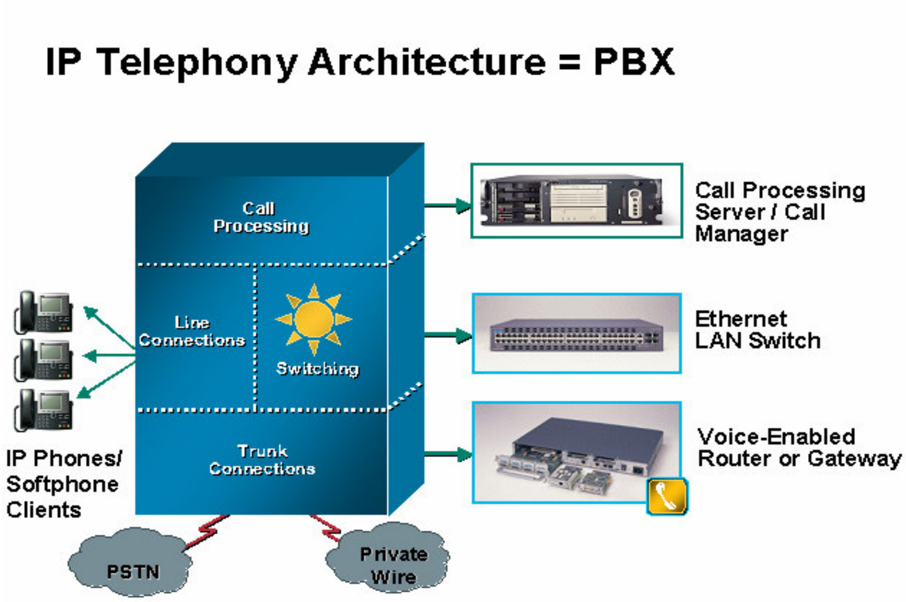


Figure 4 Major VoIP Components

In fact, different scenarios can take place in a VoIP call; Figure 5 shows all the possible scenarios with different possibilities.

- PC-to-PC voice communication.
- PSTN phone to PSTN phone.
- PC to PSTN phone.

All the above scenarios utilize the following steps in performing a VoIP call: The analog voice signals must be converted to digital signals, so it can be transmitted to the IP network. After that, an encoding technique is implemented using different voice-encoding schemes. Table 1 shows some of these coding techniques (Mahbub et al., 2000). The source and destination voice encoders and decoders must implement the same coding scheme, so that the destination device can successfully recover the analog waveforms. Once a voice signal is digitally encoded, it becomes just another form of data for the network to transport. These steps are shown in Figure 6.

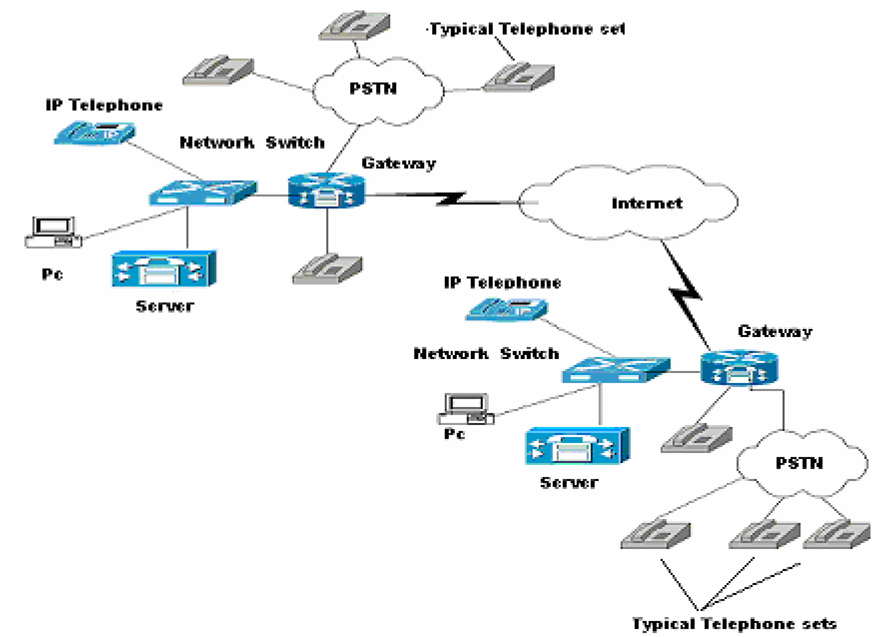


Figure 5 VoIP Scenarios

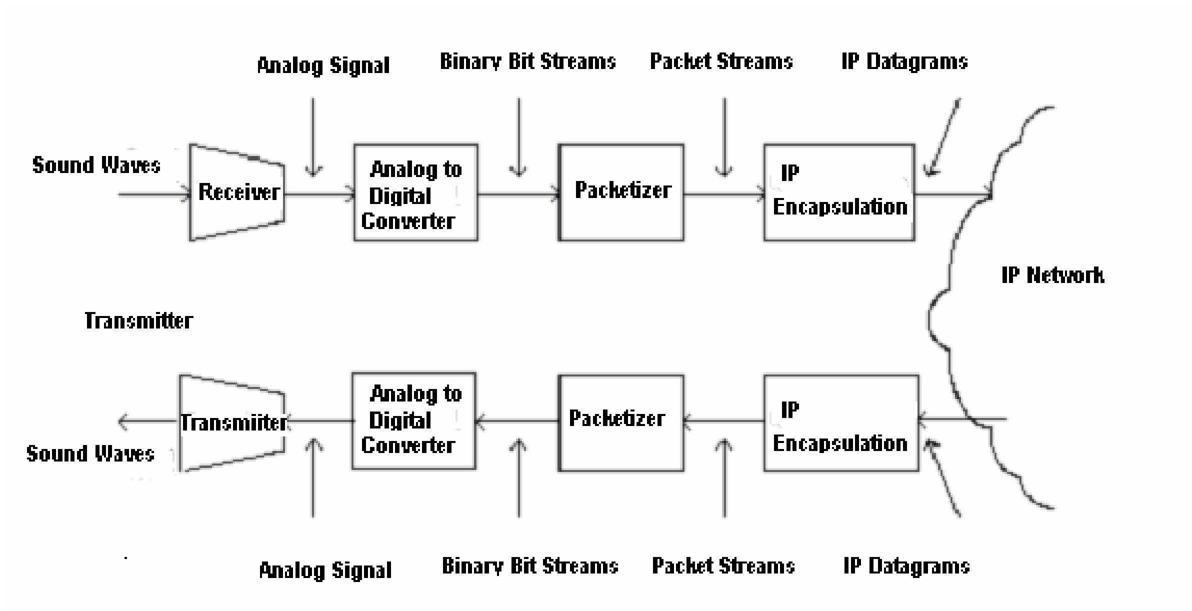


Figure 6 VoIP Stages

Table 1 VoIP Coding Techniques (Mahbub et al., 2000).

No.	Coding standard	Compression algorithm	Bit-rate	Frame processing delay (ms)	Look ahead delay (ms)	Total encoding delay (ms)	Typical decoding delay (ms)
1	G.711	PCM	64	0	0	0	0
2	G.729	CS-ACEIP	8	10	5	15	7.5
3	G.723.1	ACELP	5.3/6.4	30	7.5	37.5	18.75

4 Quality of Service Requirements

In order to have a good VoIP service that may constitute an acceptable alternative to the PSTN network, it is extremely important that such a service must meet some quality of service regulations and standards such as: end-to-end delay, also called latency, packet loss, and delay variation which is also called jitter.

End-to-End Delay

End-to-end delay or latency is the time delay incurred in speech by the IP telephony system. One-way latency is the amount of time measured from the moment the speaker utters a word, until the listener actually hears the word (Held, 2000). Round trip latency is the sum of the two one-way latency figures that compose the user's call. The lower the latency, the more natural interactive conversation becomes and the additional delay incurred by the VoIP system is less discernable, (Ma, 2001). Delay below 150 ms is acceptable for most applications. As delays exceed 150 ms, the performance of voice decreases. However, delays between 150 and 250 ms are still acceptable for long distance communications (Mahbub et al., 2000).

Types of Delay

Delays encountered by the VoIP packets can be classified into four main parts (Mahbub et al., 2000):

1) Codec delay: codecs perform voice compression to reduce the bandwidth requirements of voice transmission over digital networks. Higher compression will lead to lower bandwidth, but this is achieved at the price of longer delay. Table 1 lists encoding and decoding delays for several voice coding standards standardized by the International Telecommunication Union (ITU).

2) Serialization and processing delay: which defines the time required to place a packet on the transmission line, after being processed by the router.

3) Queuing delay occurs in the communication nodes: such as routers and gateways, where voice packets wait behind other packets, waiting to be transmitted over the same outgoing

link. In fact, this delay is considered the most important delay that may increase the end-to-end delay for real-time packets.

4) Propagation delay. It is the time required for signals to travel from one point to another point. This delay can be calculated knowing the distance and the speed of light. This delay becomes significant in long distances, such as transmitting data through the satellite link. Figure 7, which shows the different types of delays in the IP network.

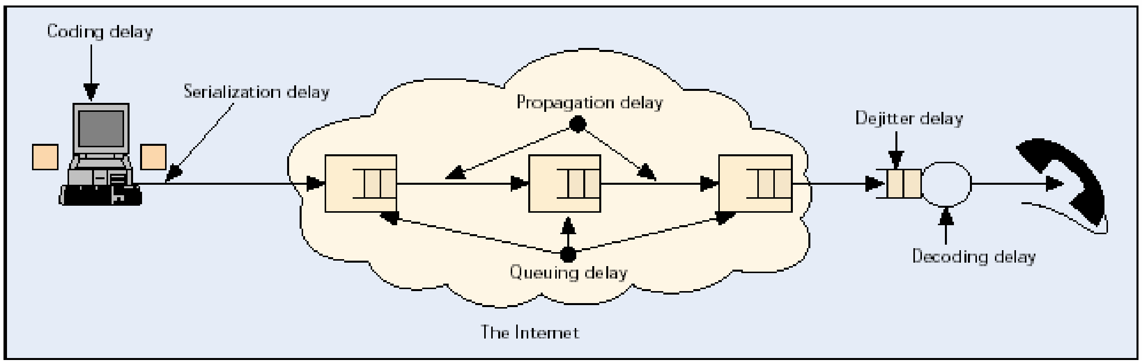


Figure 7 Delay Types (Mahbub et al., 2000)

Packet Loss

Usually, real-time services use the User Data Protocol (UDP) for transmission. The reason behind using this protocol is that the transmitter will not wait for receiving an acknowledgment from the receiver for the sent packets, before sending the next packet. Thus, no re-transmission mechanism exists in case of lost packets. In fact this transmission method seems logical for real-time services, because when some packets get lost, it is better for the receiver to ignore these packets and receive the new ones, than to ask the transmitter to re-transmit these packets again, which may lead to un-acceptable delay. Therefore, lost packets is an important issue that should be minimized as much as possible. Below, Figure

8 shows how the Mean Opinion Score (MOS), which is a measure for voice quality, is sharply affected by packet lost.

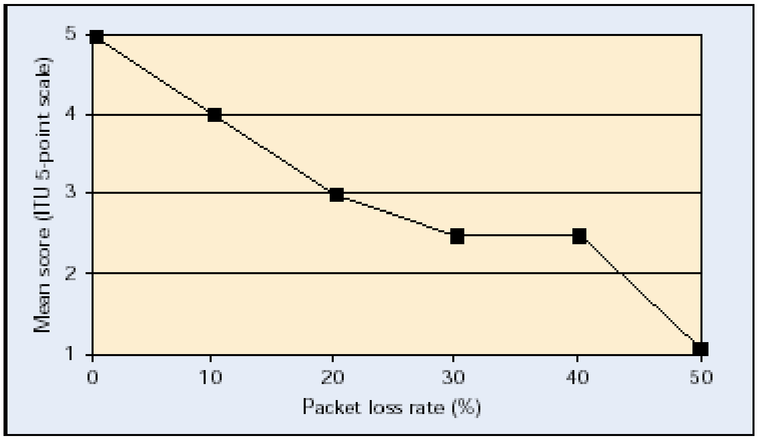


Figure 8 MOS versus Packet Loss, (Mahbub et al., 2000)

Jitter Delay

Because IP networks cannot guarantee the delivery time of data packets or their order, the data will arrive at inconsistent rates. The variation in inter-packet arrival rate is known as jitter, which is introduced by variable transmission delay over the network. Figure 9 shows different packets arriving at different times.

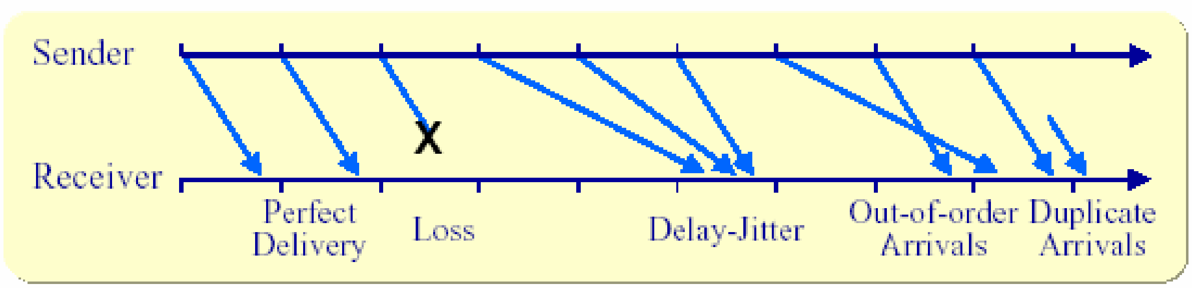


Figure 9 Packets Arriving Cases (Kevin, 1999)

To reduce the variation on the delay, some buffering technique at the receiver's side is used. This buffer is called de-jittering buffer. The idea behind it is that when the packets are received at different arrival rates, it is not necessary to play back each packets that arrive at the receiver's side, the de-jitter buffer may introduce some extra delay, to assure that all the stream of packets, which might arrive at different arrival times and rates, are played back uniformly at the receiver's side. Figure 10, shows how the packets are delayed, according to a pre-defined playout latency value.

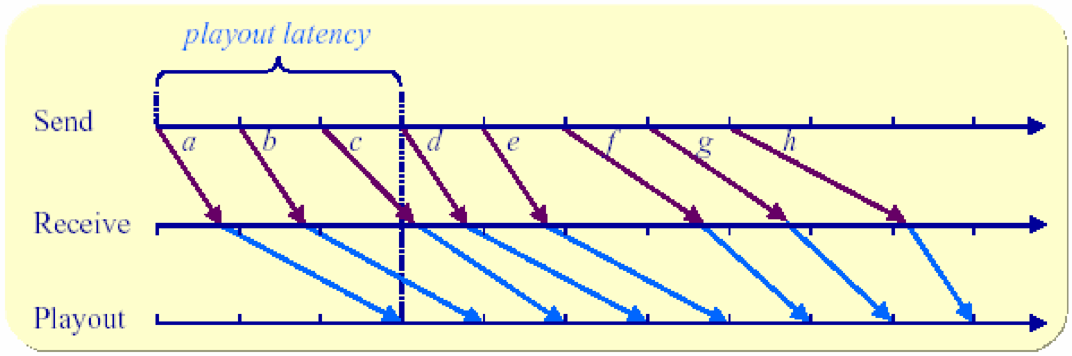


Figure 10 De-jittering Technique (Kevin, 1999)

5 Voice over IP Performance Evaluation

Mean Opinion Score (MOS)

Described in ITU-T P.800, MOS is the most well known measure of voice quality. It is a subjective method of quality assessment. Test subjects judge the quality of the voice transmission system either by carrying on a conversation or by listening to speech samples. They then rank the voice quality using the following scale:

5 – Excellent, 4 – Good, 3 – Fair, 2 – Poor, 1 – Bad.

MOS is then computed by averaging the scores of the test subjects. Using this scale, an average score of 4 and above is considered as toll-quality.

6 Thesis Objectives

After studying this interesting technology, and understanding how it can be implemented, and what are the factors that determine the quality of this service, we will try to improve the performance of VoIP by choosing suitable QoS routing algorithm. As described in the introduction section, the current Internet is designed to carry best-effort traffic, that doesn't require any time assurance or certain QoS requirements, so the typical shortest path first routing algorithm is used. In fact, this routing algorithm will not be suitable for carrying real-time services, such as the VoIP. Instead, QoS routing algorithm can be used. We are trying to find a suitable QoS routing algorithm, that is capable of carrying the voice packets along the Internet, with the minimum end-to-end delay, with minimum variation in the delay, and with the maximum throughput (minimum packet loss); thus obtaining high performance. In the next section, we will study the concept of quality of service, and the concept of QoS routing. Then, we will choose some efficient QoS routing algorithms, and study the performance of VoIP under them.

Quality of Service

1 Preview

As discussed in quality of service section, many challenges and problems must be overcome, in order to successfully implement VoIP service. One of these challenges is developing and enhancing the current Internet infrastructure, to differentiate between best effort, and real-time traffic loads. To achieve that, two main architectures are suggested by the Internet Engineering Task Force: the Integrated Services (IntServ) and Differentiated Services (DiffServ) models (Wang, 2001). Moreover. These models require the existence of a new routing algorithm, called QoS routing algorithm (Wang, Crowcroft, 1996) that is capable of handling QoS requirements, submitted by real-time applications. This part will discuss the above QoS architectures, with more emphasis on the IntServ architecture, used in our study The QoS routing subject is also discussed, mentioning the QoS extensions added to the well known open shortest path first routing protocol, to make it able to respond to real-time QoS requirements (Apostolopoulos et al., 1999).

2 QoS Definition

Quality of service generally describes the assurance of sufficiently low delay and packet loss for certain types of applications or traffic (Zhao et al., 2000). The requirements are given by real-time applications in the form of a delay bound, that should not be exceeded, or a certain amount of bandwidth that should be reserved and assured (Kuipers et al., 2003). The original service model of the Internet promised best-effort packet delivery, which is insufficient for many classes of applications like real-time video conferencing and VoIP that are sensitive to delay and packet loss. As such, the internetworking infrastructure has been undergoing a tremendous amount of changes (Crawley et al., 1998), to support real-time quality of service applications. Extensions are necessary at all levels of the network, starting from developing and enhancing new QoS signaling protocols, such as the Resource Reservation Protocol (RSVP) (Zhang et al., 1993), developing new queuing mechanisms that take into consideration traffic types so higher priority is given to real-time flows (Floyd, Jacobson, 1995), and developing new QoS routing algorithms capable of selecting routes that meet the QoS demands and requirements. Consequently, the application requesting a particular type of service can be given end-to-end quality assurances.

3 QoS Frameworks

The IETF, has suggested two main QoS architectures to make the Internet able to handle real-time services, and to overcome the challenges and problems currently exist. These architectures are:

- Integrated services architecture (IntServ).
- Differentiated services architecture (DiffServ).

Below, a brief description of both architectures is depicted, with more emphasis on the IntServ architecture, used in our thesis. Moreover, a simple comparison between the two architectures is presented, noting the reasons behind selecting the IntServ architecture model in our study and simulation.

3.1 Integrated Service Architecture

In early 1990, the IETF started the integrated services working group (Schooler, 1997), to standardize a new resource allocation mechanism and a new service model, to be able to support real-time traffic on the current Internet architecture. Real-time flow can be defined as a "distinguishable stream of related datagrams that results from a single user activity and requires the same QoS" (Braden et al., 1994). The IntServ architecture is based on per-flow resource reservation (Schooler, 1997). So an application must make a reservation on the available resource to receive resource assurance.

Integrated Services Reference Model

Figure 11 shows the major components in the reference model for the IntServ. The model can logically be divided into two parts: (Wang, 2001)

- Control plane, which sets up resource reservation.
- Data plane, which forwards data packets based on the reservation state.

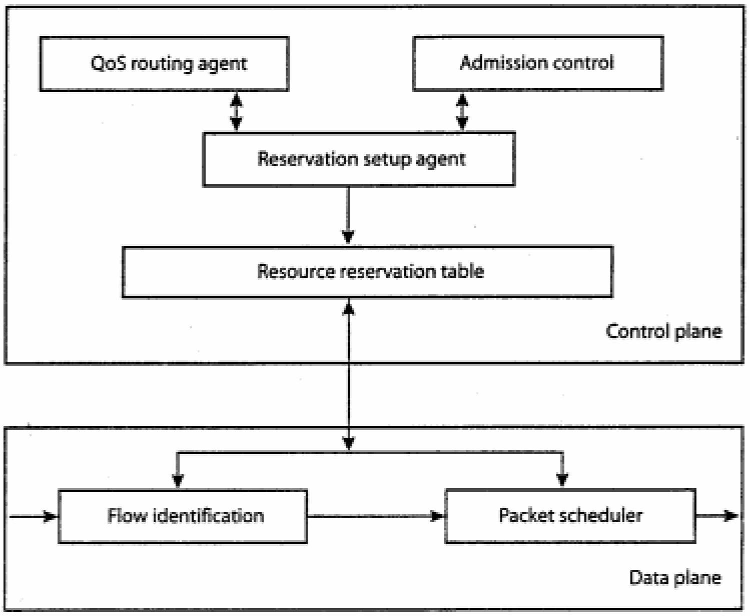


Figure 11 Integrated Services Reference Model (Wang, 2001)

Wang (Wang, 2001) described in details the above model structure. Below, each stage functions are summarized, mentioning its role in the reservation process. We will describe the reservation process briefly, and then each stage will be discussed independently.

Reservation Process

To set up a resource reservation, an application first characterizes its traffic flow by specifying the QoS requirements. These requirements are often called the flow specifications. The reservation setup request can then be sent to the network, and then when a router receives the request, it has to perform two tasks:

- It has to communicate with the QoS routing agent, which uses a special QoS routing algorithm capable of determining the next hop to which the reservation request should be forwarded.

- It has to coordinate with the admission control, to decide whether there are sufficient resources, such as the bandwidth, accordingly, the request can either be set or refused.

Once the reservation setup is established, the information regarding the reserved flow is stored into the resource reservation table. This information is used to configure the flow identification module and the packet-scheduling module in the data plane. So, when new packets arrive, the packets are identified to check that they belong to a reserved flow. These packets are put on the suitable queues, where the packet scheduler allocates the resources to the flows based on the reservation information stored in the reservation table.

1) Flow Specifications and Identifications

Before making a reservation, an application must specify the traffic characteristics that it will demand from the network, and specifies the QoS requirements of these packets. These specifications are usually called flow specifications. These specifications determine the traffic rate, and the requirements that the source will ask the network to provide. The following parameters are some examples of the flow specifications:

- Peak rate, which describes the highest rate at which a source can generate traffic.
- Average rate, which describes the average transmission rate over a time interval.
- Burst size, which is the maximum amount of data that can be injected into the network at the peak rate.
- Delay, the delay requirement can be specified as the average delay or worst-case delay.
- Minimum bandwidth, which describes the minimum amount of bandwidth required by an application flow.

- Delay jitter. a delay-jitter requirement specifies the maximum difference between the largest and smallest delays that packets experience.
- Loss rate, which is the ratio of lost packets and total packets transmitted. Packet losses in the Internet are often caused by congestion.

The parameter ID consists of two numerical fields, one identifies the service associated with the parameter (the <service number>), and the other identifies the parameter itself (the <parameter number>). (Wang, 2001). The textual form is:

<Service number, parameter number> is used to write a service-number/parameter-number pair*. Integrated services support two types of services for real-time flows: guaranteed and controlled service.

Guaranteed Service:

This service provides guaranteed bandwidth and strict bounds on end-to-end queuing delay for real-flows. The service is required by the applications that require the highest assurance on bandwidth and delay. An application invokes guaranteed service by specifying a traffic descriptor (TSpec) and a service specification (RSpec) to the network. (Braden et al., 1994).

Controlled Service Model

In the guaranteed service model, the reservation process must be done according to the worst case bandwidth and delay bounds. For bursty traffic, reservation must be done satisfying the maximum bit-rate that the traffic source might require, which leads to low network utilization and increased cost for resource reservation. Moreover, determining the exact bandwidth and delay requirements for certain application isn't an easy job.

* More information regarding parameter numbers and its value settings can be found on Wang's book, (Wang. 2001) pages 31,32.

For example, consider the playback of a compressed digital movie, the peak rate of the movie could be substantially higher than the average rate, and the burst size is probably hard to quantify without detailed analysis of the traffic.

For some applications which have some burst nature, a service model with less strict guarantees and lower cost would better serve their needs. As a result, The IntServ working group proposed the controlled load service (Braden et al., 1994). The controlled load service does not provide any quantitative guarantees on delay bound or bandwidth. Instead, it tries to emulate a lightly loaded network for applications that request the service. Its characteristics fit well with adaptive applications that require some degree of performance assurance but not absolute and exact bounds.

How Routers Process a Flow

After the real-time flow has been identified and characterized, the router must examine every incoming packet, and decide if the packet belongs to one of the reserved flows. In addition to the above parameters, an IP flow is identified by five fields in the packet header: source IP address, destination IP address, protocol ID, source port, and destination port. The five fields are often referred to as the five-tuple .To determine if a packet matches an RSVP flow, the flow identification engine must compare the five-tuple of the incoming packet with the five-tuple of all flows in the reservation table. If there is a match, the corresponding reservation state is retrieved from the reservation table and the packet is forwarded to the packet scheduler with the reservation state associated with the flow (Wang, 2001). Note that flow identification and processing must be performed on every packet. That may introduce some challenges, especially in high-speed backbones, where hundreds of thousands of packets are processed.

2) Route Selection

At each node, the network must determine which path to use for setting up the resource reservation. The path must be selected so that it is likely to have sufficient resources to meet the application requirements. For example, suppose that an application needs to reserve a path of 50 Mbits/sec bandwidth to a particular destination. It is important that the router selects a path that have residual bandwidth, which is equal or above 50Mbps.This routing selection mechanism is done using a QoS routing algorithm, which can accommodate QoS requirements when selecting the suitable path.

3) Reservation Setup

To set up a reservation, we need a reservation setup protocol that goes hop by hop along the path to install the reservation state in the routers. (Zhang et al., 1993). The protocol also carries the information about traffic characterization and resource requirements so that at each node along the path, it can determine whether the new reservation request can be accepted or not. In IntServ, the RSVP protocol has been developed as the reservation setup protocol for the Internet (Zhang et al., 1993). More details about the RSVP protocol will be discussed later.

4) Admission Control

In order to offer guaranteed resources for the reserved flows; a network must monitor its resource usage and state. It should deny reservation requests when no sufficient resources are available. An admission control agent performs this task as a part of the reservation process. Before a reservation request is accepted, it has to pass the admission control test.

Admission control determines whether a new reservation can be set up or not by investigating and monitoring the status of the available resources.

5) Packet Scheduling

The last step of resource reservation, and probably the most important one, is packet scheduling. The packet scheduler is responsible for enforcing resources allocation. It directly affects the delay that packets will experience. The main task of a packet scheduler is to select a packet to transmit when the outgoing link is ready.

In fact, the current queuing discipline which applies the role of First-Come-First-Served (FCFS), and is designed for the best-effort traffic, can not support resource guarantees (Stallings, 2002). More advanced scheduling algorithms are necessary to support the IntServ model. The Class-Based Queuing mechanism (CBQ) is one of those scheduling algorithms, that can be used with the IntServ architecture (Floyd, Jacobson, 1995).

Class-Based Queuing Mechanism

The CBQ mechanism is based on the notion of controlled link sharing. Where the user's traffic is organized into a tree or hierarchy of classes. A class can be an individual flow or an aggregate of flows representing different applications. Below is an example of simplified CBQ, supported by the QRS simulator (Zhang et al., 2000), used in our simulation. As seen in Figure 12, three kinds of flows are assigned to different workload types. They are class A, class B, and class C. The priority decreases from class A to class C. Class A is the highest priority for control and singling traffic, i.e. RSVP traffic and route traffic. Class B is for real-time traffic workloads, and class C is the lowest priority for best-effort services, i.e., FTP, Telnet and HTTP workload. Traffic with higher priority will be served before traffic with a lower priority in accordance with the CBQ scheduling algorithm.

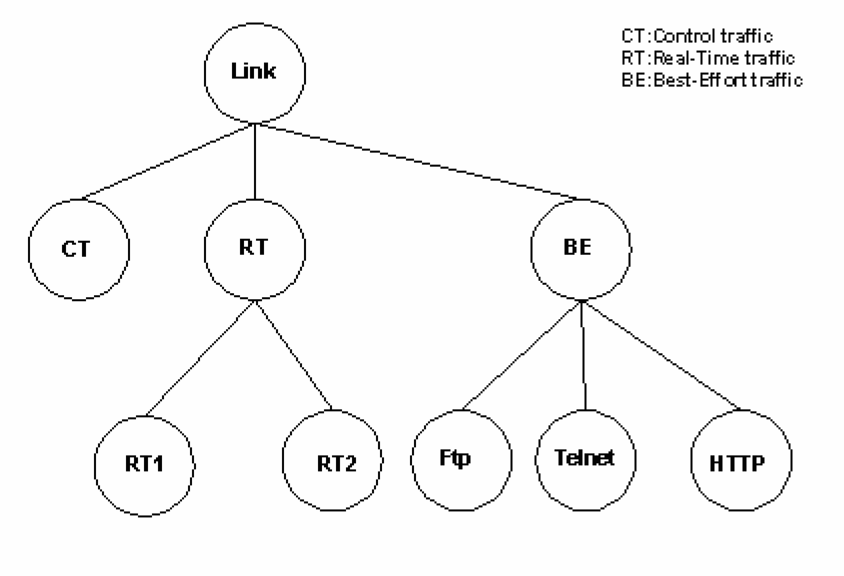


Figure 12 Traffic Classes and Types

Each traffic class is then assigned certain priority value. At the same time a separate queue is maintained for each traffic class to ensure individual traffic class service requirements are satisfied, as seen in Figure 13. In our simulation, we have selected CBQ; because it is one of the most promising traffic scheduling algorithms used in future networks (Floyd, Jacobson 1995).

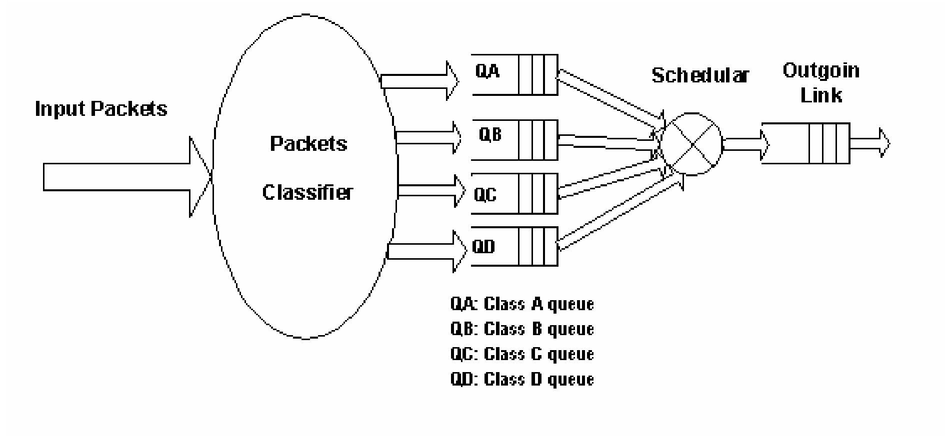


Figure 13 Traffic Classes Queues

RSVP Protocol

Resource Reservation Setup (RSVP)

With the best-effort model, an application can send packets whenever it wants. However, the IntServ architecture requires an application to set up a reservation before it can transmit traffic (Wang, 2001). This requires the existence of a new protocol for setting up resource reservation in the network. RSVP is a resource reservation setup protocol developed by the IETF for this purpose (Zhang et al., 1993). The RSVP protocol is used by hosts and routers in the network to establish a reservation state along a path.

Operation Overview

The RSVP protocol is used to establish a resource reservation between a sender and a receiver. RSVP makes a reservation in only one direction (simplex flow). Although an application may act as both a sender and a receiver, RSVP treats a sender as logically distinct from a receiver. Thus, in a two-way communication, the two ends must establish a reservation for both directions. In RSVP there are two types of messages: PATH messages and RESV messages. In Figure 14, the PATH messages are sent from the source to the destination.

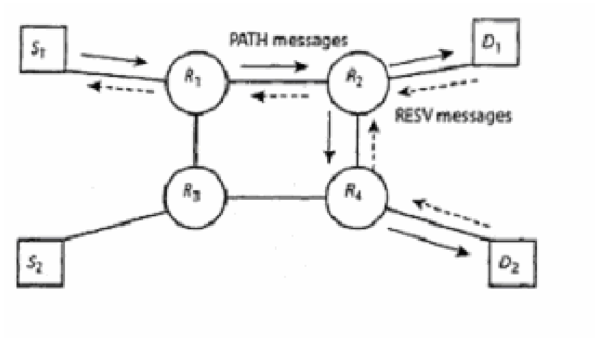


Figure 14 RSVP Messages (Wang, 2001)

These messages serve many purposes:

- First, they distribute information about the traffic source to the receivers.
- Second, the PATH messages are used to pass on characteristics of the path.
- Last, the PATH messages install the necessary state for the RESV messages to find out how to reach the senders from the receivers.

After receiving the PATH messages, receivers can request reservation by sending RESV messages upstream toward the source along the exact reverse path of the PATH messages. The RESV messages specify the resource requirements and set up the state in the routers along the path. After receiving the RESV messages, senders can start to transmit packets along the reserved paths (Wang, 2001).

Reservation Termination

Reservation can be terminated when one of the following situations occur:

- If the sources violate their traffic description (for example, by sending at a higher rate than the agreed-on), the network will obviously not be able to keep its promises, so the reservation is no longer valid.

- The reservation setup protocol must deal with changes in the network topology. For example, if a link goes down, the reservation protocol should set up a new reservation and tear down the old reservation.

3.2 Differentiated Services Architecture

The best-effort model and the IntServ architecture, represent two extremes of the resource allocation mechanism, the best-effort traffic model works on a per-packet basis, so each packet is treated independently, while the IntServ architecture deal with individual flows. The IETF suggested the DiffServ approach, so it is somewhere in between these two extremes; "it takes one small step further from the best-effort model to offer a better than best-effort service" (Wang, 2001).

In DiffServ, traffic is divided into a number of groups, called forwarding classes. All the packets that belong to the same forwarding class are coded with a 6-bit value, inserted in a field in the IP header, called a Differentiated Services Code Point (DSCP). So, all packets with the same code point will receive the same forwarding treatment, in terms of drop priority and bandwidth allocation. In DiffServ network, the nodes at the boundary of the network called (boundary nodes or edge nodes), and nodes inside the network called (interior nodes or core nodes), have different responsibilities, as seen in Figure 15 when traffic arrives at the boundary of the DiffServ domain, the boundary node performs two basic tasks: packet classification and traffic conditioning. They include mapping packets to different forwarding classes, and checking whether the traffic flows meet the service agreements, then packets are forwarded based only on the forwarding classes in the packet header.

DiffServ / IntServ

In DiffServ, only boundary nodes at the edge of the network classify traffic and mark packets. Once the packets are marked, the interior nodes use the forwarding classes encoded in the packet header, to determine the treatment of the packets. While the IntServ require all nodes to perform packet classification, and to support RSVP signaling protocol to identify packets and to know whether they belong to a reserved flow or not. So DiffServ is easier to implement in real-networks and more scalable.

The approach that DiffServ use for resource allocation is done to aggregated traffic rather than individual flows, so resources are allocated to individual classes, while in IntServ, resource reservation is based on per-flow behavior, so IntServ can provide better control on QoS than DiffServ.

In our thesis, we have adopted the IntServ architecture model, since it can provide us with better QoS control and assurance on the real-time flows, which we are interested in studying their performance.

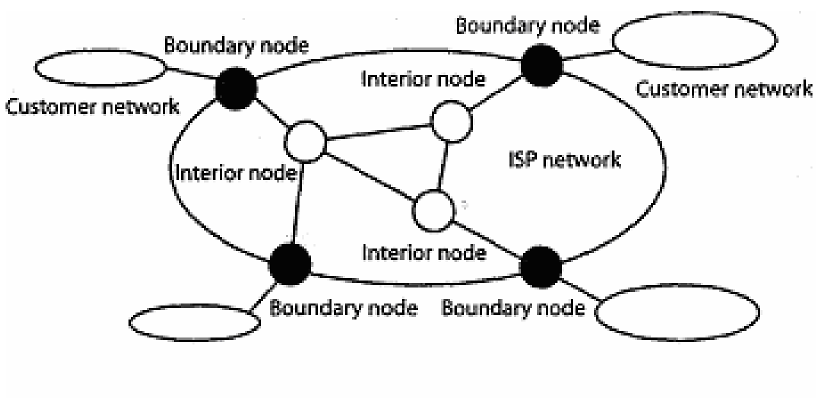


Figure 15 Differentiated Services Domain (Wang, 2001)

4 QoS Routing

As described above, QoS routing is one of the essential parts of both QoS architectures; DiffServ and IntServ. In order to understand the concept of QoS routing, a general overview for the current best-effort routing mechanism and algorithm is presented.

4.1 Routing Process

A computer network consists of transmission links that connect computers together. Usually a network is modeled by a graph, which consists of a finite set of points called nodes or vertex, these nodes are connected together by lines called links; usually these nodes represent routers, and the links correspond to the transmission paths (Cravis, 1981).

A typical example of a graph is shown below in Figure 16. Nodes are depicted as small circles with numbers inside them, and the links are the lines that connect them, Generally, if the set of nodes is *N* and the set of links is *L,* we will refer to the graph as *G (N L).*

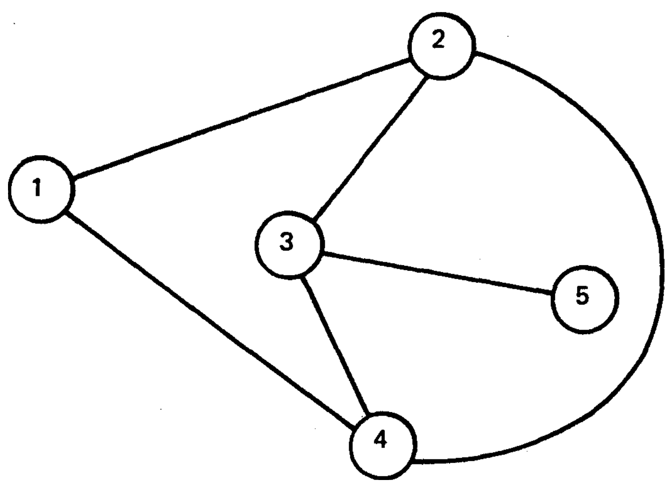


Figure 16 Typical Graph (Cravis, 1981)

Usually in computer networks, there is no difference between link (i, j) and link (j, i), where i and j being node numbers. Another point to consider is that any pair of nodes are connected by at most one link; of course, there may be no link, as in the case of nodes 1 and 5 of Figure 16.The routing is defined as the process of moving information across a network from a source to a destination, in which at least one intermediate node typically is encountered (Cravis, 1981). Routing process involves two basic activities: determining optimal routing paths, and transporting packets through that path (Steenstrup, 1995).

Path Determination

The routing protocol uses a certain algorithm called routing algorithm, in determining the path between two nodes in the network, this algorithm requires certain link parameters which characterizes the link. These parameters are called link metrics, which are the standard for measurement used by the routing algorithm in determining the optimal path between a source and a destination (Steenstrup, 1995). After investigating link metrics, the routing algorithm determines the routing path for the incoming packets and maintains routing information in terms of routing tables, which contain information such as the next hop address.

4.2 Routing Algorithms

Depending on how routers gather information about the structure of the networks, two major routing algorithms exist:

- ❖ Distance Vector (DV) algorithms.
- ❖ Link State (LS) algorithms.

In distance vector also known as Bellman Ford routing algorithm, every router has to know the weight (metric) of the links connected directly to it, and save this information in its routing table, and after a specific interval, it sends the routing table to its neighbors, and receives the routing tables from its neighbors. Based on the information in its neighbor's routing tables, it updates its own routing table. One of the most important distance vector protocol is the Routing Information Protocol (RIP).

In link state algorithms, every router has complete information about all other routers in the network, and the traffic status of the network. Every router advertises this information to all other routers in his network, not just neighboring routers (Stallings, 2002).

Link state algorithms are also known as Dijkstra routing algorithms. One example of link state protocol is the OSPF protocol.

In our thesis, we are interested in studying the QoS extensions added to the OSPF routing protocol, to make it able to support real-time applications. We have chosen the OSPF routing protocol for the following reasons:

- The OSPF protocol is now considered the preferred interior routing protocol for the TCP/IP based Internet (Moy, 1998).
- OSPF depends on Dijkstra algorithm in determining the shortest path, this algorithm is efficient, simple and has low computational complexity, and so it can be enhanced to make it able to support QoS requirements.

Below, we will discuss the Dijkstra algorithm, and see how it can determine the shortest path between a source and a destination node.

Dijkstra's Algorithm

The Dijkstra algorithm (Steenstrup, 1995) works on a directed graph. Let $G=(V, E)$, it finds the shortest paths from the source node r, to all the other nodes. The main idea of the Dijkstra's algorithm is to change the temporary labels associated with nodes into permanent ones. The permanent label of a node denotes the shortest path weight from the source node to the current node. For node i, we denote:

$$A(i) = \{ j := e = (i, j) \in E, j \, has \, temporary \, label \}.$$

At the beginning, a node s is given a permanent label 0, $j \in A(s)$ temporary label $c_{(s,j)}$ and all other nodes a temporary label ∞ . Denote P to be the set containing all the nodes with permanent labels, and $T = V - P$ to be the set containing all the nodes with temporary labels. At each step, the algorithm chooses the node $i \in T$ with the minimum temporary label, and makes it permanent, record its predecessor index, and update the temporary values of all the nodes $j \in A(i)$. This procedure is repeated until all nodes become permanent ones. Figure 17 depicts the Dijkstra pseudocode, used in the link state routing protocols.

$Begin$

$\quad P := \{r\}; T := V - \{r\};$

$\quad d(r) := 0; \; and \; pred(s) := 0;$

$\quad d(j) := c_{sj} \; and \; pred(j) := r \; for \, all \, (s, j) \in A; d(j) = \infty \; for \, other \, nodes$

$\quad while \, P \neq V \, do$

$\qquad Choose \; the \; \min i \in T:$

$\qquad d(i) := \min\{d(j) : j \in T\}$

$\qquad Update \, P, T : P := P \cup \{i\}, T := T - \{i\}$

$\qquad Update \, temporary \, lables:$

$\qquad for \, all \; j \in A(i), compute \, d(j) = \min\{d(j), d(i) + c_{ij}\}, set \; pred(j) := i$

$\quad end \; do$

end

Figure 17 Dijkstra Pseudocode.

4.3 Routing Metrics

As mentioned above, the routing algorithm needs information about the connected links, so it can use this information in routing the incoming packets to its destinations. Currently, the simple Internet traffic, called the best-effort traffic, uses the shortest path first algorithm, which depends on minimizing a certain cost function; this cost function uses one of the following metrics.

- Hop-count metric.
- Delay metric.
- Hop-Normalized metric.

Hop-Count Metric

This metric specifies the number of passes through internetworking nodes, such as routers, that a packet must go through before reaching its destination; this is widely used in the current routing algorithms.

Delay Metric

The delay metric determines the propagation delay, processing, switching and queuing delay that the packets encounter on the outgoing links. For every packet, the router receives and forwards, it measures queuing and processing delay to which it adds transmission and propagation delay. For each of its out-going links, it averages this total delay over a ten-second period and compares the average to the last reported value for the link. If the difference passes a significance criterion, a routing update is generated for distribution to the rest of the network (Khanna, Zinky, 1989).

With the delay metric, routing decisions are based on the actual measured link delay values, which were calculated during a previous interval and propagated via routing updates. Khaanna and Zinky**,** (1989) studied the delay metric in details, and found that the measured packet delay on a link is a good predictor of the link delay encountered after all nodes re-route their traffic based on this reported delay (Khanna, Zinky, 1989). Thus, it is an effective routing mechanism, only if there is some correlation between the reported values and those actually experienced after re-routing. They found that the correlation between successive measured delays is high when a network is lightly loaded. But the predictive value of measured delays declines sharply under heavy traffic loads. So, the delay metric under heavy loads or congestion links doesn't reflect the actual status of the loaded links. Making the delay shortest path algorithm does not select the shortest-delay path ,as it is supposed to do in lightly loaded networks. Moreover, they found that using the delay metric may cause in some cases routing instability and oscillations (Khanna, Zinky, 1989). To solve the above problems, associated with the delay metric, they proposed some modifications to the delay metric, so the modified one, can reflect the actual state of the link under both heavy and light load conditions, so these congestion links are avoided. The modified metric is called the hop-normalized metric. Below we will briefly discuss it, and see how it can reduce the congestion on the network.

Hop-Normalized Metric

The main idea behind this metric is to normalize the link cost in terms of hops. So, when a link reports a cost, the cost is relative to the costs of alternate links, so the reported cost values of the links will reflect the true image for the traffic load conditions and congestion of the whole network links. This way the congested links could be avoided, which will increase the utilization of the network. We will try to extend Khanna results obtained on best-effort traffic, and examine this attractive metric with the real-time traffic, and study the effect of using such a metric with some QoS routing algorithms, and how the performance of these real-time services; such as the VoIP, is affected. For the algorithm code, see Appendix B.

4.4 Unicast / Multicast Routing

Unicast routing is the common routing process in which the router tries to find a path between single transmitter and single receiver, while multicast routing concern with finding paths between single user and multiple users, so packets can be sent from single user to multiple users simultaneously, this is very useful in some applications such as the videoconferencing.

4.5 QoS Routing Definition and Algorithms

Current routing protocols, use the shortest path algorithm that characterizes the network with a single metric, such as the hop-count or delay, and try to find a path that will minimize that metric. However, in order to support QoS requirements, current routing protocols need to consider more than one single metric. So, the routing algorithm will be able to find a path that satisfies multiple constraints (Chen, Nahrstedt, 1998).

In order to study the feasibility and complexity of taking more than one metric in the routing algorithm, we will describe the types of the used metrics.

Routing metrics available follow one of three categories:

1. Additive metrics, in which the cost for the whole path between the source and the destination is equal to the sum of each link cost along that path. Below are some examples of theses additive metrics:

- The end-to-end delay, which is equal to the sum of the propagation, switching and queuing delay encountered by each link along the path.
- The jitter delay.
- The hop-count.

2. Concave metrics, such as the bandwidth, in which the bandwidth of the total path is determined by the maximum or minimum value of any link bandwidth along that path.
3. Multiplicative metrics, such as loss probability.

Wang and Crowcroft (1996) studied the complexity of taking more than one metric into consideration in the routing process and obtained the following results:

- A two additive metrics together will form a NP-complete problem that can't be solved in polynomial form and which is computationally inefficient. To solve such a problem, some heuristic approaches exist, which will take for example two additive parameters into consideration in computing the routing path, and try to solve it in a polynomial form.
- A problem with two constraints is also not feasible in terms of complexity and timing, and lead to a NP- complete problem.
- To get a feasible and efficient QoS routing method, the chosen metrics should be orthogonal to each other, to remove any redundant information between the metrics.

According to the above roles, it is clear that any two or more of delay, delay jitter, cost, loss probability in any combination as metrics are NP-complete. The only feasible combinations are bandwidth and one of the four (delay, loss probability, hop-count and delay jitter). Wang and Crowcroft (1996) developed an algorithm, which is efficient in terms of time complexity, and takes the bandwidth and hop-count into consideration. This algorithm is called the Shortest-Widest Path algorithm (SWP), described below.

Shortest-Widest Path Algorithm

Shortest-widest path (Wang, Crowcroft, 1996), selects the path with the largest available bandwidth. If several paths exist with as large bandwidth, the one with the smallest hop count is selected. This algorithm is considered efficient, since it applies Dijkstra's algorithm twice. First for finding the widest path, that satisfy the bandwidth constraint requested by the real-time application, then after pruning all the links that do not meet the bandwidth constraint, it will select the widest one in terms of the residual bandwidth. If more than one link exists with the same residual bandwidth, the one with the shortest path, in terms of hop-count is chosen.

Another feasible algorithm that has low time complexity, and can be solved using Dijkstra's algorithm is the Widest-Shortest Path (WSP)(Apostolopoulos et al., 1999).

Widest-Shortest Path Algorithm

This algorithm selects the minimum hop-count path among those that satisfy the bandwidth requirements. If there are several paths with the same hop count, the widest, that is the one with most available bandwidth, is selected. In fact, this algorithm was suggested to be used with the QoS OSPF protocol (QOSPF) (Apostolopoulos et al., 1999).

Through our study, we will try to examine several metrics with the widest-shortest path algorithm, so this algorithm will be called Widest-Least Cost (WLC) algorithm, and every time, we assign different metric to it, when the hope metric is used as the cost function (as it is designed in the original algorithm). It is called the widest-shortest path algorithm.

From the above two algorithms, it should be noted that the standard routing algorithms are typically single objective optimizations, i.e. they may minimize the hop-count, or maximize the path bandwidth, but not both. Double objective path optimization is a more complex task, and in general, it is an intractable problem (Chen, Nahrstedt, 1998).

Simulation Models and Settings

1 Preview

After introducing the definition of VoIP and studying the key elements, which determine its quality, through simulation, the performance of this service is investigated, and then the effective metric that should be used with the QOSPF routing algorithm is determined. In this section, a general discussion of the topologies and scenarios used in simulation is described.

2 Simulation Environment and Parameters

Simulation Tool

To study the performance of computer networks running certain applications, network simulators are widely used among researchers and analysts .One of these simulators is the Quality of Service Routing Simulator (QRS) (Zhang, Kantola, 2002). We have implemented our simulation using the QRS router simulator, running under Linux redhat operating system, on Pentium II, 400 MHz computers. The reasons behind choosing this simulator are:

- The simulator is designed specially to study the QoS routing mechanisms, so many facilities and algorithms are supported and can be customized according to the user's requirements (Zhang, Kantola, 2002).
- The results obtained by this simulator are accurate and precise (Zhang et al., 2000), so many researchers adopt it and use it in their papers and research (Ma, 2000), (Zhansong et al., 2001).

Below is a general description of the simulator written to present how this simulator works, and what are the required settings and configurations that must take place.

QoS Routing Simulator (QRS)

QRS is developed on the core of Maryland Routing Simulator (MaRS). (A laettinoglu, 1994).The development work is being carried out at networking laboratory, Helsinki University of Technology (HUT). These developments include adding a number of QoS related components and functions such as: simplified RSVP signaling protocol, Resource Management (RM), QOSPF routing protocol. Real-time flows are among the new modifications that take place. Moreover, a number of route computation algorithms and link-state update methods are implemented in QOSPF. QRS can be used for studying QoS routing and investigating the performance of real-time traffic flows within an intra-domain IP networks. (Zhang et al. , 2000).

Using QRS

In QRS , networks with different topologies are modeled using specific configuration files, these configuration files describe the network components ,such as the nodes and links, and how these components are connected together, and what are the traffic loads used in the simulation .Each network component has a certain number of parameters which determines the characteristics of each element ,for example ,the bandwidth is one of the parameters of the link component ,which determines the available bandwidth on that link. All of these parameters can be configured by the user according to his requirements. Moreover, the parameters required to be monitored are specified, so when the simulator runs these files, the required results can be obtained using certain log files.

Network debugging is also available in forms of a debugging log file, which gives the user information about the simulation stages and parameters status. More details about the simulator, configuration files, and logging files can be found in Appendix A.

3 Simulation Scenarios

To produce meaningful results; the simulation environment must be configured carefully; this simulation set-up is called a 'simulation scenario'.

Generally, a simulation scenario consists of two different main components:

- Network topology and connections.
- Traffic flows and sources.

Network Topology

One basic issue in network simulation is what topology to use for the network being simulated. Unfortunately, the topology of the Internet is difficult to characterize (Floyd**,** Paxson, 2001); because of the fact that the Internet structure is constantly changing, and there is no typical topology which might resemble the random behavior of the Internet's loads and traffic models. In fact, simulations are sensitive to topological structure, and the obtained results may be affected with the variation of the selected topology (Floyd**,** Paxson, 2001) .So, to help researchers in choosing appropriate and realistic topology, the research community has made significant advances in developing topology-generators for the Internet; which can create networks with locality and hierarchy based on the structure of the current Internet. Moreover, they recommend some common topologies, which have characteristics similar to the Internet, or at least can be located in some important Intranet and backbones (Floyd**,** Paxson, 2001). In our Thesis, simple topologies are used first to investigate and understand the operation of the QRS simulator and how the QOSPF is capable of reserving resources for real-time applications, after that, more complex topologies are used; such as the NSFNET topology.

3.1 Traffic Flows

There are many types of traffic in the Internet: simple and real-time traffic loads.

- Simple Internet traffic or sometimes-called Best-Effort traffic (BE) is the largest source of traffic in the Internet (Held, 2000). HTTP, FTP and Telnet are some examples of this type of traffic. In fact, this traffic is the basic traffic for which the Internet was built.
- Real-Time traffic (RT), this type of traffic consists mainly of voice and video traffic, these are the new added traffic loads to the current Internet that require special treatments and have QoS demands (Wang, 2001). Below is a quick description of the nature of this type of traffic and its QoS demands. In the QRS simulator, traffic flows are generated in traffic sources connected to source nodes and they sink in traffic destinations connected to destination nodes. Both types of sources (real and non-real) time traffic are supported. Many parameters of these workloads can be adjusted such as: transmitting rate, which is determined by setting different inter-departure times between consecutive packets. Also the period of duration of traffic production and pauses can be set, which is useful in simulating the VoIP traffic loads, which has the talk-silence nature (Fiorini, 2000). Moreover, both packet size and starting time of traffic flows can be determined.

3.2 Real-Time Traffic Characteristics

1) Voice over IP Traffic:

In evaluating the performance of VoIP traffic, it is important to have an accurate and realistic model, which has characteristics similar to the real life VoIP traffic. So, to have an accurate modeling of voice traffic, it is important to understand the speech process between two persons. It has been found that the speech model can take many states such as: talk-spurt, pause, doubletalk, mutual silence, alternative silence, interruption, speech after interruption, and speech before interruption (Fiorini, 2000). In fact, these events or "states" can be placed in a discrete Markov Chain and transition probabilities assigned. In Figure 18, a six state Markov Chain is demonstrated this model is called the "Brady Model and it is one approach for modeling VoIP traffic (Fiorini, 2000).

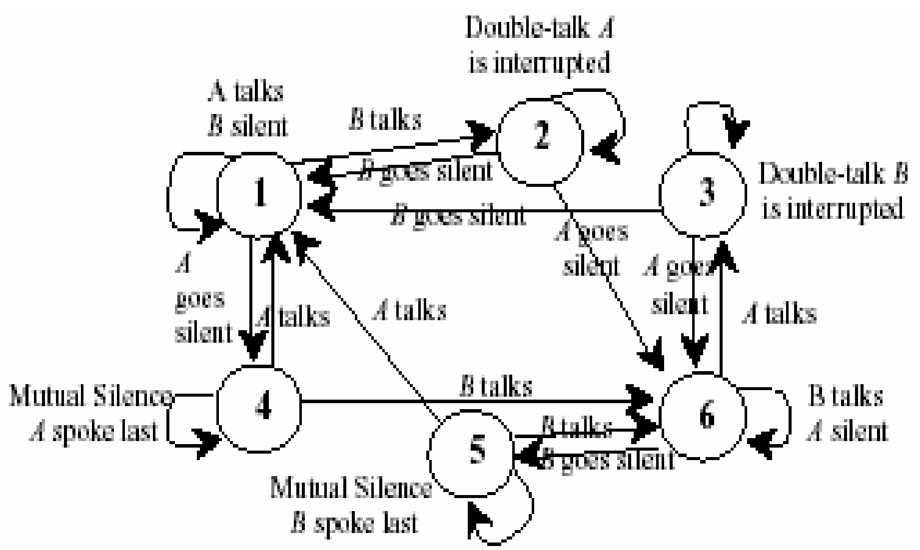


Figure 18 The Brady Markov Model with two Speakers A and B (Fiorini, 2000).

In fact, we can think of voice traffic using a "two-state" process. In other words, some user A alternates between periods of "talk-spurts" also called "on period " and "silence periods" or "off periods ". Figure 19 illustrates this type of model. This model has immediate applications since studies of telephone users have demonstrated that the average talk-spurt is exponentially distributed and lasts between 0.4-1.2 sec followed by an exponentially distributed silence period of 0.6-1.8 sec in length (Fiorini, 2000). More specific studies indicate that the talk-spurt lasts approximately 352 ms; and, the average silence period lasts around 650 ms (Fiorini, 2000).

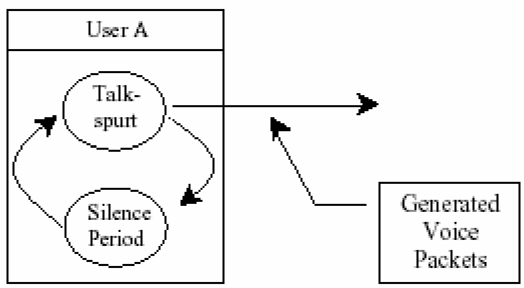


Figure 19 A simple two-State Traffic Model (Fiorini, 2000).

According to the above discussion, we have followed the same voice model used in the telephony network, since VoIP must be similar in nature to the traditional telephony system, so it can be considered as an attractive alternative to the typical telephony systems (Held, 2000). So, we have randomly chosen average talk-spurt periods that is exponentially distributed and lasts between 0.4-1.2 sec followed by an exponentially distributed silence period of 0.6-1.8 sec in length .In Table 2, some examples for VoIP traffic sources are mentioned.

Table 2 VoIP Traffic Models

Traffic ID	Encoding Rate	Start After	Talk	Silence	Talk	Silence	Talk	Silence	Talk	Silence
			VoIP source 1		**VoIP source 2**		**VoIP source 3**		**VoIP source 4**	
RTH1	64 Kbps	15 sec	0.40	0.60	0.60	0.74	1.1	1.2	1.20	1.80
RTH2	64 Kbps	17 sec	0.44	0.74	0.60	0.82	1.0	0.90	1.10	1.25
RTH3	64 Kbps	21 sec	0.58	0.63	0.90	0.77	1.09	0.84	1.16	1.78
RTH4	64 Kbps	24 sec	0.60	1.04	0.82	1.15	0.91	1.41	1.13	1.72
RTH5	8 Kbps	19 sec	0.43	0.62	0.52	0.75	0.86	1.25	1.16	1.38
RTH6	8 Kbps	26 sec	0.45	0.61	0.56	0.94	0.68	1.32	0.92	1.46
RTH7	6.4 Kbps	18 sec	0.41	0.63	0.57	0.85	0.64	1.51	0.78	1.59
RTH8	6.4 Kbps	24 sec	0.46	0.89	0.56	1.56	0.76	1.66	1.05	1.73

Another important point is deciding when each VoIP session starts? QSR simulator gives us the capability of choosing the starting time of transmission for each real-time traffic, and in order to be more realistic and reasonable, we have randomly chosen various starting times, which has an exponential distribution, since the arrival time for packets is known to be either exponential or poison distribution (Stallings, 2002).

VoIP Bandwidth Requirements

In the simulation, the bandwidth of VoIP traffic is selected according to the different rates standardized by the International Telecommunication Union (ITU) (Hassan et al., 2000). Table 3 shows the ITU recommendations, for different compression techniques used in VoIP, associated with the required bit rate.

Table 3 ITU Recommandations (Hassan et al., 2000)

No.	Coding Standard	Compression Algorithm	Bit Rate Kbps
1	G.711	PCM	64
2	G.729	CS-ACElP	8
3	G.723.1	ACELP	5.3/6.4

Video Real-Time Traffic

The other type of real-time traffic is the video traffic. This type of traffic has many forms and shapes; such as video conferencing, and video on demand. This type of traffic doesn't have the behavior of on-off periods described in the VoIP traffic model. Usually this traffic tends to be continuous in transmission with high bandwidth requirements.

Bandwidth Requirements

Bandwidth requirements for network multimedia applications can range anywhere from 100 Kbps to 70 or 100 Mbps. Figure 20 shows the amount of bandwidth that the various types of network multimedia applications require (Cisco, 2002).

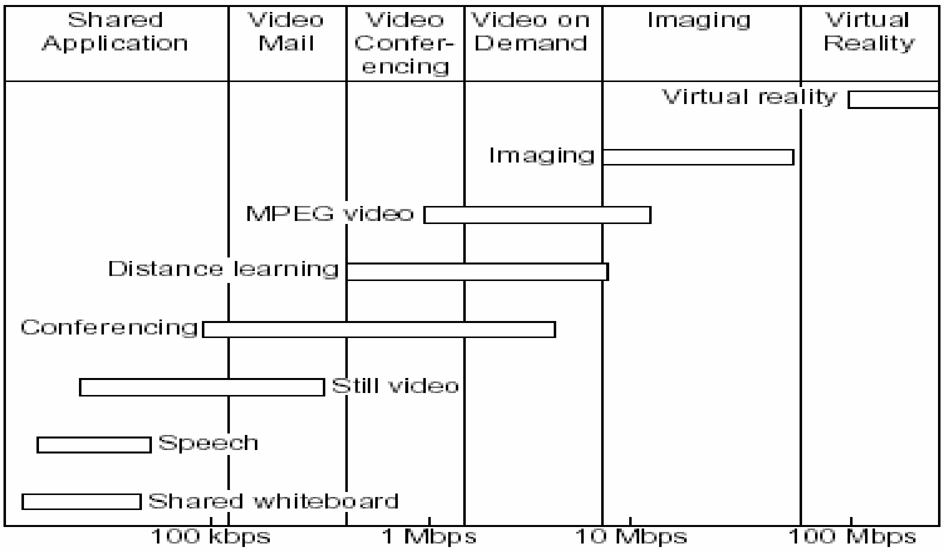


Figure 20 Video Bandwidth Requirements (Cisco, 2002).

Streaming Traffic

Streaming refers to an application that generates a constant stream of data to send either audio or video information across the IP network (Ramkishor, Mammen, 2002). According to that definition, VoIP with silence period equal to zero is considered as audio streaming source, regarding the video traffic, usually it is continuous streaming traffic.

4 Performance Measures

In our simulation, three main performance criteria are used:

- Throughput
- End-to-end delay
- Jitter

Throughput is defined as the amount of data transferred from one place to another or processed in a specified amount of time, throughputs are measured in Kbps, Mbps and Gbps. Usually throughput is calculated by determining the amount of data in bytes or bits received in one second. In our simulation, the length of load packet is 512 bytes and the header is 32 bytes long. So throughput is calculated taking into account that the size of packet is 544 bytes. We record the number of received packets at the destination and then multiply it by the packet size then dividing the total number by the reception time.

End-to-end delay is defined as the time duration a packet travels from source to destination nodes (Hassan et al., 2000). The unit of delay in is second or millisecond (ms). End-to-end delay is calculated as follows:

End-end-delay = (packet's receiving time)-(packet's sending time). So both sending and receiving time are logged for each packet. This delay is composed of propagation delay, transmission delay, switching and processing delay and queuing delay. We assume that the propagation delay is constant and each link is assigned a value that represents the propagation delay and it is set to 1ms.

In addition, Transmission delay = packet size/ link bandwidth, this value is also constant and predefined since both link bandwidth and packet size are constants defined by the user. Finally, queuing delay occurs inside the network, due to the fact that routers generally need to store packets for some time before forwarding them on an outbound link. Queuing delay varies with the length of queues in buffers. Therefore, the only variable factor is queuing delay.

Neither throughput nor end-to-end delay are enough to judge the performance of the real-time application, in other words, even though packets can have low packet loss and low end -to-end delay, it may suffer from variation in the delay at the receiver's side or what is called jitter. This important parameter is calculated from the received time of each packet. The receiving time of a packet is subtracted from the receiving time of the previous one, this value represents the variation on the delay that the receiver encounters when receiving consecutive packets. Obviously, to have perceptible voice service, it is necessary that this variation in delay is kept as low as possible and it mustn't alternate widely. Table 4 shows the delay /loss sensitivity for different Internet traffic (Curado, 2001).

Table 4 Delay/Loss Sensitivity (Curado, 2001).

Class	Delay Sensitivity	Loss sensitivity	Application
1	High	Low	Video real time
2	High	Medium	VoIP
3	Low	Low	Best-effort

5 First Scenario: Simple Topology

Objectives

To be familiar with the simulation environment, a simple topology is chosen, thus basic concepts can be tested and verified. Moreover the routing is simple and can be predicted. Throughout this scenario, we tried to study the concept of link utilization, congestion, and how real-time applications get affected if no QoS mechanism is implemented.

Network Topology

This topology simply consists of two nodes that act as routers, connected together with a communication link .The bandwidth of the link can vary according to each case studied. As seen in Figure 21, this topology will be used in cases 1,2 and 3 of this scenario.

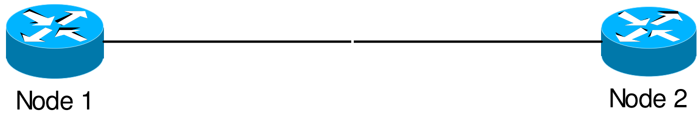


Figure 21 Simple Topology

Case 1

The link bandwidth is set to 3 Mbps; two components are attached to the network. One is called a simple-traffic-source component. This component generates simple (non real-time) traffic such as the common Hyper Text Transfer Protocol traffic, at a value pre-defined by the user. This value is set to 4 Mbps. Note that this number is chosen to be higher than the link bandwidth (3 Mbps), so congestion and drop packets will occur.

The other component is called simple-traffic-sink, which is attached to the sink (destination) node, Figure 22.

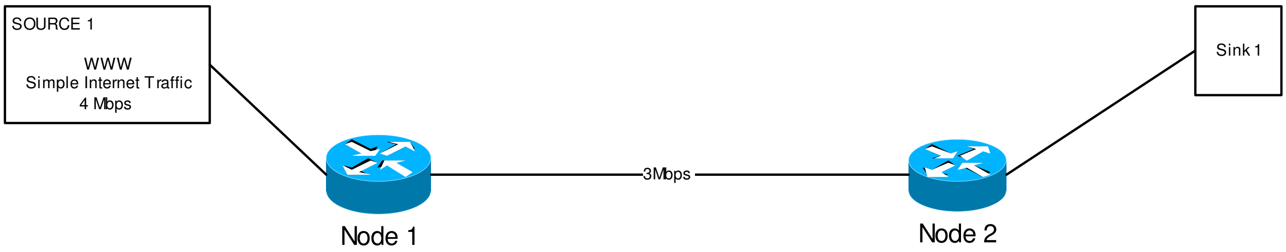


Figure 22 Scenario 1,Case 1

Case 2

A new component is attached to node 1, called real-time-source1; this one will generate real-time traffic, which will ask for bandwidth reservation and will interact with both the RSVP and the RM components which are also connected to both source and destination nodes, Figure 23. The value of traffic production and pause are set to zero, the transmission rate is set to 1 Mbps. Note that this value is chosen so that the required bit rate, which is 1 Mbps, is lower than the link bandwidth, so reservation for this real-time source can occur. To measure the end-to end delay encountered by each packet, both transmission and receiving time is monitored for each packet.

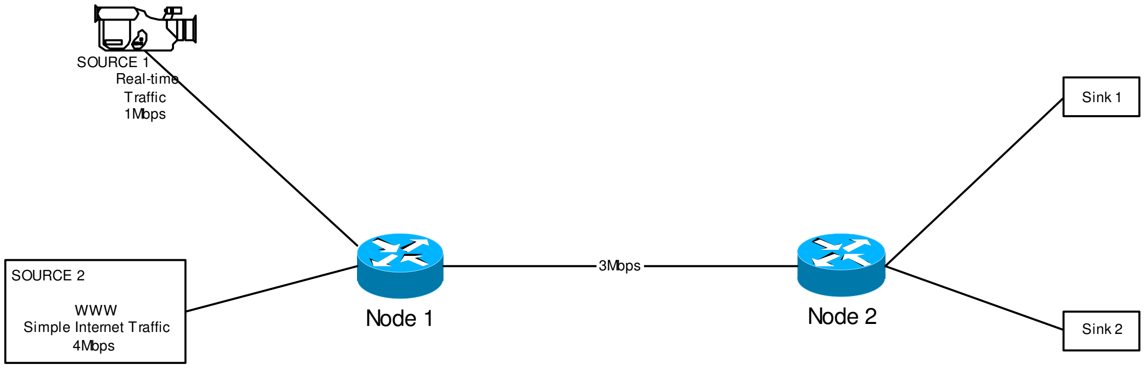


Figure 23 Scenario 1,Case 2

Case 3

The same scenario of case 2 is repeated, but this time without having any QoS mechanism or any QoS routing algorithm, in other words, the real-time traffic is treated similar to the non real-time traffic, which uses the traditional SPF routing algorithm, Figure 24.

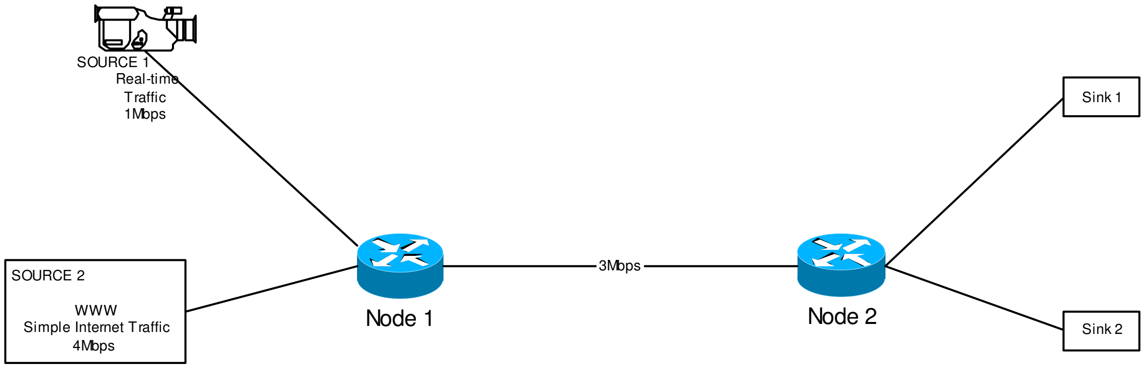


Figure 24 Scenario 1,Case 3

6 Second Scenario: Mesh Topology

Objectives

After investigating simple topology in scenario one, more advanced topology is used here in this scenario; this topology has more routing alternatives to be considered. More cases are studied and discussed such as the network utilization, resource reservation process, and finally, two QoS routing algorithms ; Widest-Least Cost algorithm and Shortest-Widest Path algorithm,are used in the simulation.

Network Topology

A simple 2*2 mesh topology is used in this scenario, as shown in Figure 25. This topology consists of 4 nodes, each node represents router device, connected together with communication links; the bandwidth of each link is set in a different way in each case.

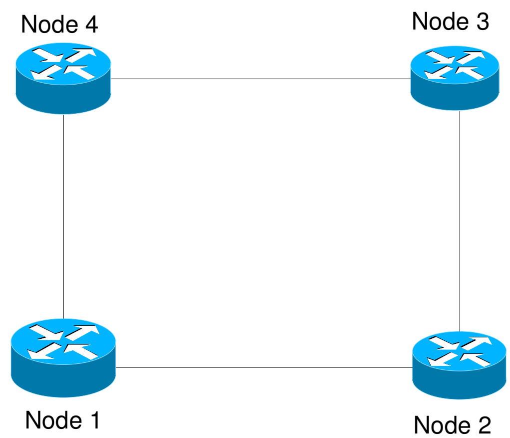


Figure 25 Mesh Topology

The objective behind studying this scenario is trying to gain more understanding of the process of bandwidth reservation and performance of real-time sources. Moreover the advantage of QoS routing which tries to avoid congestion links, and obtain better network utilization is investigated. Finally, the problem of best effort traffic in generating congested links, and the bandwidth reservation process for the real-time flows is again addressed and discussed.

Case 1

The link bandwidth of the above topology is set in a way such that the available bandwidth along any path between node 1 and node 3 is lower than the required bandwidth requested by the real-time flows. A real-time source is attached to node 1, with its sink component at node 3, the bit rate of this real-time source is set to 1 Mbps, while the available links bandwidth is set to 0.5 Mbps, as shown in Figure 26.

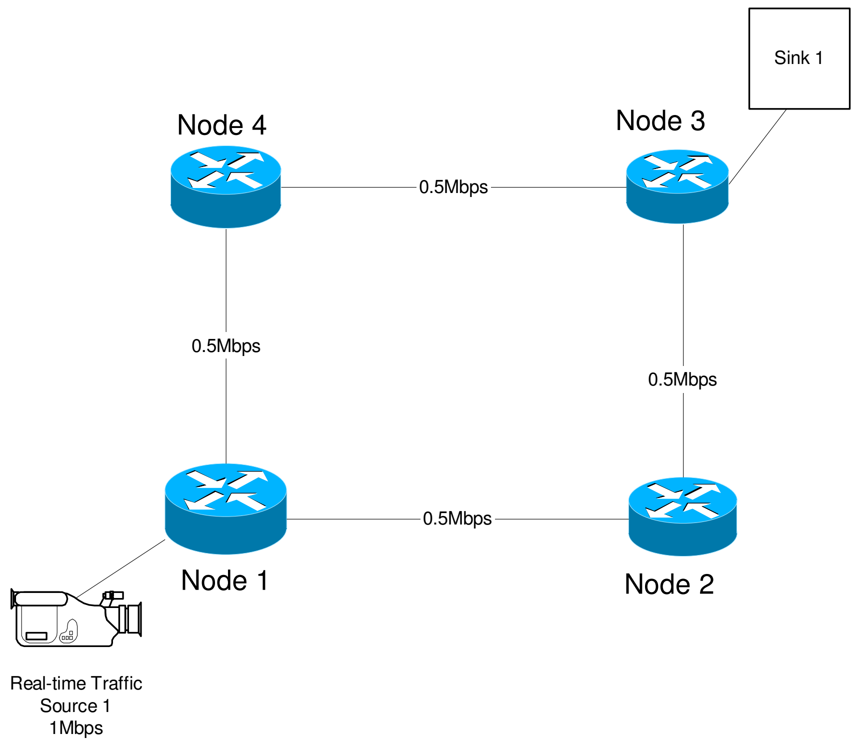


Figure 26 Scenario 2,Case 1

Case 2

The same topology and link bandwidth is used as in case 1, but with minor modifications. The link between node 1 and node 2 is increased to 1.5 Mbps, so the link bandwidth, connecting node 1 with node 2, is higher than the requested bandwidth by the real-time flow, which is 1 Mbps, see Figure 27.

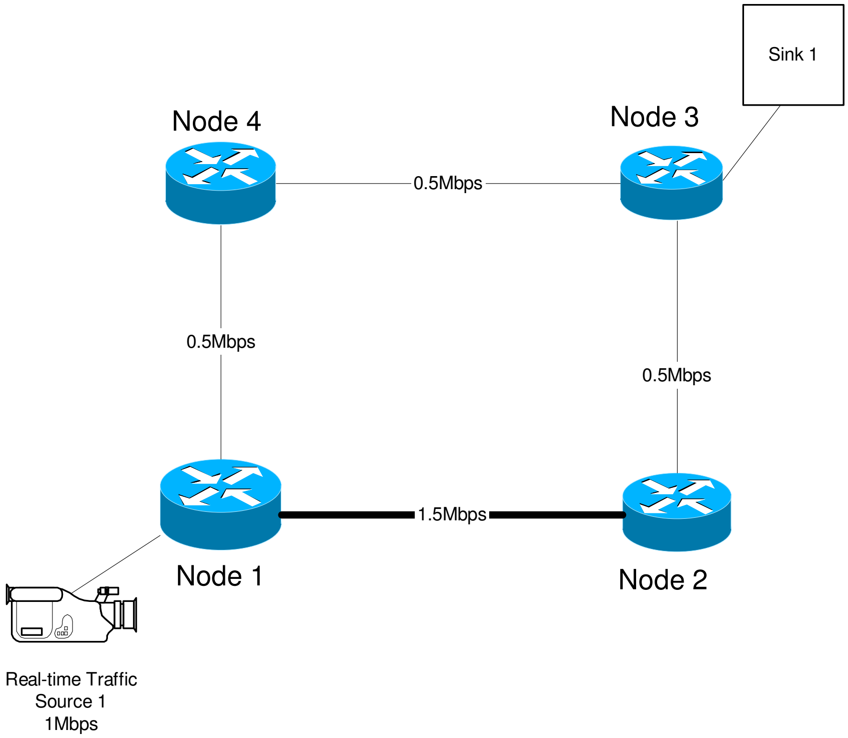


Figure 27 Scenario 2,Case 2

Case 3

The same topology of the previous case is used, but this time the link bandwidth connecting node 2 and node 3 is also increased to 1.5 Mbps, so by this modification, there will be a path between node 1 and node 3, which satisfies the bandwidth requirements of the real-time traffic source, Figure 28.

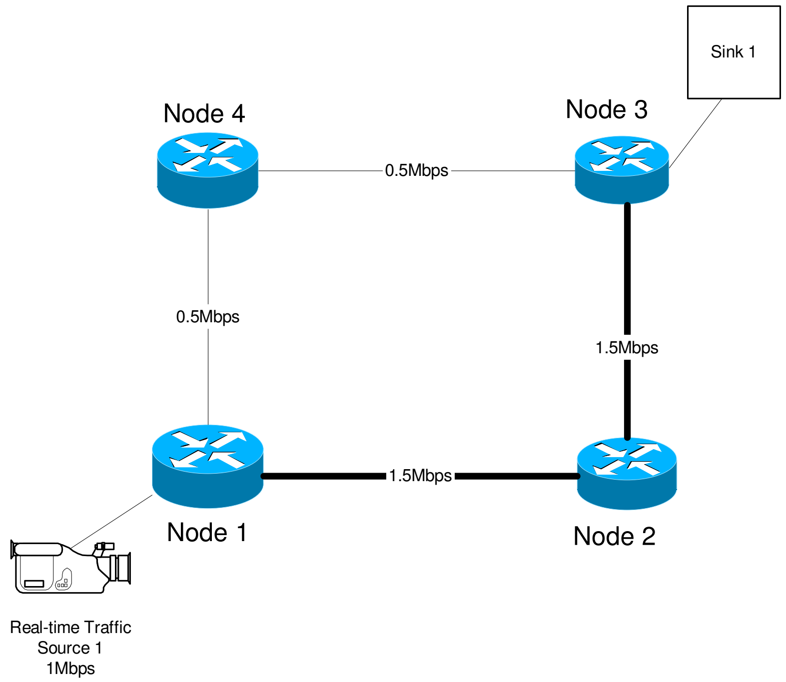


Figure 28 Scenario 2,Case 3

Case 4

In this case the bandwidth of each link is set to 6 Mbps, two real-time traffic sources with bit rate of 4 Mbps each, are attached to node 1, while their sink components are attached at node 3. Note that the link bandwidth is capable of carrying only one real-time traffic, since the bandwidth required for transferring two real-time traffic is equal to 8 Mbps, while the link bandwidth is 6 Mbps, Figure 29.

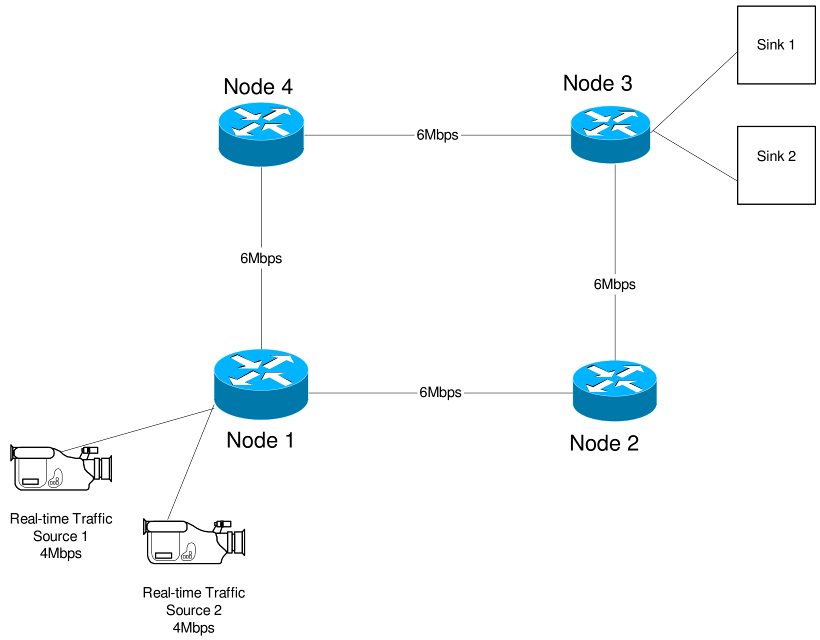


Figure 29 Scenario 2,Case 4

Case 5

The same settings used in case 4 are repeated here, but with two simple traffic sources instead of the real-time sources, with the same bit rate, as shown in Figure 30.

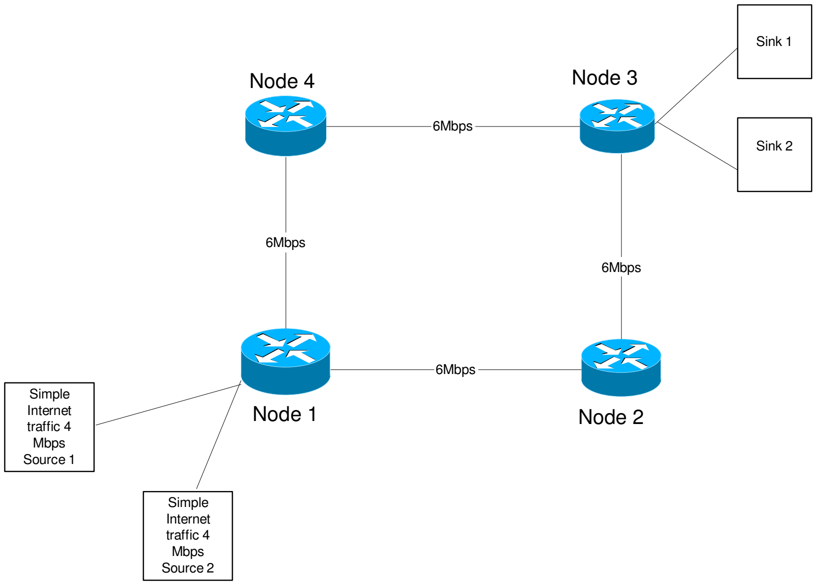


Figure 30 Scenario 2,Case 5

The purpose behind studying case 4 and 5, is trying to see how a real-time traffic source which uses QoS routing strategy and mechanisms such as the IntServ architecture; is capable of utilizing the network resources, better than the best-effort traffic which uses the SPF algorithm

Case 6

In this case, a simple modification to the mesh topology; described at the beginning of scenario two, is introduced by connecting node 1 and node 3 directly, as shown in Figure 31 The link bandwidth of this link is set to 3 Mbps, while the other links bandwidth are set to 6 Mbps. Two simple traffic sources are connected to node 1, with a bit rate of 4 Mbps each, while the sink components for both sources are connected to node 3. The bit rate of each source is chosen such that the total bandwidth required to transfer the two sources, which is 8 Mbps, is higher than the available bandwidth link which is 6 Mbps. The connected link between node 1 and node 3 is introduced as the shortest path in terms of hop count between node 1 and node 3, and the link bandwidth is chosen lower than the others to show how the best-effort traffic will always try to seek the shortest links even if it has lower bandwidth or higher traffic than the other links.

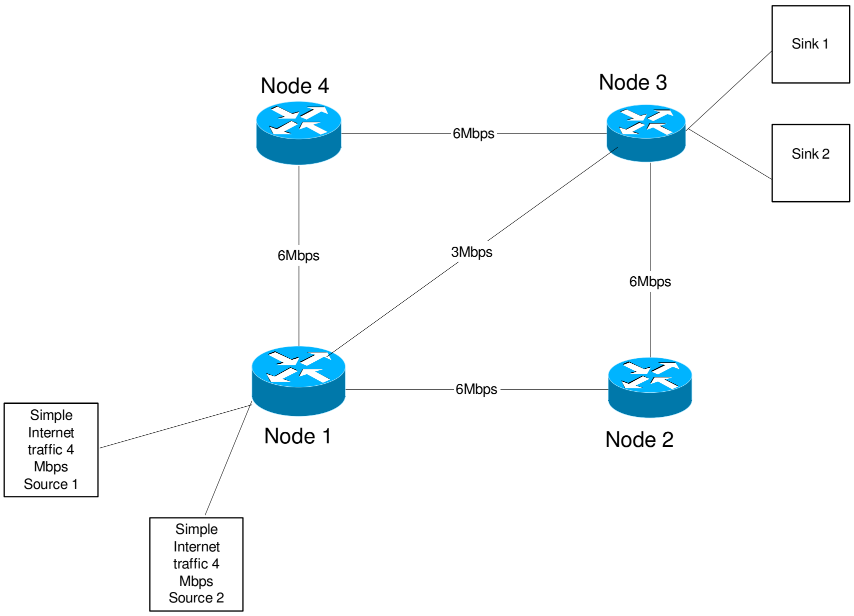


Figure 31 Scenario 2,Case 6

Case 7

The same network architecture is used, but a new real-time source of 1 Mbps is connected to node 1, its sink component is connected at node 3, the bit rate of the real-time traffic source, is chosen such that any available path is capable of transporting this real-time traffic source, i.e. the available link bandwidth is higher than the real-time requested bandwidth. The sink component of this real-time source is connected to node 3.The routing mechanism used for the real-time traffic is configured to use the Widest-Least Cost algorithm, taking the hop-count as its cost function.

Case 8

The same scenario described in case 7 is repeated, Figure 31, but using the Shortest-Widest path algorithm as the QoS routing mechanism used for the real-time traffic source. The reason behind studying case 7 and case 8 is to compare the two QoS routing mechanisms;

the Widest-Least Cost algorithm, and the Shortest-Widest algorithm, and to see how they use the bandwidth metric in their routing algorithm.

7 Third Scenario: NSFNET Topology

Objectives

To gain more insights, more complex topology that has many alternative routes and paths between each node, is examined. One of these topologies is the NSFNET topology, widely used in network simulation (Zhansong et al., 2001). Throughout this scenario, the two QoS routing algorithms discussed in QoS section; the Widest-Least Cost algorithm, and the Shortest-Widest Path algorithm are studied. However, Different cost functions and metrics are used with the Widest-Least Cost algorithm. The performance of the real-time traffic such as VoIP and video traffic is studied, trying to suggest the most appropriate QoS routing algorithm, with the most effective metric, that should be used with the QOSPF routing protocol. To achieve that, several real-time traffic loads are examined, and evaluated using the performance measuring parameters, such as the throughput, end-to-end delay and the jitter.

NSFNET Network Topology

A high speed hierarchical "network of networks" in the US, funded by the National Science Foundation (NSFNET). At the highest level, it is a backbone network comprising 16 nodes connected to a 45Mb/s facility, which spans the continental United States. Attached to that are mid-level networks and attached to the midlevels are campus and local networks. NSFNET also has connections out of the US to Canada, Mexico, Europe, and the Pacific Rim, Figure 32.

In this topology, many alternative routes exist between the source and destination, and in order to make a path more attractive than others, some internal paths are set to have higher bandwidth than others, so all the links bandwidth are set to 1.554 Mbps, which is the same as T1 standard, while some of the internal paths; such as link 13-9, link 9-10, link 10-11, link 11-12, link 12-4, link 12-5, link 12-8 are set to have 3 Mbps. After studying the two QoS routing algorithms, we found the bandwidth metric is crucial in both algorithms, so we have chosen some paths in the network, and allocate higher bandwidth for them than the other paths. This was done to provide the network with some paths, which have higher bandwidth, so it can be used to avoid congestion, we would like to see how the two QoS routing algorithms utilize these paths, and what is the effect of using them to the performance of real-time traffic. However, more clarification is provided when analyzing the results.

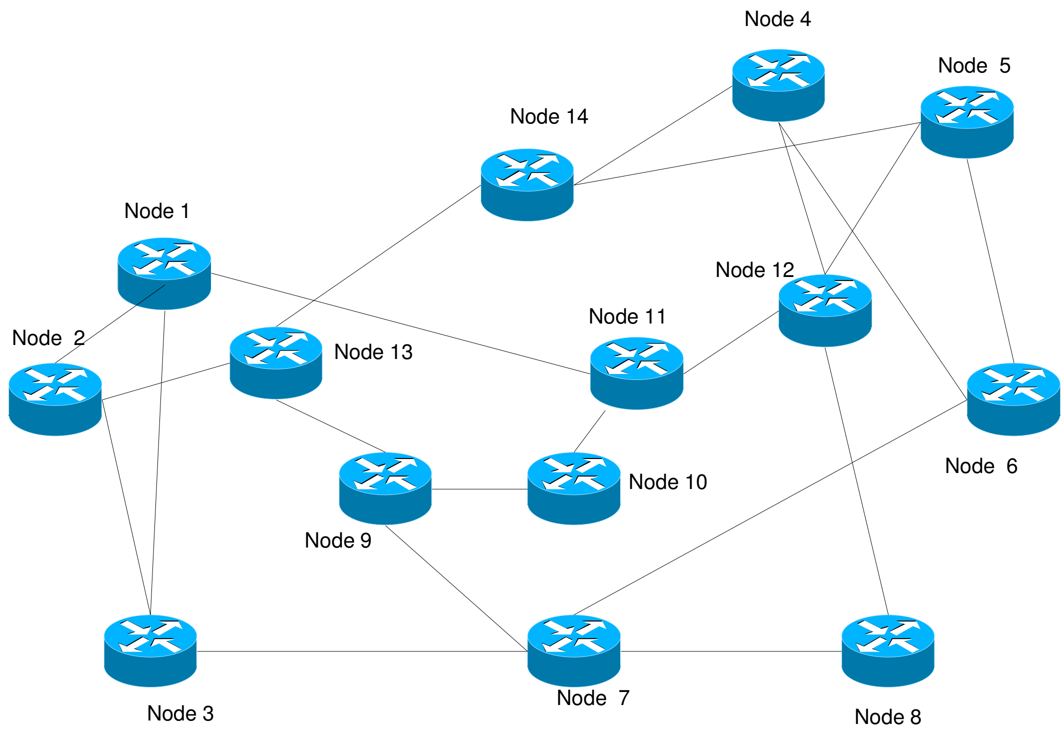


Figure 32 NSFNET Network Topology

Workloads and Traffic Sources

The traffic loads used in the simulation throughout this scenario, are shown in Table 5.

Table 5 Scenario Three Traffic Loads

NO.	Traffic Type	Source	Destination	Bit Rate (Kbps)	Talk Period (sec)	Silence Period (sec)	Arrival Time (ms)
1	VoIP	2	5	64	0.6	0.58	0.2
2	VoIP	2	5	64	0.9	0.77	0.15
3	VoIP	2	5	64	0.89	0.56	0.1
4	VoIP	2	5	64	0.57	0.85	0.5
5	VoIP	2	5	64	0.82	0.99	7.5
6	VoIP	2	4	64	0	0	0.8
7	Video Traffic	2	5	512	0	0	0
8	HTTP Traffic	2	4	2048	0	0	0

9	HTTP Traffic	1	6	512	0	0	0
10	HTTP Traffic	7	5	512	0	0	0
11	Ftp Traffic	1	5	200	0	0	0

Below, full details and elaborations behind studying the above workloads are discussed .

VoIP workloads

Since we are interested in studying voice over IP traffic, several VoIP workloads are used, the source/destination nodes are chosen such that there are many alternative routes between them, as seen in Figure 32, node 2 and node 5 and 6 are located in the edges of the network, which allow for many alternative routes and paths between theses nodes. According to the ITU recommendation (Hassan et al., 2000), among the recommended bit rates, 64 kbps bit rate is chosen, since this bit-rate is the highest one, which is supposed to give us highest quality. Moreover, VoIP sources demand higher bandwidth than the other two recommended bit-rates. Silence/ talk periods are chosen according to the VoIP traffic model, discussed above in section 3.2, as well as the arrival time for each source.

Video Traffic

The real-time traffic includes voice traffic as well as video traffic, so to be more realistic in the simulation, one video traffic load is chosen, between node 2 and node 5, these nodes are chosen again, since they are located at the edge of the network, so many paths and routes exist between these two nodes. The bit-rate and arrival time are chosen according to the specifications of the video traffic source discussed in section 3.2. If Figure 20 is examined, it shows different types of video applications with their recommended bit-rates, the bit-rate used to simulate video traffic is 0.5 Mbps, which successfully simulates video conferencing or distance learning application, which are two valuable and important applications, that might widely be used in the future (Baldi, Ofek,1998).

HTTP Traffic Sources

One of the most important aspects that should be studied when deploying real-time applications is the performance of the best-effort traffic (Chen, Nahrstedt, 1998). The objective is trying to have maximum performance of real-time traffic with minimum impact on the best-effort traffic. To study this aspect, three HTTP traffic sources are studied, the source/destination source for them are chosen such that they will cover many routes and occupy many links in the network that are used as a background traffic . The bit-rate of source 8 is chosen higher than the bandwidth of T1 links which might be a part of its path, between node 2 and node 4,so in this case congestion can occur in that link, so its effect on the real-time traffic loads can be studied.

FTP Traffic

To make our study comprehensive, it includes most of the Internet traffic sources and types, FTP traffic source is introduced between node 1 and 5 as part of the background traffic used in the simulation.

Several runs of the simulation took place, using different routing algorithms with different metrics. In all simulations, the following parameters are monitored and logged:

- VoIP traffic transmission time at the source node and the transmission packet sequence number.
- VoIP traffic receiving time at the receiving node, and the received packet sequence number.

- Video traffic transmission time at the source node and the transmitted packet sequence number.
- Video traffic transmission time at the source node and the transmitted packet sequence number.
- HTTP traffic receiving time at the destination node, with the received packet sequence number.

Note, that the HTTP traffic transmission time isn't monitored; so end-to-end delay isn't studied for this type of traffic, due to the fact that HTTP traffic isn't very sensitive to end-to-end delay. The following QoS routing algorithms are used:

- ❖ Widest-Least Cost algorithm.
- ❖ Shortest-Widest Path algorithm.

Three types of cost metrics are studied with the Widest-Least Cost algorithm; delay, hop-count, and hop-normalized metric, the reason behind choosing these metrics are mentioned in QoS section . The Widest-Shortest Path algorithm is used with minor modifications that allow us to study different cost metrics rather than the hop-count metric used in the original algorithm, so this algorithm will be called the Widest-Least Cost algorithm. In studying the performance of VoIP and Video traffic sources; the following performance measures are used:

- ❖ Throughput at the receiving end.
- ❖ End-to-end delay.
- ❖ Delay variation called Jitter.

In addition, the Mean Opinion Score, is used to give us an idea about the quality acceptance of the real-time services. The study cases are summarized in the Table 6.

Table 6 Scenario, Cases Study

Case No.	Description	Justification
Case 1	The real-time flows will use the traditional SPF algorithm.	This case will show the importance of having QoS algorithm and QoS framework, to make the existence of real-time applications possible and efficient.
Case 2	The real-time flows will use Widest-Least Cost routing algorithm, using the hop-count as its cost metric.	To study the performance of real-time traffic using the hop as its routing metric with the Widest-Least Cost algorithm
Case 3	The real-time flows will use Widest-Least Cost routing algorithm, using the delay as its cost metric	To study the performance of real-time traffic using the delay as its routing metric with the Widest-Least Cost algorithm
Case 4	The real-time flows will use Widest-Least Cost routing algorithm, using the hop-normalized as its cost metric	To study the performance of real-time traffic using the hop normalized metric
Case 5	The real-time flows will use Shortest-Widest Path algorithm	To study the performance of real-time traffic using the Shortest-Widest path algorithm

It is important to note that the above routing algorithms are used with real-time flows, while the traditional shortest path algorithm is used with the best-effort traffic. During our work, we have studied the performance of all of the available traffic loads mentioned in Table 5. The throughput, end-to-end delay and jitter are measured for them, and for convenience, we have chosen some of these sources to show the results obtained in details, thus the results for three real-time sources are mentioned fully, i.e. their results are depicted in terms of performance graphs. We have chosen one of the five VoIP traffic sources, that is transmitted from node 2 to node 5, so we have chosen VoIP source number 1, Also VoIP source number 6, which is transmitted for node 2 to node 4, is chosen. Finally the video source running between node 2 and node 5 is also chosen, which is called source number 7, according to Table 5.

The Widest-Least Cost Algorithm

This algorithm checks the available links connected to the node, to ensure that these links have higher bandwidth than the requested bandwidth by the real-time application. So it prunes the links that don't meet this bandwidth constraint, and then apply Dijkstra algorithm in order to find the path with minimum cost function. We have extended Apostolopoulos work (Apostolopoulos et al., 1999). In trying to investigate more cost functions rather than the hop-count metric used in his algorithm. The hop-count, the delay and the bandwidth are the three metrics recommended to be used in designing the QoS routing algorithm for the QOSPF routing protocol (Apostolopoulos et al., 1999), since the bandwidth is embedded in the algorithm itself, we have used the delay and hop-count metrics as the cost function required to be minimized. Moreover, the hop-normalized metric developed by Khanna and Zinky (Khanna., Zinky, 1989) is also examined.

Case 2: Widest-Least Cost Algorithm Using the Hop-Count Metric

The hop-metric was the suggested metric to be used with the Widest-Least Cost algorithm, so it is called Widest-Shortest Path algorithm (Apostolopoulos et al., 1999) . (We will examine the same Widest-Shortest algorithm described in previously, but we will use more metric rather than the hop-count, so we will call the algorithm the Widest-Least Cost, and each time we use it, we specify the used metric) . The idea behind choosing the hop-metric is trying to pick a path with the minimum number of hops among these paths that can support the required bandwidth, when more than one link exists with the same hop-count, the path with highest bandwidth is selected. The interpretation behind using the hop-count with the cost function is trying to consume a minimal amount of network resources. (Apostolopoulos et al. , 1999).

We have investigated this metric, and as it is noticed that the real-time flows select the shortest path between node 2, and node 5,which is the shortest path between these two nodes in terms of hop-count. The algorithm checks the bandwidth of the connected links to node 1, i.e. link 2-13, link 2-1 and link 2-3. All of these links have higher bandwidth than the required bandwidth requested by the real-time flows, link 2-13 has a bandwidth of 3 Mbps, and both link 2-1 and link 2-3 has a bandwidth of 1.5 Mbps. While the requested bandwidth by the real-time flows is 64Kbps for each VoIP sources, and 0.5 Mbps for the video source. So none of these links is pruned in the bandwidth checking mechanism, after that Dijkstra algorithm is applied ,taking the hop-count as the minimizing cost function, so the next hop ,which has the minimum cost is chosen ,and then the algorithm is repeated again on the new node. Thus, the real-time source will be routed to the shortest path between node 2 and node 5, so it is obvious that both the real and non real traffic loads will share almost the same path, since the real-time source between node 2 and node 5 will follow the shortest path between these nodes, which is link 2-13, link 13-14, then link 14-5, and the real and non real-time traffic , between node 2 and node 4 will follow the shortest path between these nodes which is :link 2-13,link 13-14, then link 14-4. So it is obvious that the best-effort traffic source No. 8 path, is the same as the path of the real-time traffic source No. 6, while the real-time sources between node 2 and node 5, share the best-effort source No.8 with some links .As mentioned in scenario 2, the priority for the real-time traffic sources is higher than the best-effort traffic. That results in reducing the amount of available bandwidth for the best-effort traffic.

Throughput Analysis

According to the above discussion, it is expected that the best-effort traffic load between node 2 and node 5 will suffer, as seen in Figure 61, throughput of the best-effort traffic source No.8 is about 0.75 Mbps, which is equal to 50 % of the available link bandwidth, while the other 0.75 Mbps is reserved by the real-time flows. Notice that the bandwidth of the shortest path between node 2 and node 5 is 1.5 Mbps, which is equal to the minimum bandwidth link between the two nodes. Figure 58 shows the bandwidth of the VoIP traffic source No.1, notice that this source is capable of reserving most of its bandwidth requirements, the average throughput of this source is calculated in Table 7, which is equal to 61.43 Kbps, which is very near to the requested bandwidth, Figure 59 shows the throughput of the VoIP source No. 6, the average throughput of this source is 56.9 Kbps. It should be noted that this source may get lower quality than the others, due to the reduction in the throughput .In fact even though the reservation process always tries to reserve the bandwidth for the real-time sources, packet loss is still possible, especially in congested links . Figure 60 shows the throughput for the video traffic source No. 7, the average throughput is 456 Kbps, which is very near to the requested bandwidth, but still better performance and throughput can be obtained , if the congested links are avoided. Both best-effort traffic sources No.9 and 10 are capable of achieving their bandwidth requirements, which is about 0.5 Mbps, as seen in Figures 63, and 64. This is because no other traffic source may share the traffic along their shortest path, which is link 1-3,link 3-7,link 7-6 for source No.9, and link 7-6, link 6-5 for source No.10. To evaluate the performance of the real-time sources, the MOS described in VoIP section, for each source is calculated in Table 7.

Table 7 Throughput Evaluation, Case 2

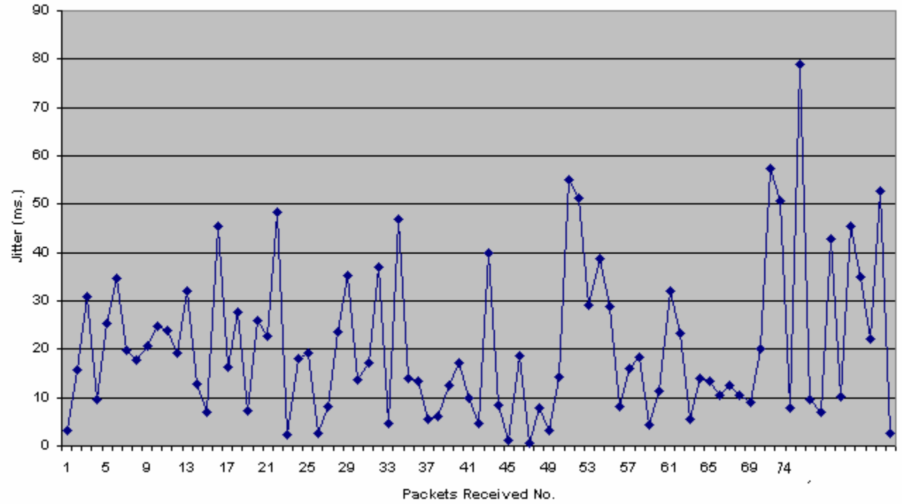
Source No.	Requested Bandwidth	Obtained Bandwidth	Packet lost%	MOS (According to Figure 8)	Performance
1	64 Kbps	61.43	4%	4.5 out of 5	Excellent
6	64 Kbps	56.9	11%	4 out of 5	Good
7	512 Kbps	457.5	10.6%	4 out of 5	Good

End-to-End Delay Analysis

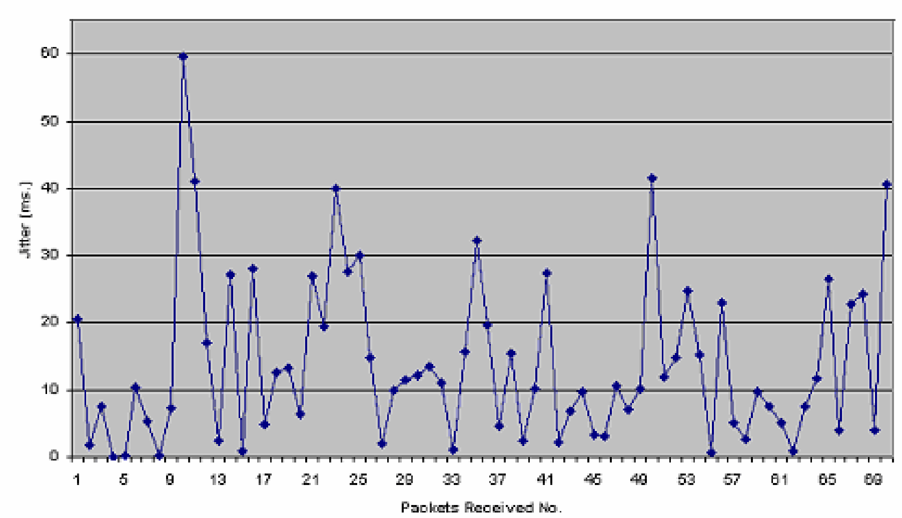
Regarding the end-to-end delay; as it is expected, the real-time flows will suffer from high queuing delay due to the fact that they are routed to the congested path links, connecting node 2 and node 5, notice that even though some links have higher bandwidth, and lower traffic load, even so, the real-time traffic loads ignored these links , which has lower traffic, so lower queuing delay , and follow the shortest path between node 2 and node 5. Notice that in this case, as seen in Figure 65, the delay may exceed the recommended value suggested by the ITU , the average delay for the VoIP source No.1 is about 156.3 ms, and for the other VoIP source No. 6, is about 157.7 ms, while the average delay for the video traffic is around 157 ms. By investigating the delay figures, one can notice that some packets may face large end-to-end delay, for example, Figure 65, some packets face 250 ms end-to-end delay, which is too high and may degrade the voice quality, such packets are better to be discarded, this is done at the receiver's buffer.

Jitter Analysis

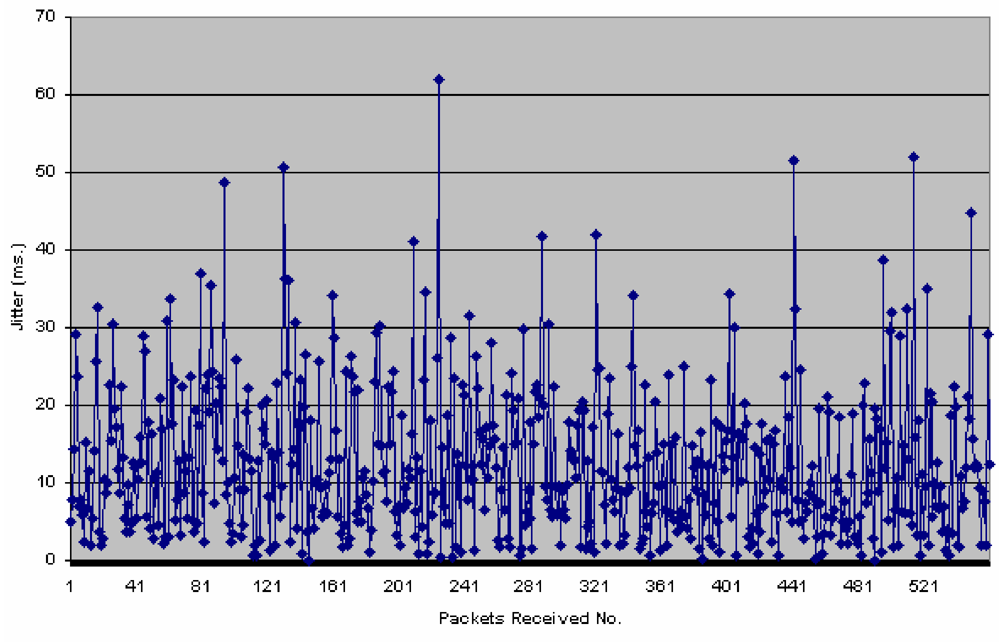
The results of each case are drawn independently, since it is difficult to conclude results from the jitter Figures 68, 69, and 70 shown above. Below Figures 71, 72, 73 show the jitter delay for the same real-time traffic sources, which we are interested in studying theses results obtained considering the hop-count as the routing metric with the Widest-Least Cost algorithm.



Figures 71 Jitter Delay for VoIP Traffic Source No.1



Figures 72 Jitter Delay for VoIP Traffic Source No.6



Figures 73 Jitter Delay for Video Traffic Source No.7

The variation on the delay for all of the real-time flows are shown above, is relatively high, this also expected, since congested links suffer usually from high variation in the delay ,which degrades the performance and leads to a non-uniform arrival for the real-time packets. Moreover, since the end-to-end delay itself is above the acceptable value (150 ms), so de-jittering technique, at the receiver's side may not work efficiently.

Case 3: Widest-Least Cost Algorithm using the Delay Metric

As mentioned in QoS section, the delay metric is used as a cost function, needed to be minimized. With the delay metric, routing decisions are based on the actual measured link delay values, which were calculated during a previous interval and propagated via routing updates. The underlying assumption here is that the measured packet delay on a link is a good predictor of the link delay encountered after all nodes reroute their traffic based on this reported delay. Thus, it is an effective routing metric only if there is some correlation between the reported values and those actually experienced after re-routing. In fact the correlation between successive measured delays is high when a network is lightly loaded. But the predictive value of measured delays declines sharply under heavy traffic loads. So the delay metric under heavy loads or congestion links doesn't reflect the actual status of the loaded links. These facts are verified in our simulation, as we described later, the traffic load for the best-effort traffic running between node 2 and node 4, is higher than the path bandwidth (1.554 Mbps), so it is expected to have some congestion and significant queuing delay on that link. Unfortunately the reported value for the link delay metric doesn't reflect that fact, which verifies that the correlation between the reported values and the actual link state of the links, which constitutes the shortest path between node 2 and node 4, is low under heavy traffic loads and congestion conditions. So this metric will not always inform the routing algorithm with the congested links to be avoided, when it makes its routing decision.

Throughput Analysis

The above facts regarding the inefficiency of the delay metric under congestion, is verified if we look at the throughput of best-effort traffic source No. 8, the throughput again declines sharply to a value similar to the case in which the hop- count is used; i.e. case 1 which is about 0.7 Mbps on average. That proves that most of the real-time packets follow the same shortest path used by the best-effort traffic source No.8, or at least these real-time flows share some links of the shortest path link between node 2 and node 4.

VoIP source No. 1 throughput is equal on average to 57.45 Kbps, which give us an acceptable quality, regarding VoIP source No.6, the average throughput is 54.8 Mbps, and finally the average throughput for the video source is equal to 466.8 Kbps on average. To evaluate the performance of these sources, the MOS for each source is calculated in Table 8.

Table 8 Throughput Evaluation, Case 3

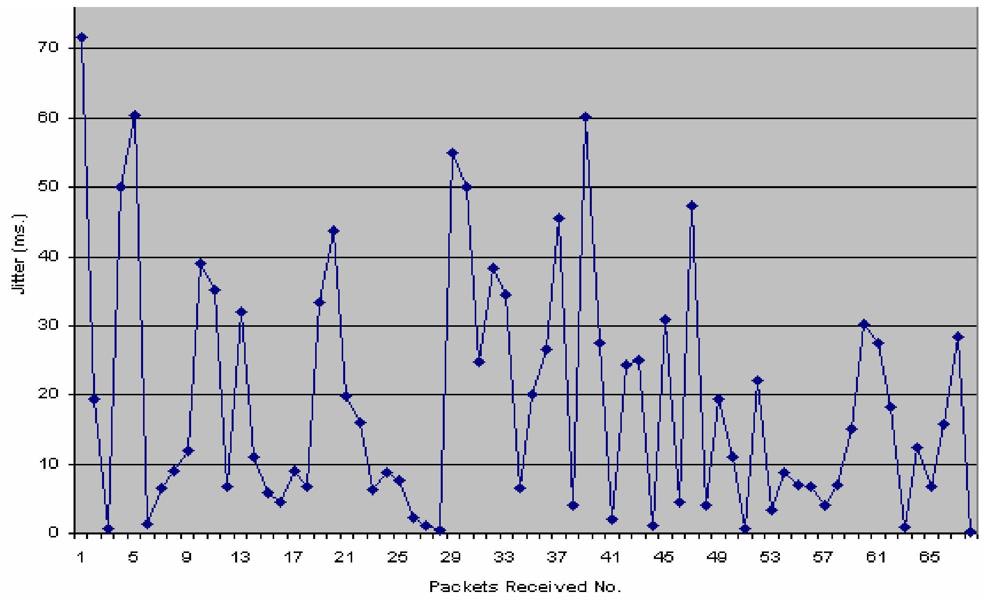
Source No.	**Requested Bandwidth**	**Obtained Bandwidth**	**Packet lost%**	**MOS (According to Figure 8)**	**Performance**
1	64 Kbps	57.45	10.2 %	4 out of 5	Good
6	64 Kbps	54.79	14 %	3.5 out of 5	Good to Fair
7	512 Kbps	466.8	8.8 %	4.2 out of 5	Excellent to Good

End-to-End Delay Analysis

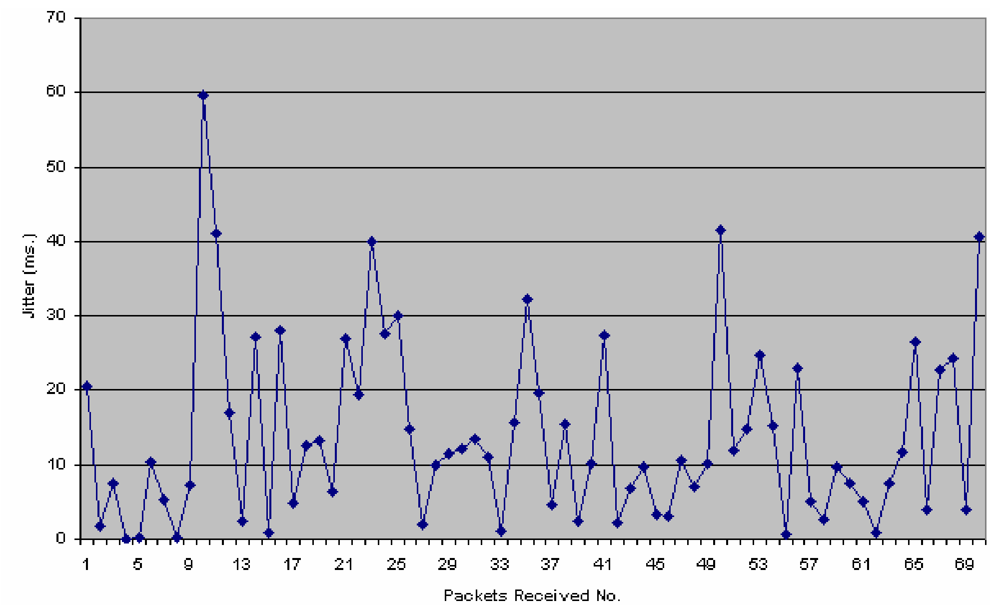
Due to congestion, high end-to-end delay is expected, according to Figures 65, 66 and 67, it is obvious that the average delay for the three real-time source is above the accepted value, the average delay for VoIP source No. 1 is 160.58 ms, VoIP source No. 6 is 156.8 ms, and finally the video source delay is 160 ms, all of theses values above 150 ms.

Jitter Analysis

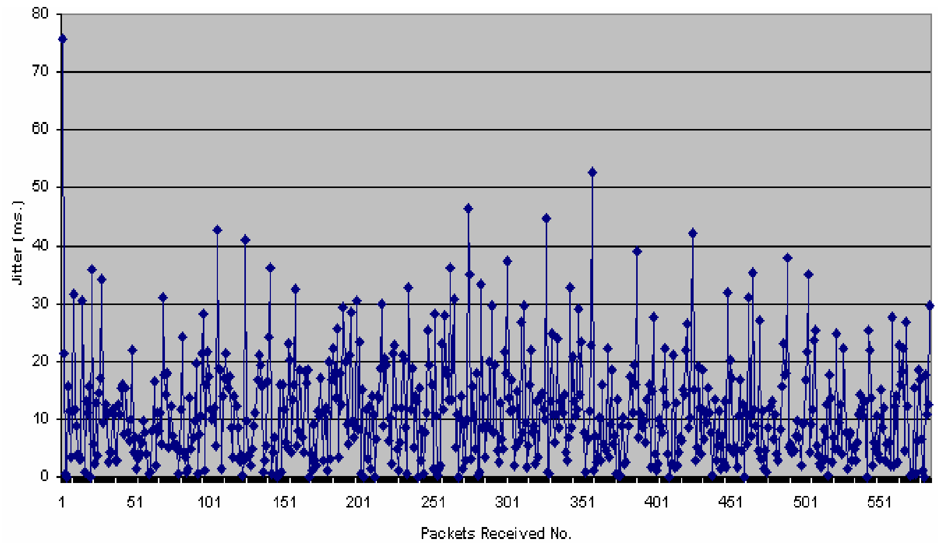
To study the jitter, the results of each case are drawn independently, to be able to conclude results. Below, Figures 74, 75, 76 show the jitter delay for the same real-time traffic sources, which we are interested in. Theses results obtained considering the delay as the routing metric with the Widest-Least Cost algorithm.



Figures 74 Jitter Delay for VoIP Traffic Source No.1



Figures 75 Jitter Delay for VoIP Traffic Source No.6



Figures 76 Jitter Delay for Video Traffic Source No.7

As seen from the figures above, VoIP sources as well as the video source, encounter high variation in the delay, due to the congestion, which exists in the shortest path between node 2 and node 4. Again, it is difficult to use the de-jittering technique used at the receiver's side, since the total end-to-end delay is relatively high and above the recommended value set by the ITU, which is 150 ms.

Case 4: Widest-Least Cost Algorithm Using Hop-Normalized Metric

To solve the problems associated with using the delay in routing decisions, Khanna (Khanna , Zinky, 1989) proposed some modifications to the delay cost, so the new metric called Hop-Normalized metric, described in QoS section, which is designed to reflect the actual state for the link traffic, which helps the routing algorithm in selecting the path with minimum congestion even if this path is longer in terms of hop-count , In fact, this metric sheds the traffic flows from the congested links, so in our case, the real-time traffic may be routed, for example to the link 2-13 ,link 13-9,link 9-10,link 10-11,link 11-12 and finally to either link 12-5 to reach node 5 or to link 12-4 to reach node 4.These routes are longer in terms of hop-count ,but have higher bandwidth , another suggested route which real-time flows may select is ,link2-3,link 3-7 ,link 7-6 and finally link 6-5 ,this flow may be suitable to the real-time sources which destination node is node 5. In fact it is difficult to judge how the routing occur and what are the different routes selected by each real-time flows, but we can notice that there are many alternative routes, which may be different in terms of hop-count, and links bandwidth. And this algorithm is capable of affording the real-time traffic with minimum delay and packet lost.

Throughput Analysis

The simulation results are very consistent, with the operation of the algorithm using this metric, the throughput for the best-effort traffic source No.8 increases significantly, to reach to 1.4 Mbps, which is equal to 90.3 % utilization for its link, which indicates that most of the real-time flows avoided the links that is used by the best-effort source No. 8. And seek for another link that doesn't suffer from congestion. Consequently, the best-effort traffic throughput increases significantly.

Regarding the throughput of real-time sources, Table 9 shows average value for the obtained bandwidth and the performance evaluation for each source.

Table 9 Throughput Evaluation, Case 4

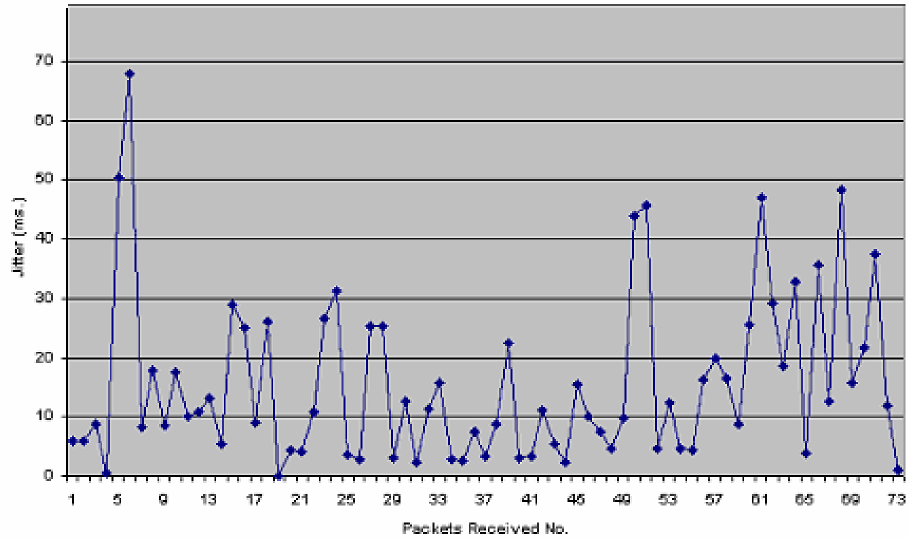
Source No.	**Requested Bandwidth**	**Obtained Bandwidth**	**Packet lost%**	**MOS (According to Figure 8)**	**Performance**
1	64 Kbps	59.2	7.5 %	4.25 out of 5	Good to Excellent
6	64 Kbps	57.8	9.3 %	4.1 out of 5	Good
7	512	457.5	10.6 %	3.9 out of 5	Good

End-to-End Delay Analysis

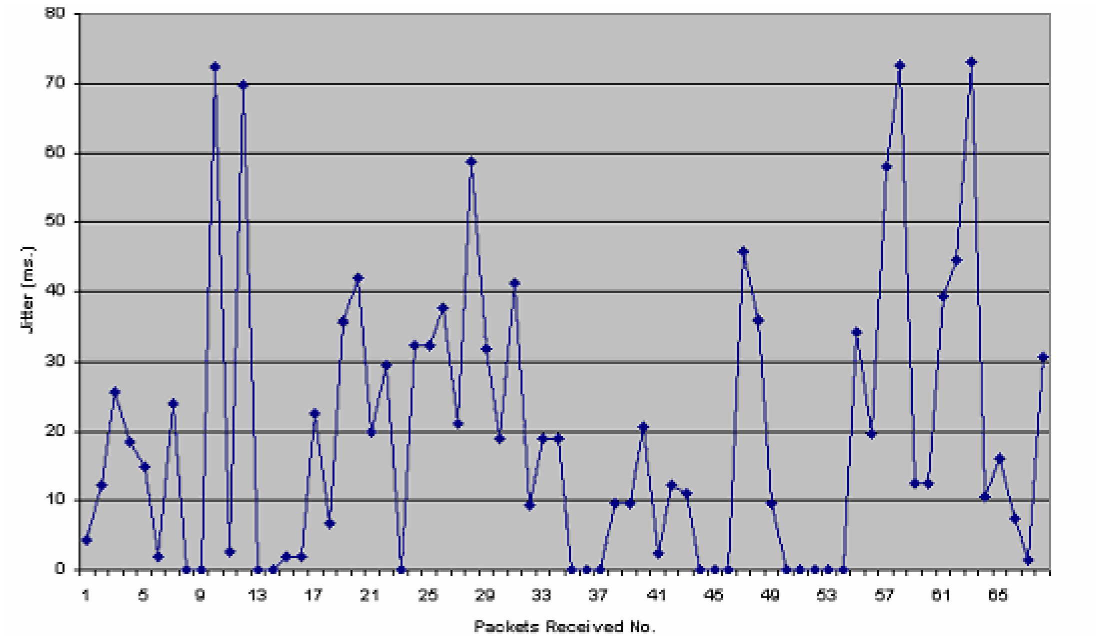
The end-to-end delay for the real-time flows decreases to a value below than 150 ms, 67, the average delay for VoIP source No.1 is 132.9 ms, and for the source No. 6 is 133 ms and finally for the video traffic source is 133.8 ms. Notice that in some packets, for example in packet number 12 for VoIP source No.1, the delay may reach up to 200 ms, these kinds of packets are better to be discarded than delaying the whole packets ,this is usually done at the receiver's buffer.

Jitter Analysis

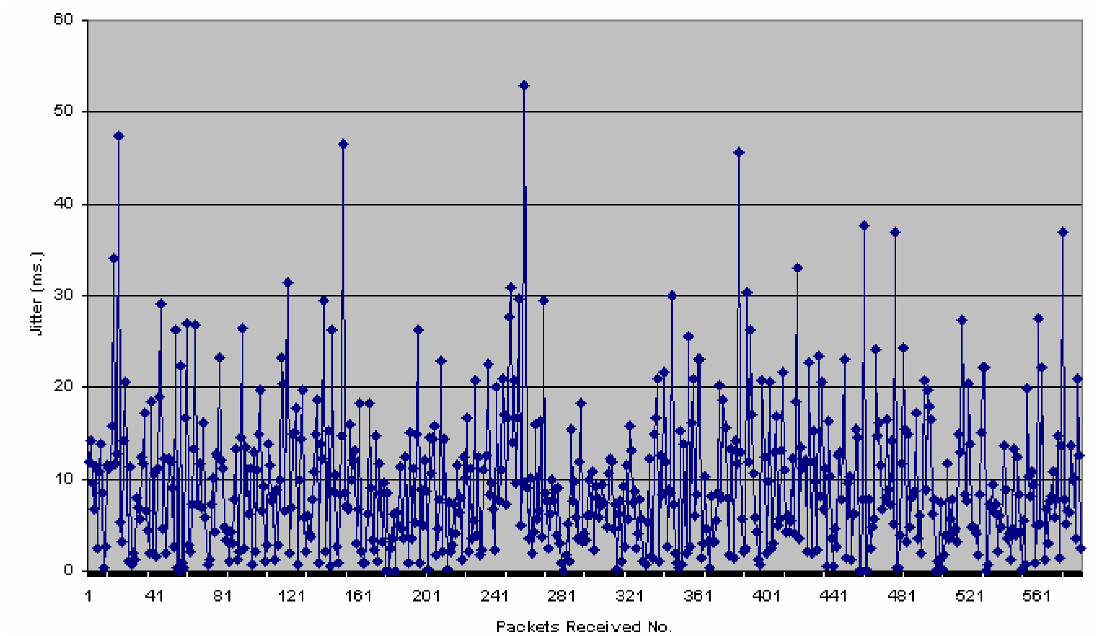
To study the jitter, the results of each case are drawn independently, to be able to conclude results. Below Figures 77, 78, 79 show the jitter delay for the same real-time traffic sources, which we are interested in, these results obtained considering the hop-normalized as the routing metric with the Widest-Least Cost algorithm.



Figures 77 Jitter Delay for VoIP Traffic Source No.1



Figures 78 Jitter Delay for VoIP Traffic Source No.6



Figures 79 Jitter Delay for Video Traffic Source No.7

The Jitter delay as expected is uniform and low, as we can see in Figure 77, the jitter delay encountered by most of the packets lies within the 10 ms region, which is small, and most of them has the same jitter values, and the values are very close, consequently, de-jittering can be successfully implemented to remove these delay variations on the received packets. Figure 78 has similar results, while the jitter delay encountered by the video traffic is higher than 10 ms, as shown in Figure 79, most of these packets encounter a delay within 20 ms, since the end-to-end delay is about 133.8 ms, which is lower than 150 ms, de-jittering technique can be used successfully to remove these variations in the delay at the receiver's side.

Case 5 Shortest-Widest Path Algorithm

This algorithm selects the path with the largest available bandwidth. If several paths exist with as large bandwidth, the one with the smallest hop-count is chosen. This algorithm is attractive and efficient, in our case, since link 2-3 has higher bandwidth than the other two links, link 1-2 and link 2-13, the routing algorithm selects the link 2-13 which is the "widest " in terms of the available bandwidth, then at the next node; node 13 , this algorithm selects the link 13-9, which has 3 Mbps, which is greater than the 1.5 Mbps of link 13-14,so the real-time traffic will not share the 1.5 Mbps link connecting nodes 13 with node 14.That leads to higher throughput for the best-effort traffic . And again the algorithm is repeated until it reaches the destination node, the selected path is link 2-13,link 13-9,link 9-10,link 10-11,link 11-12 and finally link 12-5 or link 12-4 depending on the traffic destination.

Throughput Analysis

Figures 60 shows the throughput for best-effort traffic source No.8, it is obvious that the throughput for this source reach up to 1.4 Mbps, which is equal to 90.3 % utilization for the shortest path link between node 2 and node 4. Moreover, the average throughput for the real-time sources is very near to the required bandwidth. Table 10 shows average value for the obtained bandwidth and the performance evaluation for each source.

Table 10 Throughput Evaluation, Case 5

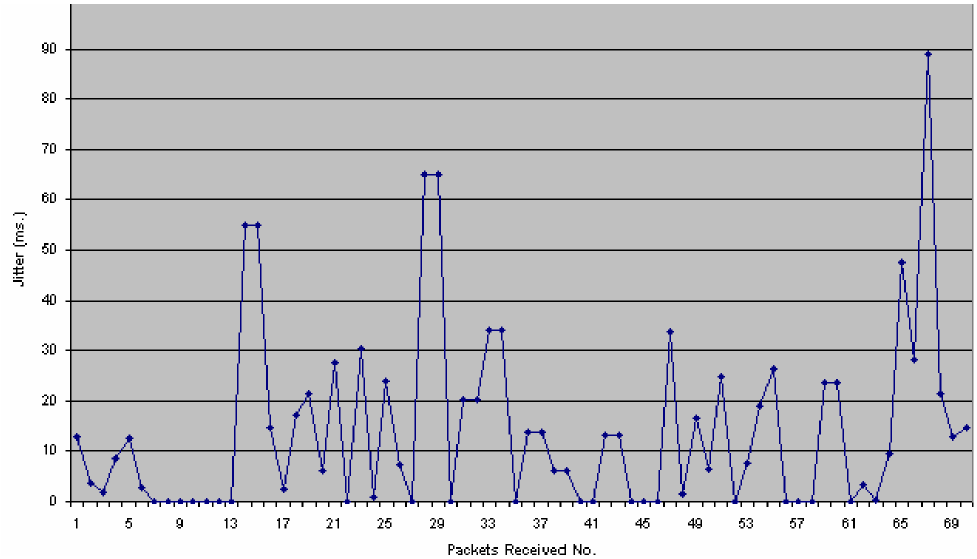
Source No.	Requested Bandwidth	Obtained Bandwidth	Packet lost%	MOS (According to Figure 8)	Performance
1	64 Kbps	61.9	3.2 %	4.7 out of 5	Excellent to Good
6	64 Kbps	60.18	5.9 %	4.1 out of 5	Good
7	512 Kbps	475.7	7.08 %	4.3 out of 5	Good to Excellent

End-to-End Delay Analysis

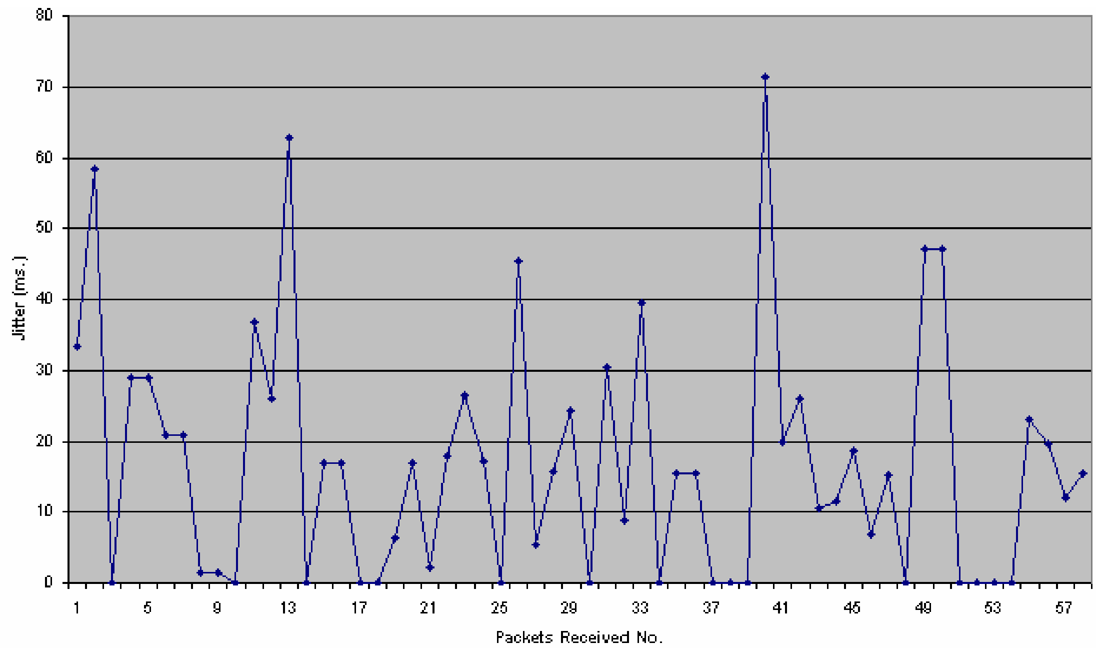
Since the real-time traffic sources avoided using the congested links, the end-to-end delay encountered by these traffic was acceptable and within the recommended bound, Figures 65, 66 and 67 shows how the delay for this algorithm is less than the delay produced by case 2, and case 3 of this scenario. And it is very near to the delay curves for case 3. The average delay for VoIP source No. 1 is 140 ms, and for VoIP source No. 6 it is 146 ms, and finally for the video source it is 143.5 ms. All of these values are below the recommended 150 ms, which results in acceptable delay and performance.

Jitter Analysis

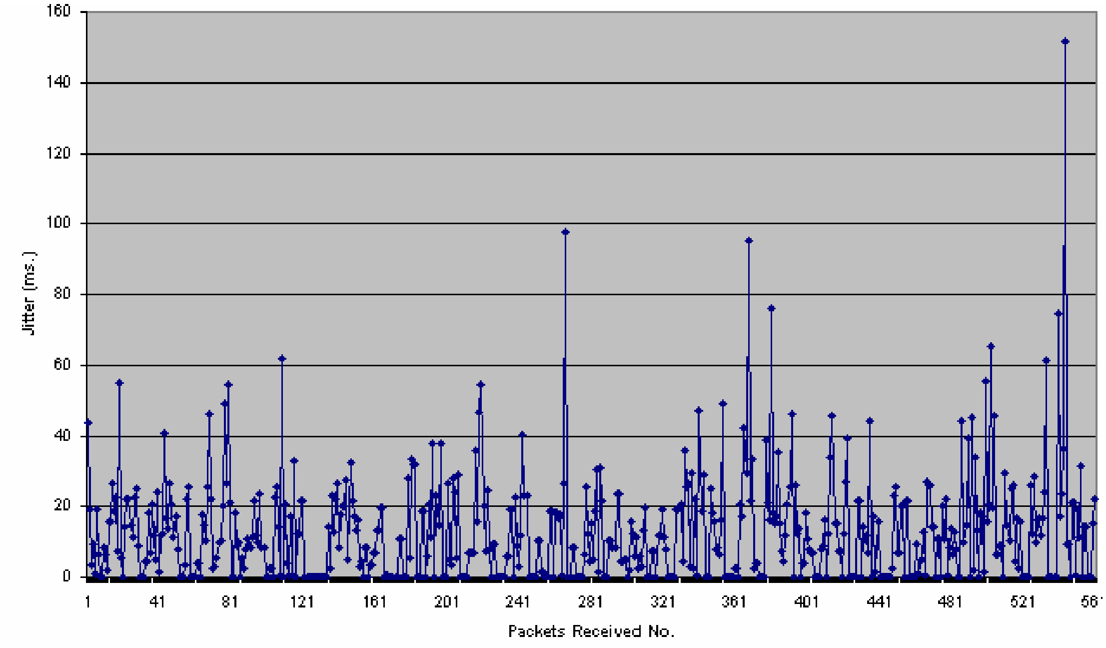
To study the jitter, the results of each case is drawn independently, Below Figures 80, 81, 82 show the jitter delay for the same real-time traffic sources, which we are interested in , theses results obtained, using the Shortest Widest Path routing algorithm.



Figures 80 Jitter Delay for VoIP Traffic Source No.1



Figures 81 Jitter Delay for VoIP Traffic Source No.6

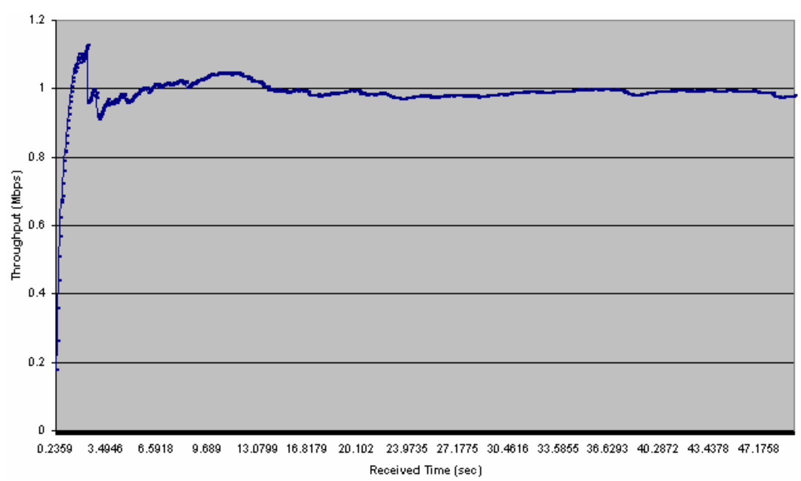


Figures 82 Jitter Delay for Video Traffic Source No.7

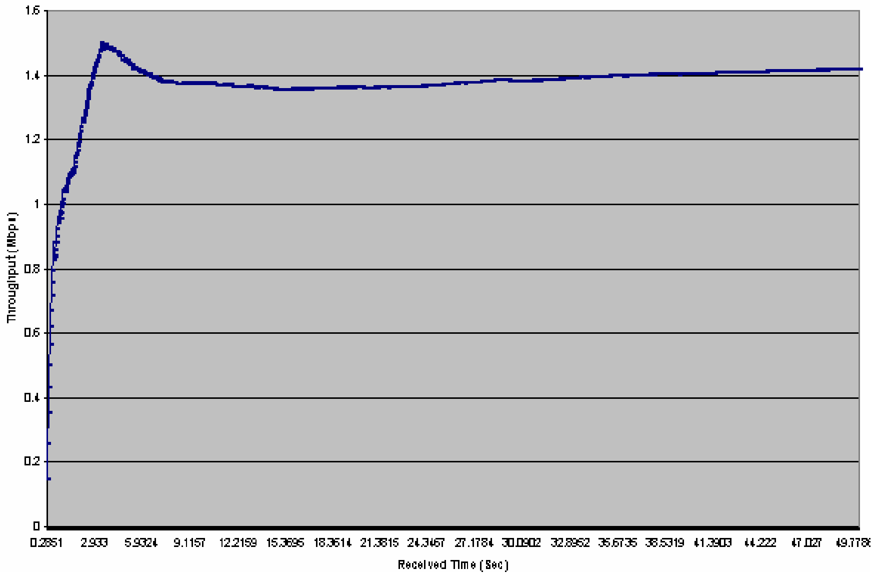
The figures above show that the jitter delay is within the range of 20 ms for the VoIP source No.1, for the VoIP source No.6, it is within the range of 20–25 ms, and finally, for the video traffic it is within the range of 20 ms. In fact, it is noticeable that the jitter values have almost the same, and the variation in the delay is not large, as in case 2, or case 3. Moreover, some packets encounter high variation in the delay, usually, such packets are ignored from the receiver side.

Shortest-Widest Path Draw Back

From the above results, the performance of real-time service, using Shortest Widest Path was good and attractive, but in fact, this algorithm may not always avoid the congested links. In our case, some links are set in away that their bandwidth is higher than the other links, so the congested link that carry the best-effort-traffic source No. 8 is avoided using this algorithm, but suppose that all of the links bandwidth are similar, i.e. 1.554 Mbps, and suppose that we have one of the real-time flows that want to transfer traffic from node 2 to node 5, the algorithm will check first the three outgoing links connected to node 2, i.e. link 2-13, link 2-1, link 2-3, since all of the three links have the same bandwidth , it will choose the node which leads to the shortest path between node 2 and node 5 ,so it will choose node 13,and the algorithm repeats again. Accordingly the shortest path between node 2 and node 5 is chosen for that real-time source, and the real-time source is routed to that path .To study this case, we have repeated the simulation, using the same bandwidth setting for all the links, we set all the links bandwidth to 1.554 Mbps, and we make the video source which has a bit-rate of 0.5 Mbps, request a path between node 2 and node 5, the same best-effort background sources exist, we used the Shortest-Widest Path algorithm and the Widest Least Cost algorithm with the Hop Normalized metric. The throughput of the best-effort traffic source No.8, is monitored and the following results are obtained, Figure 83, shows the throughput of the best-effort traffic source No. 8, when the Shortest-Widest Path algorithm is used, and Figure 84, shows the throughput of the best-effort traffic source No. 8 when the Widest-Least Cost algorithm using the hop normalized metric, is used.



Figures 83 Throughput of the Best-Effort Traffic source No, 8



Figures 84 Throughput of the Best-Effort Traffic source No, 8

From Figure 83, the throughput of the best-effort traffic is around 1 Mbps, which proves that the video traffic reserve 0.5 Mbps along that path, so the best-effort traffic utilizes the residual bandwidth. While when the Widest-Least Cost algorithm is used, the congested link that carry the best-effort traffic between node 2 and node 8 is avoided, so the video traffic is routed using another links, so the best-effort traffic utilizes most of the available link bandwidth (about 1.4 Mbps).

7 Our Contribution and Related Work

We have improved the performance of real-time traffic loads, such as the VoIP, by using the hop-normalized metric, developed by Khanna (Khanna ,Zinky, 1989) with the Widest-Least Cost algorithm, instead of the hop-count metric used with the original algorithm. In order to compare our results with others, we have studied the performance of real-time traffic using the original hop-count metric suggested by Apostolopoulos (Apostolopoulos et al., 1999), we have found that the performance of real-time applications such as the VoIP and video streaming is improved when we took into consideration the hop-normalized metric with the Widest-Least Cost algorithm, instead of the hop-count metric. The two results obtained from using the hop-count and hop-normalized metric were compared, which showed significant improvement on the performance of both types of traffic loads, the real and non real-time traffic loads. These improvements were as follows:

1) Using this metric, the QoS routing algorithm was able to route the real-time traffic to a non congested links, which results in improving the performance by reducing the end-to-end delay, reducing the jitter and increasing the throughput.
2) Regarding the best-effort traffic, the throughout increased, since the real-time traffic didn't follow the same path for the best-effort traffic, Thus the overall network utilization is improved.

Moreover, Khanna and Zinky (Khanna ,Zinky, 1989) tested the performance of their metric under best-effort traffic loads, which shows that using this metric with the shortest-path algorithm, reduces the congestion ,Thus improving the network efficiency. We have extended their work and tested their metric under real-time traffic loads. Our results showed that the hop-normalized metric, can be used efficiently with the real-time traffic, as well as the best-effort traffic loads.

Finally, we have investigated the Shores-Widest algorithm, developed by Wang and Crowcroft, (Wang, Crowcroft, 1996) and study the performance of real-time applications using this algorithm, we have found that this algorithm is efficient, and can avoid the congestion links, which improves the performance of both ,real and non-real-time traffic loads. But since this algorithm depends on the hop-count, we have discovered that it didn't always success in avoiding the congestion links.

Conclusions and Recommendations

The purpose of this thesis is to evaluate the performance of VoIP under various QoS routing algorithms and metrics, to achieve that, several simulation models were used, the obtained results are studied and analyzed in terms of three main performance measures, the throughput, the end-to-end delay and the variation on delay, these results are depicted in terms of graphs. This section summarizes this thesis; furthermore some recommendations for future works are suggested.

1 Conclusions

Our work can be summarized in the following points:

- Real-time applications, such as the VoIP, and video streaming are expected to be wildly used in the next generation Internet.
- The current Internet architecture is based on best-effort traffic, which doesn't take QoS issues into consideration, so several modifications and improvement are required toward enabling this network with QoS capabilities. So, the Internet Engineering Task Force has proposed two main QoS architectures that can differentiate real-time traffic from best-effort traffic, thus providing these time sensitive flows with the required bandwidth and delay a long the path established between the source and destination pairs, IntServ and DiffServ are two examples of these architectures.
- The IntServ architecture model is capable of providing real-time services with more strict delay and throughput constraints, while the DiffServ model is scalable and easier to implement.

- Routing issue is one of the most important challenges that directly affects the performance of real-time flows, so selecting a suitable path that can provides the real-time traffic with the required bandwidth is an important factor. That introduces the concept of QoS routing, which concerns in finding a path between a source and a destination that is capable of providing the real-time flows with QoS requirements.
- Several QoS routing algorithms exist, but there is a few numbers of them that are considered effective and have low computational complexity. The Widest-Least Cost algorithm and the Shortest-Widest Path algorithms are some examples of effective QoS routing algorithm that has low computational complexity and can be easily deployed.
- We have improved the performance of VoIP, by using the hop-normalized metric, developed by Khanna, with the Widest-Shortest algorithm, instead of the hop-count metric used with the original algorithm. To compare our results with others, we have studied the performance of VoIP using the original hop-count metric suggested by Apostolopoulos. We have found that the performance of VoIP is improved when we used the hop-normalized metric with the Widest-Least Cost algorithm, instead of the hop-count metric, which is used with the original algorithm. The two results obtained from using the hop-count and hop-normalized metric were compared, which showed significant improvement on the performance, in terms of throughput, delay and jitter, for both types of traffic loads, real and non real-time traffic.

2 Future work

More study must take place in the QOSPF, as it is expected to be the main intra-domain QoS routing protocol. In fact, in our study, in studying some QoS routing algorithms, we were able to meet one criteria which is the bandwidth guarantee, and try to minimize the other one, which is the cost function, but we didn't study the case for trying to meet two QoS requirement criteria; such as obtaining path with both bandwidth and end-to-end delay guarantee. Moreover, we have used the IntServ architecture as our QoS architecture; similar study could be done using the DiffServ QoS architecture.

Appendix A

About the QRS simulator

QRS models the network as a combination of different kinds of components. For example, the component “Node” represents the practical nodes, using parameter “delay to process a packet” to represent the processing speed and “Buffer space” to represent the buffer. The “Link” component represents the links with characteristics of bandwidth and propagation delay The “Real-time Traffic” component initiates traffic with QoS Constrains, i.e. bandwidth requirement. It contains source and sink that are connected to the source nodes and the sink nodes respectively. The routing algorithms are implemented in the “QOSPF” component, and the implementations of these two new routing algorithms are added into this component. Every node has a “QOSPF” component to maintain the routing information. The functionality of signaling path setup is simulated by the “RSVP” component. The “RM” component is responsible for the resource reservation. Every node has a “RSVP” and a “RM” component With these components, the basic procedure for the set- up of a flow is like this: Node requests RSVP for flow set-up, and then RSVP inquires QOSPF for information about the next hop, after that, RSVP sends PATH message to the next hop according to the reply from QOSPF, finally, if an acknowledgement from the destination is received, RSVP requests RM to re serve the resources.

Network Settings and Traffic Profile

Here below is the most important parameter settings used throughout our simulation

No.	Parameter Description	Values
1.	**Node Characteristics**	
	Delay to process a packet	0.1 ms.
	Buffer space in bytes	10000 bytes
2.	**Link Characteristics**	
	Link propagation delay	1 ms.
	Link bandwidth	Depends on each case
	CBQ: class A,B,C queue size	50000 for each class type.
3.	**QOSPF Component**	
	Routing method	0 for Widest -Least Cost algorithm. 1 for Widest- Shortest Path algorithm. The default is 0.
4	**HTTP Traffic characteristics**	
	Arrival time	0, which means that traffic start when the simulation starts
	Bit-rate	Determined differently in each case
5	**FTP Traffic characteristics**	
	Arrival time	0, which means that traffic start when the simulation starts
	Bit-rate	Determined differently in each case

6	**Real- Traffic characteristics**	
	Arrival time	Exponentially distributed ,as mentioned in table 5.
	Bit-rate	Determined differently in each case
	Silence/talk periods	Exponentially distributed, as mentioned in table 5
7	**Cost function characteristics**	
	Cost metric used	1: Hop normalized 2: Delay 3: Utilization 4: hop The default value is 4.
Notes: It is important to mention that due to the statistical nature of QSR simulation used; which depends on generating a random The "seed" which is used in performing the results static and calculations, each case of the above are done several times and average for the results obtained from these runs are used in our study**.**		

Appendix B

HN Metric

The Pseudocode of the HN Metric, suggested by Khanna:

Function HN-SPF (Measured. Delay, Line. Type) returns Reported. Cost

Sample. Utilization =Delay .To. Utilization [Measured. Delay]

Average.Utilizatio=.5*Sample.Utilization+. 5*Last.Average

Last. Average =Average. Utilization (stored for each link)

Raw .Cost = Slope [Line. Type] *Average. Utilization + Offset [Line. Type]

Limited. Cost = Limit. Movement (Raw. Cast, Last. Reported, Line. Type)

Revised. Cost = Clip (Limited. Cost, Maximum [Line. Type], Minimum [Line. Type]

Last reported = Revised. Cost (store for each link)

Return (Revised. Cost)

Explanations:

The value of delay is first transformed into an estimate of the link utilization. A simple M/M/l queuing model is used with the service time being the network-wide average packet size (600 bits/packet) divided by the trunk's band- width. The result is then averaged with previous utilization estimates using a recursive filter. Next, the average utilization goes through a linear transformation to normalize the metric. The change in reported cost from one update to the next is limited both in how little and much the cost can change. The final component of the HN-SPF Module enforces absolute limits on the value of the metric which is part of the normalization process.

Interpretation and example

The key to understanding SPF is to normalize the link cost in terms of hops. When a link reports a cost, the cost is relative to the costs of alternate links. For example, when a link reports a cost of 91 units while the rest of the links in the network report 30 units, the implication is that an alternate path with 2 additional hops should be used before using that link. When there are many alternate paths, most of the routes will move off this link. An interpretation which normalizes the reported cost by dividing it by the ambient cost of alternate links takes into account the effect of the reported cost relative to other links.

References

A laettinoglu, A.C., (1994). Design and implementation of MaRS: A routing testbed, Journal of **Internetworking: Research and Experience**, 5(1): 17-41.

Apostolopoulos, G., Guerin, R., Kamat, S., Orda, A. and Tripathi, S. K. (1999, September-October). Intra-Domain QoS Routing in IP Networks: A Feasibility and Cost/Benefit Analysis ,**IEEE Network**, Vol. 13, (No. 5), pp. 42-54.

Apostolopoulos, G., Guerin, R.and Kamat, S. (1999). Implementation and Performance Measurements of QoS Routing Extensions to OSPF, **Proceedings of Infocom** ’99, pp. 680-688.

Baldi, M. and Ofek, Y. (1998,March/April). End-to-End Delay of Videoconferencing over Packet Switched Networks, in Proceedings of the Conference on **Computer Communications** (IEEE Infocom), San Francisco, California, pp. 1084.

Braden, R., Zhang, L., Berson, S., Herzog, S. and Jamin,S. (1997, September). Resource ReSerVation Protocol (RSVP) - Version 1 Functional Spec, **Network working Group** RFC 2205.

Chen, S. (1999).**Routing Support for Providing Guaranteed End-to-End Quality of Service**, PhD Thesis, University of Illinois.

Chen ,S. and Nahrstedt ,K. (1998, November/December).An Overview of Quality of Service Routing for Next-Generation High-Speed Networks: Problems and Solutions. **IEEE Networks**.

Cisco Systems, (2002), Designing Internetworks for Multimedia, from **http://www.cisco.com/univercd/cc/td/doc/cisintwk/idg4/nd2013.htm#18023**

Cravis, Howard, (1981). **Communications Network Analysis**, (1st ed). Canada: Lexington Books.

Crawley ,E., Rajagopalan, B., and Sandick ,H. (1998, August) , A Framework for QoS-based Routing in the Internet, **Network working Group**, RFC 2386.

Curado ,Marilia, Joao, Quadros, Gonçalo, and Monteiro Edmundo, (2001,June), Simulation Analysis of the UC-QoS Routing Strategy, in Proc. of the SCI2001, 5th World Multiconference on **Systemics, Cybernetics and Informatics** (SCI 2001), Orlando, USA.

Fiorini, P.M. (2000).**Voice over IP (VoIP) for Enterprise Networks: Performance Implications & Traffic Models** ,Technical report, BMC Software.

Floyd, Sally.and Paxson, Vern. (2001, February). Difficulties in Simulating the Internet,AT&T Center for Internet Research at ICSI (ACIRI),Berkeley. To appear in **IEEE/ACM Transactions on Networking**.

Floyd, Sally. and Jacobson, Van. (1995, August), sharing and Resource Management Models for Packet Networks,this paper appeared in IEEE/ACM **Transactions on Networking**, Vol. 3 No. 4.

Held, Gilbert. (2000), **Voice and Data Interworking**, (2^{nd} ed). United State: McGraw-Hill.

Huitema, Christian., Cameron, Jane., Mouchtaris, Petros. and Smyk, Darek,(1999,May/June). An Architecture for Residential Internet Telephony Service, **Telcordia Technologies IEEE Network**

Khanna ,Atul. and Zinky, John.(1989),The Revised ARPANET Routing Metric, Proceedings of the ACM **SIGCOMM**, pp. 45-56, 1989.

Kuipers, Fernando., Korkmaz, Turgay., Krunz, Marwan. and Mieghem, Piet Van. (2003), Performance evaluation of constraint-based path selection algorithms, to appear in **IEEE Network.**

Li Bo, Hamdi Mounir, Jiangi Dongy, and Cao Xi-Ren(2000, April) ,QoS-Enabled Voice Support in the Next-Generation Internet: Issues, Existing Approaches and Challenges, **IEEE Communications Magazine** .

Ma. Angus, (2001), **Voice over IP (VoIP),** Technical report, Spirent Communications.Inc.

Mahbub, Hassan., Alfandika, Nayandoro. and Atiquzzaman, Mohammed. (2000, April). Internet Telephony: Services,Technical Challenges, and Products, **IEEE Communications Magazine.**

MINO, D. and Minoli, E. (1998). **Delivering Voice over, IP Networks**, John Wiley & Sons, Inc.

Moy, J. (1998,April). OSPF Version 2. **,** Ascend Communications, Inc , **Network Working Group,**RFC 2328 .

Ramkishor, K., James, P., Ramkishor,and Mammen.(2002, January), Bandwidth Adaptation for MPEG-4 Video Streaming over the Internet , DICTA2002: **Digital Image Computing Techniques and Applications,Melbourne**, Australia**.**

Reynolds, R. J. B. and Rix, A. W. (2001,April), **Quality VoIP - an engineering challenge**, BT Technology Vol 19 No 2.

Schooler, M., Eve (1997), **QoS in the Internet: An Overview**, Broadband Information Systems Lab HP Labs, Technical report, Palo Alto, CA.

Stallings, William.(2002). **High-Speed Networks and Internet: Performance And Quality Of Service**, (2nd ed). New Jersey: Prentice Hall.

Steenstrup, Marth. E. (1995). **Routing In Communications Networks**, (1st ed). New Jersy: Prentice-Hall.

Wang, zheng, (2001). **Internet QoS Architectures and Mechanisms for Quality of Service**, (1st ed). San Francisco: Morgan kaufmann.

Wang, Z. and Crowcroft, J. (1996, September). “QoS Routing for supporting Multimedia Applications”, IEEE J **Select. Areas Commun.**, 14(7):1188-1234. Salkewicsz .

Zhang, P. and Kantola, R. (2002, December). "**QoS Routing Simulator: User's Manual**". Hilsinki University of Technology, Finland.

Zhang ,P. and Kantola, R. (2001, May). "**QoS Routing in DiffServ Networks: Issues and Solutions**",Technical report. Hilsinki University of Technology, Finland.

Zhang, P., Kantola, R.and Ma, Z. (2000, July). Design and Implementation of A New Routing Simulator. 2000 SCS Symposium on **Performance Evaluation of Computer and Telecommunication Systems** (SPECTS'2K). Vancouver, Canada

Zhansong, Ma., Zhang, Peng.and Kantola, Raimo. (2001, May).Influence of Link State Updating on the Performance and Cost of QoS Routing in an Intranet. 2001 IEEE Workshop on **High Performance Switching and Routing** (HPSR 2001) (accepted), Dallas, Texas USA.

Zhang, L., Deering, S., Estrin, D., Shenker, S. and Zappala, D. (1993). RSVP: A New Resource ReSerVation Protocol**, IEEE Network** 7:8-18. Estrin.

Zhao, Weibin., Olshefski, David. and Schulzrinne, Henning. (2000). **Internet Quality of Service: an Overview**, Technical Report, Hilsinki University of Technology, Finland.

معدل التأخير الأدنى في نقل الرزمة عبر شبكات الاتصال الرقمية

إعداد

خولة توفيق رشيد عمر

المشرف

الأستاذ الدكتور جميل أيوب

الملخص

تبحث هذه الرسالة في موثوقية شبكات الحاسوب ذات الرزمة عن تعرضها لظروف العطل العشوائي.تعتمد الطرق التقليدية في الإعتمادية على حساب مقدار الربط الموجود بين نقطتين، أو على حساب احتمالية نجاح الاتصال بين نقطتين . وتعتمد طرق أخرى على حساب كفاءة الشبكة عند خلوها من العطل.

النموذج المقترح من قبلنا يأخذ مقاييس الإعتمادية السابقة الذكر بعين الاعتبار، ويحاول أن يكون اقرب إلى ظروف العمل الحقيقية لأي شبكة حيث أن هذا النموذج يستند إلى حقيقة واقعة وهي مقدرة الشبكة على الاستمرار في نقل المعلومات عند تعرض بعض خطوطها للعطل ، وإمكانية حساب كفاءة الشبكة تحت هذه الظروف. مقياس الكفاءة يعتمد عل الطريقة المتبعة لتسير الرزم في الشبكة، وعلى الافتراض أن الرزم تغير مسارها في حال حدوث عطل في مسارها الأصلي.

تتعرض الشبكات الواقعية إلى العطل بحيث يرتبط بكل خط فيها احتمالية معينة هي: احتمالية العمل الصحيح، أو احتمالية عدم التعرض للعطل. ويعتمد على هذه الاحتمالية وضع معين الشبكة. بحيث عندما تكون الشبكة في وضع معين تكون بعض خطوطها صالحة وبعضها قد تعرض للعطل.

في هذه الدراسة مقياس الكفاءة الذي تبنيناه هو متوسط الوقت اللازم لنقل جميع الرزم على الشبكة. حيث يحسب هذا الوقت الأوضاع الأكثر احتمالا التي تكون الشبكة فيها، مثل: الأوضاع التي لا يحصل فيها عطل، الأوضاع التي تعطل فيها خط واحد فقط، الوضاع التي يتعطل فيها خطين في نفس الوقت. ثم يمكن حساب متوسط الوقت اللازم لنقل الرزم عبر الشبكة من خلال حساب متوسط الوقت اللازم لنقل الرزم في جميع أوضاع الشبكة السابقة.

الطرق المتبعة لنقل الرزم تعتمد على طريقة اقصر مسار حيث كلمة اقصر تعني المسار ذات الإعتمادية الأعلى. حيث يستعمل هذا المسار في ظروف العمل الاعتيادية، وثاني اقصر مسار، حيث يستعمل عند زيادة عدد الرزم الموجهة على المسار الأول، يأتي هذا الاتجاه في محاولة لتقليل الوقت اللازم لمرور الرزم عبر الشبكة.

بعد الحصول على متوسط الوقت اللازم لمرور الرزم عبر الشبكة في جميع أوضاع الشبكة الأكثر احتمالا يمكن الحصول على متوسط لهذه الأوقات بحيث يكون هذا المتوسط مقياس الكفاءة الذي نبحث عنه.

تم تطبيق هذه الطريقة على شبكات مختلفة الحجم، كما تم بحث تأثير كمية الزم، وتأثير تغيير الاحتمالية الخاصة بعمل الخطوط على مقياس الكفاءة المتخذ.

إن هذه الطريقة توفر مقياسا مفيدا وقريبا للواقع يمكن أن يستعمل لقياس كفاءة الشبكات، كما يمكن الاستفادة منه في تصميم شبكات الحاسوب المختلفة.