Quality of service assessment and analysis of wireless multimedia networks.

AL-SBOU, Yazeed Ahmad.

Published version

AL-SBOU, Yazeed Ahmad. (2006). Quality of service assessment and analysis of wireless multimedia networks. Doctoral, Sheffield Hallam University (United Kingdom)..

Quality of Service Assessment and Analysis of Wireless Multimedia Networks

Yazeed Ahmad Al-Sbou

A thesis submitted in partial fulfilment of the requirements of
Sheffield Hallam University
for the Degree of Doctor of Philosophy

October 2006

ABSTRACT

Quality of Service Assessment and Analysis of Wireless Multimedia Networks

Recent years have witnessed a vast technological progress in the area of Quality of Service (QoS), mainly due to the emergence of multimedia networking and computing. QoS measurement and analysis have long been of interest to the networking research community. The major goals of this thesis are of two fold: Firstly, to investigate the effect of the QoS parameters on the overall QoS experienced by wireless networks. Secondly, to utilise the results in developing efficient mechanisms for intrusive and non-intrusive assessments of the performance of wireless ad hoc networks as well as the measurement of the available QoS for audio and videoconferencing applications over the IEEE 802.11 standard.

To evaluate the network performance and the overall QoS of multimedia applications, new fuzzy logic and distance measure assessment approaches were developed taking into account the QoS parameters requirements of each application. The developed approaches essentially include measuring the main QoS parameters (delay, jitter and packet loss) and use them as input to the measurement systems, which combine them and produce an output that represents the instantaneous QoS. The devised approaches showed how the QoS can be measured without a need for complicated analytical mathematical models.

In this study, several techniques were devised for estimating QoS. Firstly, a probe-based assessment method (active technique) was developed. In this method, special artificial monitoring packets were injected into the network. The overall QoS and its parameters were estimated by collecting statistics from these packets. It was possible to make reasonable inferences about the delay, throughput, packet losses and the overall average QoS using different probe rates. This technique showed some limitations for measuring the jitter. In addition, the rate of the monitoring packets played an essential role in the precision, level of resolution of estimated results and negatively impacted the network performance.

Secondly, to overcome some of the drawbacks of the probing-based method, a new assessment technique was, subsequently, devised based on passive monitoring standard sampling methods. Unlike the active technique, the new method has the advantage of not adding an extra load to the network. In addition, it is not like the typical passive methods, which require the transfer and calculations of the whole captured data. Generally, all sampling schemes provided satisfactory measures of the overall QoS and its parameters and produced very acceptable bias and Relative Standard Error (RSE) result. Systematic sampling provided the most accurate estimates compared to the stratified and random approaches. In addition, after sample fraction of 2%, the estimated overall QoS bias from the actual QoS became constant and equal to -0.5% and RSE was less than 0.005 using both fuzzy and distance assessment systems.

Thirdly, in order to overcome some negative aspects of inaccuracy and biasness caused by sampling techniques, a new scheme was proposed to correct these results to be closer to the actual traffic measurements. The new approach does not disturb the network performance (as in active methods), neither depends on the whole traffic (as in passive methods), nor bias the actual results (as in the standard sampling technique). Similarly, systematic sampling showed the best performance. Sample fractions, using the systematic sampling, greater than 2% gave an overall estimated QoS identical to the actual QoS because the obtained relative error was nearly constant and approximately close to zero using both assessment systems.

The measured QoS can be used to optimise the received quality of the multimedia services along with the changing network conditions and to manage the utilisation of the network available resources especially for ad hoc networks. Overall, the findings of this study contribute to a method for drawing a realistic picture of the wireless multimedia networks QoS and provide a firm basis and useful insights on how to effectively design future QoS solutions.

Dedication

This thesis is dedicated with affection to:

My parents, Ahmad and Rasmyah Al-Sbou, who have waited so long for this.

My soul-mate, Nesrin, who instils me the confidence that I am capable of doing anything I put my mind into! Really, I could not function without you!!

Yazeed

Acknowledgments

All thanks to Almighty **ALLAH** for giving me the guidance and the strength to achieve the goal of obtaining my PhD.

This thesis would not have been prepared without the help and guidance of a number of key people. First, I would like to express my deep thanks to my director of studies, Dr. Reza Saatchi, for his support, guidance, encouragement, valuable comments and suggestions on various aspects throughout my PhD study. The gentleness and kind treatment I have received from Dr. Saatchi will stay with me over my life. Special thanks to my supervisor, Dr. Samir Al-Khayatt for being very supportive and for providing me with his valuable guidance, advice and expertise. Dr. Samir, I will never forget your help and care, you taught me how a supervisor can be a sincere friend. Also, I would like to thank Dr. Rebecca Strachan for her valued help, support, and instructions. Thank you all for reviewing my thesis and giving me every possible advice that have significantly improved the quality of this thesis. Thanks to my Ph.D. exam committee members: Dr. Reza Sahandi and Dr. Graham Swift for their constructive and valuable suggestions.

Special thanks go to my sponsor, Mu'tah University, for the generous financial support over the past three years.

Although a few words do not do justice to their contribution, I am deeply indebted and owe a great gratitude to my loving parents, Ahmad and Rasmyah Al-Sbou, for their unlimited and non-stop support, love, trust, and sacrifices in whatever I choose to do and not only during my PhD stage but also throughout all stages of my life. Thanks for the countless things you offered which would take ages to list here. Dear parents, May **ALLAH** always give you health and happiness.

My special gratefulness and admiration to my lovely beloved wife, Nesrin Mwafi, for her endless and continuous love, support, understanding, and encouragement. Over the past two years, we have shared a lot in our life, love, happiness, worries, and even pain. Nesrin, you are amazing in your patience and care. Thank you so much! I should also thank my lovely daughter, Mariamnoor, who spent the time away from our eyes. Dear Mariam, even you are far away but you are always in our hearts and souls.

Many thanks to my dear brothers, Nidal, Nizar and to lovely sister, Huda for their selfless care and support. Also, thanks to my sisters in-law and nephews, especially Hamodeh.

I am very thankful to my father and mother in-law, Riad Mwafi and Rawadah Alzayat, to my brothers' in-law, their wives and children for their support and encouragement.

Also, I would like to thank my colleagues in Sheffield Hallam University, especially my laboratory mates, Mohammad Saraireh and Aicha Said for the nice times and valuable discussions and suggestions.

Special appreciation to my dear, Dr. Mousa Ayyash, who did not occlude any possible help.

List of Publications

Material from this thesis has appeared in the following publications, presented in chronological order:

- Al-Sbou, Y. A., Saatchi, R., Al-Khayatt, S. and Strachan, R. (2005). A Comparative Study of Sampling Methods for the Measurement of QoS Parameters. *In Proceedings of the Postgraduate Research Conference on Electronics, Photonics, Communications & Networks, and Computing Science (PREP'05)*, 38-39. University of Lancaster, Lancaster, UK.

- Al-Sbou, Y. A., Saatchi, R., Al-Khayatt, S. and Strachan, R. (2005). Sampling Methods Evaluation for the Measurement of QoS Parameters. *In Proceedings of the 1st International Conference on Internet Technologies and Applications (ITA05)*, University of Wales, NEWI, Wrexham, UK. ISBN: 0-94688-132-4.

- Al-Sbou, Y. A., Saatchi, R., Al-Khayatt, S. and Strachan, R. (2005). Estimation of the Distributions of the QoS Parameters Using Sampled Passive Measurements Techniques. *In Proceedings of 2nd International Conference on E-Business and Telecommunication Networks (ICETE05)*, (1) 324-329. Reading, UK. ISBN: 972-8865-33-3.

- Al-Sbou, Y. A., Saatchi, R., Al-Khayatt, S. and Strachan, R. (2006). Quality of Service Assessment of Multimedia Traffic over Wireless Ad Hoc Networks. *In Proceedings of Fifth International Symposium on Communication Systems, Networks and Digital Signal Processing. (CSNDSP2006)*, 129-133. University of Patras, Patras, Greece. ISBN: 960-89282-0-6.

- Al-Sbou, Y. A., Saatchi, R., Al-Khayatt, S., Strachan, R. and Ayyash, M. Quality of Service Assessment of Audio Traffic over Wireless Ad Hoc Networks. *Submitted to the ACS/IEEE International Conference on Computer Systems and Applications, AICCSA '2007*. May 13th-16th, 2007. Philadelphia University, Amman, Jordan.

Table of Contents

List of Figures

List of Tables

Abbreviations

ACK	Acknowledgement
ACR	Absolute Category Rating
AODV	Ad Hoc On Demand Distance Vector
AMP	Active Measurement Project
AQUILA	Adaptive Resource Control for QoS Using an IP-based Layered Architecture
AP	Access Point
ATM	Asynchronous Transfer Mode
BER	Bit Error Rate
BPSK	Binary Phase Shift Keying
BSA	Basic Service Area
BSS	Basic Service Sets
CAC	Call Admission Protocol
CBR	Constant Bit Rate
CDF	Cumulative Distribution Function
CFP	Contention Free Period
CI	Confidence Interval
CIL	Confidence Interval Length
CoMPACT	Change-of-measure based active/passive monitoring
CoS	Class of Service
CP	Contention Period
CSMA/CA	Carrier Sense Multiple Access/Collision Avoidance
CSMA/CD	Carrier Sense Multiple Access/Collision Detection
CTS	Clear to Send
CW	Contention Window
DCF	Distributed Coordination Function
DCR	Degradation Category Rating
DIFS	DCF Inter-frame Space
DMOS	Degradation Mean Opinion Score
DS	Distributed System
DSDV	Destination-Sequenced Distance Vector
DSSS	Direct Sequence Spread Spectrum
DSR	Dynamic Source Routing
ESS	Extended Service Set
ETSI	European Telecommunications Standards Institute
FHSS	Frequency Hopping Spread Spectrum
FSK	Frequency Shift Keying
FTP	File Transfer Protocol
GPS	Global Positioning System
GTB	Global Time Base
IBSS	Independent BSS
IEEE	Institute of Electrical and Electronics Engineering
IETF	Internet Engineering Task Force
IFQ	Inter-Frame Queuing
IP	Internet Protocol
IPM	Internet Performance Monitor
IPPM	IP Performance Metric
IR	Infra Red

IS	Integrated Service
ISM	Industrial Science Medical
ITU	International Telecommunication Union
LAN	Local Area Network
MAC	Medium Access Control
MOS	Mean Opinion Score
MPDU	MAC layer Protocol Data Unit
NAV	Network Allocation Vector
NTP	Network Time Protocol
NS	Network Simulator
OAM	Operation, Administration, and Maintenance
OSI	Open System Interface
OWD	One-Way Delay
PCF	Point Coordination Function
PESQ	Perceptual Evaluation of Speech Quality
PIFS	PCF Inter Frame Space
PLCP	Physical Layer Convergence Procedure sub-layer
PMD	Physical Medium Dependent sub-layer
PPM	Pulse Position Modulation
PPP	Packet Pair Probing
PSQM	Perceptual Speech Quality Measure
PTP	Packet Train Probing
QoS	Quality of Service
QPSK	Quaternary Phase Shift Keying
RD	Route Discovery
RF	Radio Frequency
RFC	Request for Comments
RNN	Random Neural Network
RQoSR	Robust Quality of Service Routing Protocol
RSE	Relative Standard Error
RSVP	Resource Reservation Protocol
RTT	Round-Trip Time
RTS	Request to Send
SE	Standard Error
SA	Service Assurance
SLA	Service Level Agreement
SNMP	Simple Network Management Protocol
SM MIB	Service Monitoring Management Information Base
SIFS	Short Inter Frame Spacing
TORA	Temporally-Ordered Routing Algorithm
TCP	Transmission Control Protocol
UDP	User Datagram Protocol
UTC	Coordinate Universal Time
VoD	Video on Demand
VoIP	Voice over IP
WAND	Waikato Applied Network Dynamics
WLAN	Wireless Local Area Network

Introduction

This chapter provides an introduction for the main topics and the work carried out in this study. This includes detailed objectives set, the motivation behind this research, main contributions and outlines the structure of the thesis. This chapter is organised as follows. Section 1.1 presents the thesis main aims and objectives. Section 1.2 outlines the motivations behind the study. Section 1.3 summarises the major contributions. And finally, Section 1.4 provides the outline and organisation of the thesis.

1.1 Aims and Objectives

The overall aim of this research is to analyse the Quality of Service (QoS) in wireless computer networks for multimedia transmission under various operating conditions. Several techniques are utilised so that the complex task of assessing and quantifying the QoS can be achieved effectively and efficiently. This study evaluates existing methods and devises new methods for measuring and quantifying the overall QoS of wireless networks transmitting multimedia applications. The study uses the IEEE 802.11 protocol for performing and validating the proposed measurement mechanisms.

The objectives of this study are to:

(i) Investigate techniques, which enable the QoS performance of wireless networks for transmission of multimedia applications to be assessed and quantified.

(ii) Study the effects of operational conditions and resource availability for providing the required QoS.

(iii) Quantitatively evaluate and analyse the QoS performance of wireless networks for transmission of multimedia applications.

(iv) Investigate the possibility of the inferring QoS/performance for wireless networks transmitting multimedia applications.

(v) Explore how the findings of these methods can be used as a part of an efficient wireless ad hoc network QoS monitoring system.

1.2 Research Motivations

The transmission of multimedia over computer and communication networks has gained importance during the last few years. This is due to the fact that multimedia services are playing an important role in the human's daily life. Service providers have a great challenge to ensure they provide the required quality of multimedia applications and services.

Wireless networks are gaining widespread popularity as they allow communications to be set up without the constraints of physical wires. Wire-free transmission is viewed as an advantage since people want to move freely while they are communicating with each other. Applications of wireless ad hoc networks occur in situations such as emergency search-and-rescue operations, meetings or conventions in which users wish to quickly share information, and data acquisition operations in hostile terrain. In situations like battle fields or major disaster areas, ad hoc networks need to be deployed immediately without base stations or wired infrastructures. These networks are typically characterised by scarce resources (e.g. bandwidth, battery power, processing and storage limitations, etc.), lack of any established backbone infrastructure, high error rates, and a dynamic topology since each node is free to join or leave the network at any time (Dupcinov and Krco, 2002). A challenging but critical task that researchers tried to address over the past few years is the development of monitoring schemes that suit the characteristics of ad hoc networks. To grant QoS for such applications and to ensure that the supported QoS is sustained, it is necessary to include a process to monitor the performance and the QoS of these applications and to manage the available resources of the entire network.

Network management processes employ a variety of tools, applications, and devices to assist human network managers in monitoring and maintaining networks (Cisco, 2006) (Eikenes and Grostol, 2003). Since the early years of data communications, network performance assessment and measurement have played a key role in the continuous improvement and evolution of networking technologies. Therefore, the development in the area of performance measurement is still a focus of ongoing intensive research activities.

The evolution of wireless networks and real time applications introduces new challenges in supporting predictable and reliable communication performance. These challenges are a consequence of the vastly increasing number of current and future multimedia products that find applications not only in wired networks but also in the wireless and mobile environment and hence require special attention. The quality of multimedia applications transmitted over wireless networks is governed by the QoS provided by the network. However, wireless networks operations are constrained by the limitations of the free-space channel (i.e. relatively low bandwidth, electromagnetic interference, fading, etc). In addition, QoS provided by these networks is dependent on many other factors such as the transmission power, bandwidth, form of data coding, transmission rate control, and the route that data will follow to travel from source to destination. A major challenge in such networks, due to its critical characteristics and limitations, is how to measure or infer the quality of multimedia applications accurately and efficiently for QoS monitoring and/or control purposes. In addition, this needs to be done on a continuous basis to make sure that the strict technical and commercial QoS requirements (e.g. Service Level Agreements (SLA)) are met along the service delivery.

With greater demands on wireless communications and emergence of bandwidth-intensive multimedia applications, QoS provisioning in wireless multimedia network is becoming more and more important (Kwon, et al. 2003). This is because multimedia applications contain video, data and audio elements, which need to be received with an acceptable delay, jitter, distortion and synchronization. Any violation of these requirements or insufficient bandwidth means that the received applications become useless or of limited value. Mechanisms, which improve the networks ability to transfer the multimedia with a greater quality, are of particular importance in current and future research. QoS measurement is the term, used to include all these issues which must be taken into account when wireless networks are used to transmit multimedia applications.

Wireless computer networks are evolving to provide services with diverse performance requirements. To provide QoS guarantees to these services and assure that the agreed QoS is sustained, it is not sufficient to just commit resources since QoS degradation is often unavoidable (D'Antonio, et al., 2003). A degradation of the QoS will be due to any weakening or fault in the behaviour of any network element. Hence, the QoS measurement and monitoring are essential for tracing the ongoing QoS, comparing the measured QoS against the required (expected), detecting possible QoS degradation, and then, based on the measured QoS, trying to tune the network resources accordingly to sustain the agreed QoS.

Generally, in any networking environment, there are two approaches to fulfil the QoS requirements, namely, over-provisioning and traffic engineering (Crawley, et al. 1998) and (Nahrstedt and Chen, 1998). Over-provisioning simply considers enhancing the network capabilities (e.g. buffer sizes, media types or routers upgrade) based on continuous assessment and monitoring of the network and application's QoS. Traffic engineering works by utilising resources efficiently and by making the network QoS aware (i.e. traffic classes, resource reservation, admission control, queuing mechanisms) relying on QoS measurement. Therefore, network performance monitoring is an absolute prerequisite for the QoS provision over a communications network because network managers can not manage and control their network unless they can monitor its performance (Tham, et al., 2000).

Consequently and in addition to the above, in order to better understand the network and the customer behaviours and to provide QoS to as many customers as possible, the state of the network should be always observed by obtaining measurement data from the network to accomplish the following tasks (Asgari, et al., 2003):

(i) Assist traffic engineering in making provisioning decisions for optimising the usage of network resources and take appropriate actions on setting up new routes, modifying existing routes, performing load balancing among routes, and re-routing traffic.
(ii) Assist traffic engineering in providing analysed traffic and performance information for long-term planning in order to optimise network usage and avoid undesirable conditions.
(iii) Verify whether the QoS/performance guarantees (negotiated between a customer and a service provider) committed in the SLA are being met.

As a result of the increased interest to control network and application performance, the importance of end-to-end measures, the lack of standardisation in the area of measurements and the fast pace of development, the focus of the research moved towards observing the network features, estimating and assessing the QoS measures, since they play an important role in any QoS architecture solution. The importance of QoS assessment is in its ability to greatly improve network utilisation and application performance by measuring the ongoing QoS and then to feedback of the resulting data to the service provider to ensure that the application QoS requirements have been met. Once these requirements are met, the service provider tries to keep this state over the whole service provision period by inspecting the application QoS regularly.

In this thesis, there are three issues of concern, namely QoS, wireless networks, and multimedia applications. The aim is to quantify, assess and analyse the QoS parameters and the overall QoS of wireless computer networks for multimedia transmission under varying operating conditions. Based on this analysis and quantification, the possibility of estimating the QoS for wireless networks, while transmitting multimedia, will be explored. Wireless network link stability and its resource availability change over time depending on many factors like number of users, mobility, etc. Therefore, the measured QoS is an indication of network behaviour because it reflects the resources availability that are shared among the competing traffic in the network.

In this thesis, a research plan consisting two major areas of work have been identified. The first area is to propose novel, intrusive and non-intrusive approaches to assess and analyse the overall QoS provided by the network in order to reduce or eliminate the disadvantages of current network monitoring approaches.

The second area is to develop novel alternative methods to the current performance monitoring methods to overcome their limitations for observing network conditions such as jitter, loss, delay, throughput and to allow new techniques to estimate the overall QoS. These methods determine the current status of network in a non-invasive manner, using analysis of injected traffic (as in active methods) and existing traffic (as in passive methods) using sampling techniques. It will be shown that, even the current active measurements are doing well in investigating and describing the wired network characteristics, the complexity of the wireless networks is likely to make them costly in terms of network resources.

This study seeks to address the relationships between received application quality and IP network impairments (e.g. packet loss, jitter and delay) and then looks at how the quality should be measured efficiently for multimedia applications transmitted over wireless networks. QoS measurement and monitoring are very useful in handling the challenge of unpredictable and variable QoS parameters over the wireless channel and preventing severe degradation in the applications performance. Determining the application QoS requirements allows the user's perceived quality for that application to be inferred based on the corresponding measured QoS parameters.

1.3 Contributions

This research has led to the development of efficient QoS assessment and monitoring systems for measuring the QoS of multimedia applications. The contributions of this thesis are summarised as follows:

(i) Development of a new fuzzy logic-based QoS measurement system to assess the QoS/performance of multimedia applications transmitted over wireless ad hoc networks.

(ii) Development of a new QoS assessment system using the concept of Euclidean and Minkowski distance theorems to evaluate the QoS/performance of multimedia applications transmitted over wireless ad hoc networks.

(iii) Development of techniques to monitor the performance of ad hoc networks in terms of satisfying the QoS requirements of multimedia applications based on combined active-passive measurement methods.

(iv) Development of a simple pure passive monitoring mechanism based on sampling techniques was devised to overcome some of the drawbacks of the combined active-passive monitoring method.

(v) Development of an estimation model for correcting the outcomes of the proposed passive sampling monitoring approach to be closer to the actual results.

1.4 Thesis Outline

This thesis is structured as follows. In addition to this chapter, there are eight other chapters. Chapter 2 provides the theoretical background behind this thesis which, comprises of: the QoS definition and aspects, the wireless networks revision and a description of the mechanisms and techniques used. Chapter 3 reviews the QoS assessment and measurement methods which have been classified in to two categories "subjective/objective" and "passive/active" techniques. This chapter includes a discussion of the advantages and disadvantages of each method and some examples of the measurement tools implemented based on these techniques. Chapter 4 presents research assumptions and approaches followed throughout this thesis, which involves explanation of the simulation model and the network scenarios, protocols and topologies used in this research. In Chapter 5, two new QoS assessment and measurement approaches are discussed and tested through extensive simulation experiments. Chapters 6, 7 and 8 include three different new estimation mechanisms of the overall QoS. Chapter 6 evaluates the combined active-passive estimation approach. Chapter 7 examines the suitability of the standard sampling methods for inferring the overall QoS. Additionally, Chapter 8 provides the results of another QoS estimation system based on corrected passive samples. Finally, Chapter 9 concludes this thesis and highlights future research directions and plans.

Theory and Background

2.1 Introduction

The purpose of this chapter is to provide the information and background related to the main issues of this thesis. These issues are summarised in four major sections. The first section provides an overview about the QoS aspects and multimedia applications. The second section presents a brief discussion about the wireless networks. The third section describes the fuzzy logic theory and operation. Finally, the fourth section gives some details about the Euclidean and Minkowski distance theorems.

2.2 Quality of Service Overview

The notion of QoS has become a very dominant and a widely recognised term in many aspects of our daily life, since several network applications (real and non-real time) have started to spread on a large scale. QoS is one of the greatest challenges in networking systems, wired (i.e., the Internet) and wireless, because the aim is to provide guaranteed services for telecommunication networks. Therefore, QoS has been one of the principal topics of research and development for many years (Tanenbaum, 2003), (Ferguson and Huston, 1998).

2.2.1 Defining Quality of Service

In the field of telecommunications, a “service” is defined as the ability of a network to transmit dedicated information (Galetzka, 2004). There is a close association between a service, the service provider and the network. These three terms are merged with each other through the QoS concept. QoS concept is now standardised by the International Telecommunication Union (ITU) (ITU, 1994) (ITU, 2001a) (ITU, 2001b). QoS refers to a broad collection of networking techniques where the goal is to provide guarantees on the ability of the network to deliver expected services for an application in presence of network resources sharing with different applications. QoS offers the ability to be classified qualitatively (e.g. Class of Service (CoS)) or quantitatively (e.g. delay, throughput...etc). Qualitative QoS definitions relate the treatment received by a class of packets to some other class of packets, while quantitative definitions provide metrics such as delay or loss, either as bounds or as statistical indications (Zhao, et al., 2000). So, it is the capability of the network to support service to selected network traffics over various technologies including Ethernet, wireless networks, Asynchronous Transfer Mode (ATM), ... etc (Agarwal, 2000). The other definitions for QoS are:

- "QoS is the collective effect of service performance which determines the degree of satisfaction of a user of the service" (ITU, 1994).
- "QoS is a collection of technologies, which allow network-aware applications to request and receive predictable service levels in terms of data throughput capacity (bandwidth), latency variations (jitter) or propagation latency (delay)" (Saliba, et al., 2005) .
- "QoS represents the set of those quantitative and qualitative characteristics of a distributed multimedia system necessary to achieve the required functionality of an application" (Caprihan, et al., 1997).
- "QoS in ATM is defined as a collection of rate, latency, jitter, loss ratio, and error ratio" (Maggie and Matchman, 2000).
- "QoS is a concept by which applications may indicate their specific requirements to the network, before they actually start transmitting information data" (Fluckiger, 1995).
- QoS is an answer to the question: "How well does a particular service perform relative to expectations" (Hardy, 2002).

Different viewing angles on QoS which can be summarised as follows (Räisänen, 2003):

- QoS requirements of a customer which include a statement of the quality level required by the applications of users of a service which may be expressed non-technically.
- QoS offered or planned by provider is a statement of the quality level which the service provider expected to deliver to the customer.
- QoS delivered or achieved by provider is a statement of the level of the actual quality achieved and delivered to customer.
- QoS perceived by customers is a statement expressing the quality level that they have experienced.

Therefore, QoS is a very important concept of many application domains, but especially for multimedia applications like audio, video, teleconferencing, etc. In this study, we define the QoS as the amalgamation and mapping of the main QoS parameters (delay, jitter, packet loss ratio and throughput) to obtain single representative measure of the quality achieved by a multimedia application transmitted over a computer network.

2.2.2 Issues behind QoS Assessment and Monitoring

The importance of QoS stems from the recent growth of the need of real and non-real time multimedia applications as well as the higher demand for the quality of these applications. Since communication networks have become a very essential part of our life, many efforts were made towards improving their performance. These are to achieve more and more customer satisfaction which lead to strong loyalty and therefore, to more profit for the service providers and to achieve global efficiency in resource utilisation (Alkahtani, et al., 2003). Multimedia applications QoS can be guarantied by expanding the bandwidth, but this is not always possible, costly and can not remedy the root problem. Consequently, managing and controlling the available network resources are the points to deal with to solve this problem. These can be achieved only by measuring and monitoring the network/application QoS. One of the main motivations behind deployment of the QoS is the increasing multimedia application requirements with limited resources and limited QoS support in IP networks (Braun, 2004). Therefore, QoS assessment is an essential element for satisfying different services requirements for number of applications that are sharing the same infrastructure (Jiang, 2003).

Individuals interested in the process and the result of the QoS monitoring and assessment are end-users, network manager and operators, service providers, vendors, and researchers. End-users need to perform and collect QoS measurements to make sure that the received services meet the agreed levels between them and the service providers (i.e. SLA). In addition these measurements are important for the network managers to diagnose network problems and failures, optimise the network performance, and ensure that the offered services to service providers and end-users satisfy the SLAs. Service providers depend on other parts (e.g. network provider) to grant network services to their customers. Therefore, it is essential for the service provider that the SLAs with other parts are satisfied and the services delivered meet the QoS requirements of their customers. Moreover, QoS measurements play an important role as inputs to the research communities to enhance the understanding of the network behaviours and problems which will lead to develop better solutions and build models for analysis and simulations.

Intelligent management, monitoring and control of the use of network resources within the network infrastructure are needed to meet the required QoS that will allow for these resources to be shared efficiently. In addition, QoS monitoring and assessment provide tools for the network managers to deliver mission critical business with an appropriate level of quality over public network (wired and wireless). Moreover, continuous QoS assessment allows keeping track of the network health status. However, using our QoS definition, will easily allow the manager to control and manage the overall QoS of the multimedia application. This is due to the fact that the manager will deal with one metric which is the overall QoS rather than the multiple QoS parameters as it will be discussed in the coming chapters.

QoS provision is a technique that generally consists of: a measure of network/application QoS state and a way to observe it and a heuristic that uses the information to deliver a QoS objective (Stineand and Veciana, 2004). Hence, to provide or guarantee QoS, it should be monitored firstly. For networks and especially in wireless ad hoc QoS provision is not an easy issue. Therefore, many approaches have been proposed. These approaches include call admission protocols that first assess whether a flow should be admitted into the network based on its QoS status (Chiang and Carlsson, 2001), (Dong, et al., 2003), routing protocols that attempt to control the flow of traffic through sections in the network that can best afford it with acceptable QoS (Xue and Ganz, 2003), (Curado and Monteiro, 2001), queuing schemes implemented at nodes (Kanodia, et al., 2002), medium access schemes which give access priorities to some applications to and reserve the Radio Frequency (RF) media (Sheu, et al., 2004), (Holland, et al., 2001). All of these schemes must perform QoS assessment before and after applying the proposed approach to enhance the application/network performance.

In addition to the above, in wireless networks, QoS measurement and monitoring play an important role in supplying a high QoS and in ensuring that the desired QoS properties are attained and sustained. To achieve that, the wireless channel must be kept away from reaching the congestion state. This is because loss and delay increase rapidly once this state is reached. To keep the utilisation below the congestion point is a difficult issue in wireless networks because the channel is shared between the active nodes. Therefore, each node needs to determine the network utilisation which can be inferred from the measured QoS. Once the available resources are determined, nodes can then adapt their data traffic rates to keep the channel from becoming congested.

2.2.3 Quality of Service Parameters

Different multimedia applications have different QoS requirements. The specific parameters which define QoS vary depending on the application and user requirements (Kasigwa, et al., 2004). It is very important to determine the correct set of accurate QoS parameters for the particular media being transported; otherwise QoS guarantees cannot be obtained (Cheong and Lai, 1999). QoS of transmitted application through a network is characterised, in a very general way, by four key network parameters (metrics): one-way delay (Almes, et al., 1999a), one-way jitter (delay variation) (Demichelis and Chimento, 2002), packet loss ratio (Almes, et al., 1999b), and bandwidth. Together, these parameters determine the QoS the traffic requires (Alkahtani, et al., 2003). Our research will concentrate on audio and videoconferencing multimedia applications. In the following, general definitions of the main factors that can profoundly influence the QoS of these applications are explained:

(i) One-way delay

One of the primary QoS parameters for real-time multimedia communication is the one-way delay (OWD). It may be defined as the amount of time taken to transmit a packet and to receive it at the destination. It is also defined as the elapsed time for data to be passed from the sender, through the network, to the receiver (Schmitt, et al., 2002). The majority of the real-time multimedia applications (audio and video flows) are delay sensitive because the information transmitted needs to be replayed at the receiver at real-time. A small average delay is acceptable but a more important delay quantity is the delay bound. The delay bound is the maximum delay experienced by any packet. This bound is variable and depends on the type of the application, for example, non-interactive multimedia like Video on Demand (VoD) may allow a higher delay bound than an interactive one like the videoconferencing applications. It includes all possible delays caused by transmission delay, propagation delay, queuing delay and processing delay. Due to synchronization problems between the clocks of each client, the measurement of the one way delay is a non-trivial task. For exact measurements, it is required that both clocks are highly synchronized. The delay parameter may be calculated as (Wang, et al., 2000):

$$D_i = r_i - s_i \tag{2.1}$$

where D_i is the delay (in seconds) of the i^{th} packet arrived and r_i and s_i are arriving and sending timestamps of the i^{th} packet. And the average end-to-end delay can be calculated as:

$$Average\ delay = \frac{1}{n}\sum_{i=1}^{n} D_i \tag{2.2}$$

where D_i is the packet delay from equation (2.1) and n is number of successfully received packets.

(ii) One-way jitter (Delay variation)[1]

The variation of the inter-arrival time of packets at the receiving site is known as the delay variation, also referred to as the jitter (Demichelis and Chimento, 2002). Therefore, jitter is defined as the difference between the delays of two consecutive packets; therefore, it requires the measurement of one-way delay. The variation in the inter-packet arrival times leads to gaps between two consecutive packets (Agarwal, 2000). This may be caused by the variable transmission delay over the network, variations in queue length or variation of processing time of every received packet. Delay variation has a significant influence on real-time or delay sensitive multimedia applications. The influence of jitter is less for audio than for video in which it causes observable effects on video play and leads to a stuttering with pops and clicks (Schmitt, et al., 2002). Methods to remove this variation require collecting packets in buffers and holding them for an appropriate period. This will allow the slowest packets to arrive in time to be played in a correct sequence. This, however, increases the delay for each packet transiting the network. Jitter can be calculated as (Wang, et al., 2000):

$$J_i = \left| D_i - D_{i-1} \right| \ , \quad i \rangle 0 \tag{2.3}$$

where J_i is the jitter (absolute values in second) of the i^{th} packet, D_i and D_{i-1} are the delays of two consecutive packets computed from equation (2.3). Also, average jitter for traffic flow can be calculated by:

$$Average\ jitter = \frac{1}{n}\sum_{i=1}^{n} J_i \tag{2.4}$$

[1] Jitter and delay variation will be used interchangeably throughout this thesis.

J_i is the packet jitter from equation (2.1) and n is number of successfully received packets.

(iii) Packet loss ratio

This may be defined as the percentage of packets discarded by the node or the router. It includes packet losses and out of order packets. Packet losses are due to error introduced by the physical transmission medium or due to congestion periods. But in the wireless, it is due to link errors between the two endpoints like interference, link failure, handoffs…etc, or due to collisions between packets or due to buffer overflow. This will directly affect the application quality at the receiver. The degree of degradation depends upon the type of application (Flup, 1999). In order to measure the packet loss ratio, a packet stream which includes sequence numbers is required. The percentage packet losses can be calculated as (Wang, et al., 2000):

$$L_i(t) = \left(1 - \sum R_i(t) / \sum S_i(t)\right) \times 100 \tag{2.5}$$

where L_i is the loss ratio (in %) during the i^{th} interval and $\sum R_i(t)$ and $\sum S_i(t)$ are the total number of received and transmitted packets with the i^{th} interval, respectively.

(iv) Throughput

This parameter offers the rate at which the traffic can flow through the network. Therefore, it is a measure of the capability of that network to transmit an application. It may be defined as the maximum data transfer rate that can be sustained between two endpoints for an application's traffic to be carried by the network. Bandwidth of the channel is the parameter which affects the amount of throughput given to a specific traffic. The average throughput may be calculated as the amount of data received by the destination divided by the measured time (Wang, et al., 2000):

$$T_i(t) = \sum P_i(t) / t_i \tag{2.6}$$

where T_i is the throughput (bits/s or bps) during the i^{th} interval, $\sum P_i(t)$ is the total bits of all received packets within the i^{th} interval, and t_i is the time duration of the i^{th} interval.

2.2.4 Multimedia Applications and their QoS Requirements

Multimedia applications incorporate various media such as, voice, video and data information. Multimedia may be defined in several ways. Marshall (2003) gives two definitions. These are:

- Multimedia means that computer information can be represented through audio, video, and animation in addition to traditional media (i.e., text, graphics drawings, and images).
- Multimedia is the field concerned with computer-controlled integration of text, graphics, drawings, still and moving images (video), animation, audio, and any other media in which every type of information can be represented, stored, transmitted and processed digitally.

In order to analyse the QoS of a particular application, the main QoS parameters have to be defined and explained. For example, real-time multimedia applications depend predominantly on delay, jitter and packet loss parameters of a transmission. Above all, the one-way delay is important in multimedia environments. Streaming video and audio transmissions need a low variation of delay (jitter), and nearly each application depends on a low packet loss ratio. Table 2.1 shows some of the common multimedia applications and their QoS parameter's requirements (Alkahtani, et al., 2003). Figure 2.1, also, illustrates the relative requirements of some multimedia applications with regard to error and delay requirements (Converdale, 2003).

Table 2.1: Some multimedia applications and the sensitivity of their QoS requirements.

Applications		Sensitivity			
		Loss	Delay	Jitter	Bandwidth
Data Traffic	E-mail	High	Low	Low	Low
	Confidential e-mail	High	Low	Low	Low
	File transfer	High	Low	Low	Low, Medium, High
	Money transactions	High	Low	Low	Low
Real-Time traffic	Audio on demand	Low	Low	High	Medium
	Video on demand	Low	Low	High	High
	Telephony	Low	Low	High	High
	Videoconferencing	Low	High	High	High
	Confidential Videoconferencing	Low	High	High	High

Error tolerant	**Conversational voice and video**	**Vo ce messag ng**	**Stream ng aud o and v deo**	**Fax**
Error intolerant	**Telnet, interactive games**	**E-commerce, WWW browsing.**	**FTP still image, paging**	**E-nai ar va notification**
	Conversational (delay « 1 sec)	**Interactive (delay approx 1 sec)**	**Streaming (delay <10 sec)**	**Background (delay >10 sec)**

Figure 2.1: Application qualitative QoS requirements.

QoS monitoring and measurement allow the network administrator to use the existing resources efficiently and to guarantee that critical applications receive high service quality without having to expand their networks. To achieve maximum utilisation of the network resources, several network QoS classes to carry traffic which have broadly similar QoS requirements have been proposed. Table 2.2 gives the ITU recommendations of these classes (Converdale, 2003).

Table 2.2: ITU QoS classes.

	QoS Classes					
Network Performance Parameter	Class 0	Class 1	Class 2	Class 3	Class 4	Class 5
Transfer Delay [msec]	100	400	100	400	1000	Unspecified
Delay Variation [msec]	50	50	Unspecified			Unspecified
Packet Loss ratio [%]	1*10'3					Unspecified
Packet Error Ratio	1*10'4					Unspecified

2.2.5 Service Levels of QoS

As discussed earlier, every QoS parameter may be represented by a range of values expressing maximum, average and minimum requirements. Service level is the actual QoS capability of a network to deliver service required by a specific application. QoS can provide three basic levels of agreements, which a user may request from end-to-end:

best effort, compulsory (differentiated), and guaranteed (Shah, 2001) (Schmitt, et al., 2002).

(i) Best effort service: also known as lack of QoS with no priority or guarantees. Example of this type is the service provided by the Internet to the application transmission. That is because the network accepts all requests for service and tries to deliver the data packets with no admission or flow control by hosts. In addition, when the router buffers become full, all connections through that router suffer packets loss or queuing delay.

(ii) Compulsory (Differentiated) service: this service treats some traffic better than the others with no hard or soft guarantee. Differentiated services are associated with a course level of packet classification. This means that the traffic gets grouped or aggregated into a small number of classes with each class receiving a particular QoS in the network.

(iii) Guaranteed service (Integrated Service (IS)): it is based on a reservation of network resources for a specific application. This involves reservation of bandwidth and buffer space along with suitable queuing algorithms to insure that a specific application gets a specific service level. This is achieved by allowing sources to communicate their QoS requirements to router and destinations on the data path by means of a signalling protocol such as Resource Reservation Protocol (RSVP) (Nikaein and Bonnet, 2002). Therefore, it provides per-flow end-to-end QoS guarantees. This type of service is applicable, for example, to voice and video applications because they are delay sensitive traffics. So, a guaranteed service level is intended for applications requiring a fixed delay bounds. A graphical representation of these levels is shown in Figure 2.2.

To make multimedia data transmission efficient and to offer a good user-perceived QoS, the multimedia applications must adapt to network changing conditions like losses, bandwidth abrupt changes and delay variations. Therefore, these applications must take advantage of QoS and network status information like packet losses, delay variations and available bandwidth.

QoS

Best

Guarantee low loss and
small delay

Guarantee bounded loss
and delay

Best Effort

Worst

Figure 2.2: Levels of QoS.

2.3 Wireless Networks

2.3.1 Introduction

Wireless communications have grown rapidly in the 20th century. In the 1970s, Pr. Norman Abrahamson wanted to radio-connect his university's computers located at different islands with a protocol called Aloha (Aad, 2002). The development of the Wireless Local Area Network (WLAN) started in 1991, and the first WLAN standard was created and adopted by IEEE, named as IEEE 802.11, in 1997. This new technology had a great success due to functions it provides: it complemented the widely deployed Ethernet with data rate up to 2 Mb/s. In 1999, this was followed by the completion of standards for 802.1 la and 802.1 lb, and most recently, 802.1 lg in 2003.

Nowadays, more wireless standards have been developed, data rates are becoming higher and services are becoming richer. IEEE wireless standard devices are making proliferation change in our daily life and information society. They provide numerous facilities to users where these devices can be used at the home, the office, the road, etc. Wireless devices have been become an essential feature of every day life in social, medical, industrial and military fields.

2.3.2 Radio Environment

Wireless communications differ from wired communications by the fact that the electromagnetic wave propagates in the free space instead of inside cables. Therefore, many issues emerge from this fact such as multipath, path loss attenuation, and noise and interference on the channel, making the radio channel a hostile medium in which behaviour is difficult to be predicted.

Wired communication media are usually protected against external noise sources. However, this protection does not exist in wireless communications due to radio transmissions by other stations using/interfering with the same frequency band. This will result in a wireless communications medium, which is much less deterministic and more erroneous than its wired counterpart. In wired networks, typical Bit Error Rates (BERs) are relatively very small, i.e. in order of 10^{-6} (Aad, 2002). In contrast, BERs in wireless channels are in the order of 10^{-3} and usually occurs in bursts.

2.3.3 Working Modes of Wireless Networks

An IEEE 802.11 WLAN generally consists of Basic Service Sets (BSSs) which are interconnected by Distributed System (DS) to form an Extended Service Set (ESS) (Anastasi and Lenzini, 2000). Each BSS consists of a group of wireless terminals (stations) and the area it covers is called Basic Service Area (BSA). A BSS can operate in two modes: an infrastructure-based mode in which an Access Point (AP) links the stations to the DS and infrastructureless-based or ad hoc mode which also may be called Independent BSS (IBSS).

A BSS that includes an AP within its stations can be connected to wired LAN as shown in Figure 2.3(a). All communications within a BSS go through the AP. If any two stations in that BSS want communicate with each other, frames are first sent to the AP then to the destination.

In contrast, any station in the ad hoc mode that is within the transmission range of any other can start communicating after a synchronisation phase (Chakrabarti and Mishra, 2001). In ad hoc mode, no AP is needed, however, if one of the IBSS stations has a connection to a wired network LAN, all stations that are in the receiving range of this station can gain a wireless access to the Internet. This structure is shown in Figure 2.3(b).

Wireless
station

Wireless
station

IBSS

(a) (b)

Figure 2.3: (a) Infrastructure wireless network and (b) Ad hoc network.

2.3.4 Wireless LAN

WLANs are an essential part of wireless communications. This is not only because they provide wireless connections to devices using the network directly but also because they provide means to carry data belonging to other networks. WLAN networks are based on standards, which are provided mostly by two big standardisation parties: the European Telecommunications Standards Institute (ETSI) and Institute of Electrical and Electronics Engineering (IEEE) (Syrjala, 2003).

Table 2.3: Performance overview of IEEE & ETSI wireless LAN Standards.

Standard	Frequency (GHz)	Physical speed [Mb/s]	Range
IEEE802.il	2.4	2	150 m
IEEE 802.11b	2.4	11	150 m
IEEE 802.11a	5	54	150 m
IEEE 802.llg	2.4	54	150 m
ETSI HIPERLAN/1	5	23.5	150 m
ETSI HIPERLAN/2	5	54	30-200 m
ETSI HIPERACCESS	40-43.5	25	<5 Km
ETSI HIPERLINK	17	155	150 m

As the IEEE 802.11 standard is the first standard that has been developed by IEEE task group, it is currently the most successful WLAN standard (Hannikainen, et al., 2002). Because WLANs became more popular, new demands were placed on them. One of these is higher bandwidth, especially when the network has several users. Due to this, faster WLANs solutions have been developed. The characteristics of the WLAN standards are summarised in Table 2.3 (Syrjala, 2003).

Our research will focus on the IEEE standards. IEEE 802.11 standard covers three types of physical layer and Medium Access Control (MAC) sub-layer.

2.3.4.1 The Physical Layer (PHY)

PHY layer is the lowest layer in the Open System Interface (OSI) model. It deals with the details involved in the actual radio transmission. This layer consists of Physical Layer Convergence Procedure sub-layer (PLCP) and Physical Medium Dependent sub-layer (PMD). The PLCP sub-layer is responsible for controlling the frame exchange between the PHY layer and the Medium Access Control (MAC) layer. While the PMD sub-layer controls the carrier and the spread spectrum techniques used to transmit the data over the wireless media. The IEEE 802.11 radios operate in the 2.4 GHz Industrial Science Medical (ISM) range and use spread spectrum techniques to spread the radiated power over the allowed frequency spectrum. Spread spectrum has multiple access capability, protection against multi-path interference, privacy, and anti jamming capability (Aad, 2002). In IEEE 802.11 three physical layers were specified:

- Frequency Hopping Spread Spectrum (FHSS): before transmitting, 802.11 modulates the signal by means of Frequency Shift Keying (FSK). In FHSS, the available frequency is divided into 1MHz wide, non-overlapping channels to give 75 or more channels (Lo and Ngai, 2004). The transmission of the signal is achieved across a group of frequency channels by hopping from one carrier frequency to another after a dwell time (Sweet, et al., 1999). The spreading code defines the frequency at which data bits are transmitted. Both sender and receiver should synchronously hop using the same frequency hop pattern in order to communicate.

- Direct Sequence Spread Spectrum (DSSS): DSSS generates a wide bandwidth signal and spread over the width of one channel. Each channel is 22 MHz wide with 5 MHz separation between centre frequencies. Instead of sending raw data bits, DSSS correlates data with the code chips running at higher rate (Aad, 2002). The code used is an 11-chip known sequence called Barker code (Celebi, 2002). The resulting high rate stream is modulated using the base-band modulation techniques (Binary Phase Shift Keying (BPSK) or Quaternary Phase Shift Keying (QPSK)) and transmitted in free-space. At the receiver side, the reverse procedure is applied to retrieve the original data.

- Infra Red (IR): data bits are modulated using Pulse Position Modulation (PPM) and transmitted using near visible light (800-950 nanometre). Since infrared requires line-of-sight communications, it is not widely used.

2.3.4.2 The MAC Sub-layer

The MAC provides the following functionalities:

- Reliable data delivery over the wireless medium.
- A fair regulation of accessing the wireless channel using two different methods Distributed Coordination Function (DCF) and Point Coordination Function (PCF).

The MAC protocol is concerned with per-link communications and not end-to-end. IEEE 802.11 standards MAC protocol provides two modes of operation as mentioned before: DCF and PCF.

2.3.4.2.1 Distributed Coordination Function (DCF)

In DCF mode, each station gets an equal share of the channel through contention, i.e. a station contends for the channel use before each frame waiting for transmission. The basic scheme for DCF is based on the Carrier Sense Multiple Access (CSMA) (Landfeld, 1999), (Kleinrock and Tobagi, 1975) in which carrier sense means that the station will listen before it transmits, i.e., the station must sense the channel before trying to transmit their data. CSMA protocol has two types: Collision Detection (CSMA/CD) and Collision Avoidance (CSMA/CA). A collision can be caused by two or more stations using the same channel at the same time. It also can be caused by two or more hidden terminals transmitting simultaneously. Hidden terminals are terminals which cannot hear each other (Khurana, et al., 1998).

CSMA/CD is used in Ethernet wired networks to abort transmission when a node detects that the signal it is transmitting is different from the one on the channel due to collision. This does not exist in wireless communications because the station cannot listen to the channel while it is transmitting. This is because of the big difference between transmitted and received power levels. To deal with this problem, DCF employs two handshaking techniques for packet transmission. The default scheme is two-way handshaking technique called basic access mechanism. The other is optional Request to Send/Clear to Send (RTS/CTS) four-way handshaking mechanism used to combat the effect of collisions for data packets. These two mechanisms are shown in Figures 2.4 and 2.5, respectively.

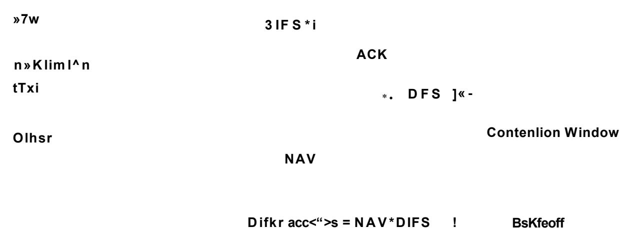


Figure 2.4: Basic access mechanism.

Using the basic mechanism, every station in the system that wishes to transmit data, waits for a DIFS (DCF Inter-frame Space) period of time, this executes a random back-off algorithm. If at the expiration of the back-off timer, the medium is still idle, the source transmits data. If the packet is successfully received by the destination, after a period of SIFS (Short Inter Frame Spacing), the destination sends an acknowledgement (ACK). Meanwhile, other nodes in the system read the duration field in the header of the data frame, and update their Network Allocation Vector (NAV) with this value as shown in Figure 2.4.

For transmission with RTS/CTS, instead of transmitting data, the station transmits an RTS frame. If it is successfully received, after a period of SIFS seconds, the destination transmits a CTS frame, which follows the data transfer as shown in Figure 2.5. Also, each node updates its NAV for each, RTS, CTS and data frame. This mechanism is used to avoid collisions with hidden nodes, when the RTS and data frames cannot be heard by other stations. In addition, this handshake is recommended to be used when the size of the MAC layer Protocol Data Unit (MPDU) is large and greater than RTS threshold, to prevent channel bandwidth wastage in case of collision of MPDUs in the medium.

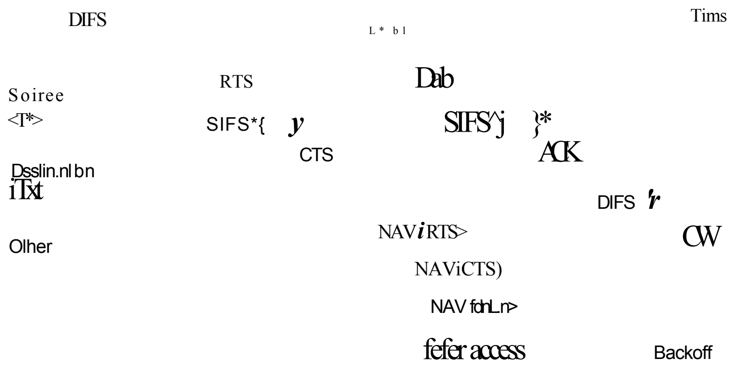


Figure 2.5: RTS/CTS access mechanism.

2.3.4.2.2 Point Coordination Function

PCF is connection oriented and provides contention free frame transfer based on poll and response protocol (Brian, et al., 1997). PCF can be used in ad hoc networks, and can be used with the DCF in an infrastructure network. In PCF, an Access Point (AP) is required to poll each station in the BSS to enable it to transmit without contending with others to access the medium. The AP through a beacon frame initiates contention free period repetition interval, which consists of Contention Free Period (CFP) and Contention Period (CP). The data transmission is quite simple. Whenever the AP finds the medium idle, it waits for a PIFS (PCF Inter Frame Space) period and then starts transmission of the beacon frame with a polling frame following SIFS period. When a station receives the poll from the AP, it reserves the medium for the duration of its transfer, and can transmit data to another station in the network. When the data transfer is finished, the AP waits for PIFS and starts polling another station. The AP can also transmit data along with the polling frame, which can lead to better utilization of the bandwidth. After the completion of the CFP, CP begins and each station then contends for the medium as discussed in DCF.

2.3.5 Attributes of Wireless Networks

Ideally, the users of wireless networks request the same services and capabilities that they have commonly used to obtain using wired networks. In contrast to the wired networks, wireless networks have some special requirements that are unique to their form of communications (Sweet, et al., 1999):

- Throughput: the WLAN capacity should be increased to meet the high demand of multimedia transmissions. Moreover, due to the bandwidth shortage in these networks, the MAC protocol should make as efficient use of this available bandwidth as possible based on techniques like service differentiation, call admission controls... etc.

- Mobility: unlike wired terminals, wireless terminals should be able to move freely in their BSS. Therefore, the system designs must accommodate handoffs between the transmission boundaries and route traffic to these terminals (Brian, et al., 1997). In addition to that, wireless users should ideally not be affected by the addition, deletion, or relocation of other wireless stations.
- Power considerations: wireless stations are typically small battery powered. Therefore, devices must be designed to be very energy-efficient which results in sleep modes and low power displays.
- Security: a wireless network is difficult to be secure, since the transmission media is open. Encryption is one solution to that.
- Interference: this is due to simultaneous transmissions by two or more sources sharing the same frequency band, which will result in collisions.

2.3.6 QoS in Wireless Networks

The evolution of wireless networks and real-time multimedia applications introduces new challenges in supporting predictable and reliable communication performance. These challenges are a consequence of the vastly increasing number of current and future multimedia products that find application not only in wired networks but also in the wireless environment and hence require special attention.

QoS in WLANs has been an area of interests since WLAN became available. Providing QoS, other than best effort, is a very complex problem especially in wireless ad hoc networks. The nature of WLANs and the network ability to provide QoS depends on the intrinsic characteristics of all the network components, from transmission links to the MAC and network layers (Chakrabarti and Mishra, 2001), (Macker and Courson, 2003). In wireless networks, the ability to provide QoS guarantees is weak, because wireless links have variable capacity, high loss rates, and high latency. In addition, the weakness is due to dynamic nature of the stations topologies, which will result in high frequent links breakages. Furthermore, the service quality of the network varies with time depending on the resource availability in the wireless medium and in the nodes: e.g., buffer and battery.

In wireless networks which use the IEEE 802.11 standard and DCF as the medium access scheme, most problems with real-time applications like audio and video services are mainly related to a trade-off between the typical QoS parameters. If a packet does not reach its destination correctly on the first attempt of sending, the data link layer of 802.11 will retransmit the packet up to a certain amount of times. If delivery still fails, it depends on the higher-layer protocols if the packet is dropped (i.e. UDP) or if retransmissions will be done (i.e. TCP). In the case of TCP, retransmissions are attempted until the packet is either delivered correctly, or until the TCP connection is dropped. In the case of UDP, the packet is dropped and delivery of the next packet is attempted.

The QoS that the network can support is not related to any dedicated network layer, instead it may require coordinated efforts from all layers. However, to accomplish real-time needs, only lower-layer protocols can be used or enhanced, since only they have the control over the resources, which influence delivery timing. For error control, also higher-layer protocols can be used. Important QoS components include QoS MAC, QoS routing, Call Admission Control (CAC), and resource reservation signalling.

QoS MAC protocols proposed mechanisms for medium accessing and contention provide reliable unicast communications and support resource reservation for real-time multimedia applications (Qiang, et al., 2004), (Romaszko and Blondia, 2004).

Another approach is to implement QoS routing which refers to the discovery and maintenance of routes that can satisfy QoS requirements under given resources constraints such as Robust Quality of Service Routing Protocol (RQoSR) (Ayyash, 2005) (Ayyash, et al., 2006). The main objectives of QoS routing can be summarised as (Celebi, 2001):

(i) Dynamic determination of feasible paths;
(ii) Optimisation of resource reservation; and
(iii) Graceful degradation in the network performance as opposed to a dramatic degradation such as in best-effort routing.

QoS signalling is responsible for resource reservation and admission control along the route determined. CAC is one of the important mechanisms that can be used to control the QoS provided by the network (Bianchi, et al., 2000), (Valaee and Li, 2002). The

CAC technique is used to determine if a new flow should be admitted into the network. The acceptance or denial of new flow depends on the availability of the network resources for the requested flow QoS. If a new traffic or call is accepted without a particular limit, QoS for other traffics in progress may be degraded below an acceptable level. This is due to total bandwidth required for all traffic which exceeds the network capacity. Therefore, the acceptance of a new application is performed by the CAC depending upon two main factors: the status of the network resources and the level of service called by the new application request. From this, it can be said that admission control is a key component of QoS-based resource management schemes.

RFC[1] 2389 characterises QoS as a set of services requirements to be met by the network while transporting a packet stream from source to destination (Chakrabarti and Mishra, 2001). The network must provide QoS to guarantee an acceptable set of measurable service attributes to the user in terms of delay, jitter, throughput, available bandwidth, losses... etc. According to (Toh, et al., 2002), in order to evaluate the communication performance of a wireless network, a number of measurements of these parameters have to be taken into account under varying conditions.

Due to contention-based channel access of an IEEE 802.11 network and depending on the traffic load in the network, an IEEE 802.11 network can be in one of three states: saturated, non-saturated or semi-saturated (Yang, et al., 2003). A saturated state means that every station in the network always has a packet to be sent which means that all the stations are overloaded. A network is in a non-saturated state when no station has a packet to be sent and their queues are mostly empty. A semi-saturated network is between the saturated state and the non-saturated state, where some stations are mostly overloaded and their queues are usually full while others are lightly loaded and their queues are often empty. Each state has its own reflection impact on the traffic performance/QoS over the network. These states and its influence on the multimedia transmissions will be studied in Chapter 4. For example, it is essential to determine the number of simultaneous audio or video applications a wireless network can support for a given state.

---

[1] Request for comment (RFC) documents are originally Internet drafts. Developing these drafts is the primary task of the Internet Engineering Task Force (IETF).

2.4 Fuzzy Logic Theory

2.4.1 Definition

Fuzzy logic is a powerful tool for decision-making involving information characterised by imprecision and uncertainties. It was created to account for smooth transitions that exist between true and false in many applications. Fuzzy set theory was first proposed by Lotfi A. Zadeh in 1965 (Zadeh, 1965).

Fuzzy logic is an excellent problem-solving mechanism with numerous applications in artificial intelligence, embedded control and information processing. It provides a remarkably simple way to draw definite conclusions from vague, ambiguous or imprecise information (Hudson and Cohen, 2000). Unlike the classical logic, which requires a deep understanding of a system, exact equations, and precise numeric values, fuzzy logic incorporates an alternative way of thinking, which allows modelling complex systems using a higher level of abstraction originating from our knowledge and experience (Kuncheva and Steimann, 1999). Fuzzy logic emerged into the mainstream of information technology in the late 1980's and early 1990's (Sadegh, 2001). It has been used as a tool to evaluate some characteristics of networking systems. Two studies are closely related to this work (i.e. used fuzzy logic); these studies are (Saraireh, 2003) and (Aboelela, 1998).

2.4.2 Fuzzy Inference Systems

Fuzzy logic provides a mechanism for handling uncertainties and nonlinearities that exist in physical systems. It is based on fuzzy sets, which are the generalisation of crisp sets. A general fuzzy logic based inference system is shown in Figure 2.6. It comprises of four main components: a fuzzification element, an inference system, rule base and a defuzzifier (Ross, 2004). The functionality and role of each component will be briefly described in the next sections.

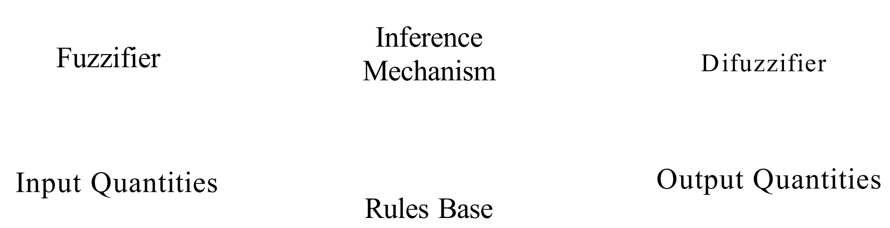


Figure 2.6: Block diagram of a fuzzy inference system.

2.4.2.1 Fuzzification

Fuzzification is the process by which the crisp input variable of the fuzzy system is converted into appropriate linguistic terms for the fuzzy logic processing. It is the process of taking actual real-world data (such as temperature) and converting them into a fuzzy input (Terano, 1992). This produces multiple fuzzy inputs for every real-input value, using a number of membership functions (We and Chen, 1999). Membership functions are graphical representations of the confidence interval that the designer has with respect to a fuzzy input. It is used to combine multiple subjective categories describing the same context. A Boolean membership function for an element is either one or zero as shown in Figure 2.7a, i.e. elements U either belong to the set (i.e. to a number in the interval [0,1]) or not. However, a fuzzy set is characterised by the membership function $\mu(x)$ as shown in Figure 2.7b.

The membership function allows gradual transition from full-belonging to the fuzzy set ($\mu(x) = 1$) to not-belonging at all ($\mu(x) = 0$) with intermediate values presenting degrees of belonging to the fuzzy set. In fuzzy logic, an element can reside in more than one set with different degrees of membership as illustrated in Figure 2.7b. Therefore, if the fuzzy set presents a concept, the value of the membership function will present the degree of fulfilment to this concept, which is a feature not available in classical set theory. Intuitively, a fuzzy set is a class that admits the possibility of partial membership in it.

Different shapes of membership functions can be used in modelling linguistic terms. This includes triangular, trapezoidal, bell-shaped, sigmoid, crisp, singleton etc. The main parameters, which characterise the membership function, are:

- **Peak value/interval** is the point/interval at which the degree of membership in a fuzzy set is maximum: $\mu(x_{peak}) = 1$.
- **The left width** of a membership function is the interval from the peak/interval value to the left point where the degree of membership function is zero. Similarly, the **right width** is the interval from the peak value to the right point where the degree of membership function is zero.
- **The crossover point** is the point at which two neighbouring membership functions cross. At the cross point, the degree of membership to both sets are equal and greater than zero.

Ciisp Sets

A A A A

Set theory

Fuzzy set theory

0.75 t

0.25 ◁

XI X2

(a) **(b)**

Figure 2.7: Membership functions (a) Boolean, (b) Fuzzy.

Fuzzy logic starts with and builds on a set of user-supplied human language rules. The fuzzy systems convert these rules to their mathematical equivalents. This simplifies the job of the system designer and the computer, and results in much more accurate representations of the way systems behave in the real world. Fuzzy logic models, called fuzzy inference systems, consist of a number of conditional “If-Then” rules. For the designer who understands the system, these rules are easy to write, and as many rules as necessary can be supplied to describe the system adequately (although typically only a moderate number of rules are needed) (Hudson and Cohen, 2000).

2.4.2.2 Fuzzy Rule and Inference Engine

The fuzzy rules are mainly defined on the basis of the observed features of the input data (Oliveira and Braum, 2004). In addition, the selection of rule-base relies on the designer's experience and beliefs of how the system should behave (Pitsillides and Sekercioglu, 1999). The rule-base component contains a set of “If-Then” rules that is the basis for the decision making process of the inference mechanism. The number of rules in a fuzzy system depends on both the number of input variables and membership functions associated with them.

Based on the defined membership functions, a set of IF-THEN type rules can be formulated. These rules and the corresponding membership functions are employed to analyse the system's inputs and determine its outputs by the process of fuzzy logic inference. In fuzzy inference systems, the output is defined by using statements of the form:

IF (Antecedent1) AND (Antecedent 2) ... THEN (Consequent)

where, the Antecedent relates the linguistic term to a fuzzy set and the Consequent represents the conclusion for the IF term. For example: IF (packet loss is high) and (delay is high) then (QoS is poor).

Since all the input values have been transferred into linguistic values, certain rules will be identified or fired. These rules are identified in order to calculate the values of the linguistic output variable. The fuzzy inference consists of two components: The first step is to determine which sets of rules apply to the current situation. The second step is to determine what conclusion should be reached. There is one conclusion for every rule that is "active". Each conclusion is highly dependent on the choice of the membership functions, antecedents of the rules and the inputs to the inference system.

The conventional linguistic operators used for two-valued logic are not applicable with fuzzy set. Given that μ_A and μ_B are degrees of memberships for the sets A and B respectively, then different fuzzy operators can be defined as follows (Mathworks, 2005),

$$\begin{aligned} AND: &\quad \mu_{A \bullet B} = \min(\mu_A, \mu_B) \\ OR: &\quad \mu_{A+B} = \max(\mu_A, \mu_B) \end{aligned} \tag{2.7}$$

The degree of truth of the IF condition is calculated using the linguistic operator to indicate how adequately each rule describes the current situation. More than one rule might be triggered simultaneously describing the current situation. Each of these rules defines an action (Consequent) to be taken in the THEN condition. This is done using the implication method. This method is defined as the shaping of the output membership functions on the basis of the firing strength of the rule. The input to the implication process is a single number given by the Antecedent, and the output is a fuzzy set. The degree to which the consequent is valid is given by the adequateness of the rule to the current situation. This adequateness is calculated by the aggregation stage as the degree of truth of the IF condition. Aggregation is a process whereby the outputs of each rule are unified. The input to the aggregation process is the truncated output fuzzy sets that returned by the implication process for each rule. The output of the aggregation process is the combined output fuzzy set.

Fuzzy inference system is the process of formulating the mapping from a given input to output by using fuzzy logic. There are two fuzzy inference methods: Mamdani and Sugeno inference methods. Mamdani's method expects the output membership functions to be fuzzy sets, after the composition process; there is a fuzzy set for each output variable that needs defuzzification. Sugeno is similar to the Mamdani method in many respects. In fact the first two parts of the fuzzy inference process, fuzzifying the inputs and applying the fuzzy operator, are the same. The main difference between Mamdani-type of fuzzy inference and Sugeno-type is that the output membership functions are only linear or constant for Sugeno-type fuzzy inference (Mathworks, 2005).

2.4.3 Defuzzification

Defuzzification is the process of converting the linguistic value of the output variable (the aggregation output fuzzy set) into a real (crisp) value by using a defuzzification method such as the centroid, bisector, middle of maximum (the average of the maximum value of the output set), largest of maximum, and smallest of maximum (Ross, 2004).

The most common defuzzification method is centroid. With this, the defuzzified values tend to move smoothly around the output fuzzy region (Fuzzy, 2005). In the centroid method, the real value of the output variable is computed by finding the variable value of the centre of gravity of the membership function for the fuzzy value (i.e., it returns the centre of area under the curve of the aggregated output values as shown in equation 2.8 (Ross, 2004).

$$Y = \frac{\sum_{i=1}^{m} y_i \times \mu_i}{\sum_{i=1}^{m} \mu_i} \tag{2.8}$$

where m represents the number of output fuzzy sets obtained after implication, y_i represents the centroid of fuzzy region i (i.e., the output universe of discourse) and μ_i is the output membership value.

In the maximum method, one of the variable values, which the fuzzy subset has its maximum truth-value, is chosen as the real value for the output variable.

2.5 Distance Measure Theory

Similarity is a quantity that reflects the strength of relationship between two objects or two features. In other words, it is a numerical measure of how alike two data objects are. If the similarity between features i and j is denoted by S_{ij}, we can measure this quantity in several ways depending on data type that we have. Distance measures the dissimilarity between two objects. It measure the discrepancy between the two objects based on several features. These features can be represented as coordinate of the object in the features space. There are many types of distance calculation techniques that can be used to measure this dissimilarity. Let the normalized dissimilarity between objects i and j be denoted by d_{ij}. The relationship between dissimilarity and similarity is given by,

$$S_{ij} = 1 - d_{ij} \tag{2.9}$$

In general, the distance, d_{ij} is a quantitative variable, which will satisfy the following conditions (Teknomo, 2006):

(i) $d_{ij} \geq 0$: distance is always positive or zero

(ii) $d_{ij} = 0$: distance is zero if and only if it measured to itself

(iii) $d_{ij} = d_{ji}$: distance is symmetry

(iv) $d_{ij} \leq d_{ik} + d_{kj}$: distance satisfy triangular inequality

Dissimilarity is usually measured by Euclidean distance and Minkowski distance. In addition, Euclidean distance is the usual use of distance measure (Teknomo, 2006). Euclidean distance or simply 'distance' evaluates the root of square differences between coordinates of a pair of objects.

$$d_{ij} = \sqrt{\sum_{k=1}^{n} \left(x_{ik} - x_{jk}\right)^2} \tag{2.10}$$

where k is the index of the object's coordinates, x_i and x_j are coordinates of the objects.

The Minkowski metric is widely used for measuring similarity between objects (e.g., images) (Li, et al., 2002). Minkowski distance is the generalised distance as can be seen in equation (2.11) (Batchelor, 1978). It is a formula derived from Pythagoras metric. This distance can be used for both ordinal and quantitative variables (Teknomo, 2006). The Minkowski distance between two vectors may be defined as the geometric distance between two inputs with a variable scaling factor, power (λ). When this value is one, the Minkowski distance is equal to the Manhattan distance. When λ is two it yields the Euclidian distance between two vectors. Thus, by increasing the power, one can place more numerical value on the largest distance (in terms of elements in the two vectors in question). A disadvantage of the Minkowski method is that if one element in the vectors has a wider range than the other elements, the larger range may then 'dilute' the distances of the small-range elements. This disadvantage will be overcome using the normalisation technique which will be discussed in Chapter 5.

$$d_{ij} = \sqrt[\lambda]{\sum_{k=1}^{n} \left(x_{ik} - x_{jk}\right)^\lambda} \tag{2.11}$$

2.6 Summary

This chapter provided the relevant theoretical background needed to support this thesis. This included an overview and general descriptions about the QoS definition, parameters and general QoS requirements and levels of multimedia applications. In addition to the QoS, this chapter outlined the wireless networks and the QoS aspects related to this type of networks when they are used to transmit multimedia applications. The basics of the fuzzy logic and the distance measure approaches theory which are used as a tool used for assessment and evaluation purposes were also explained. The next chapter will present the state-of-art of the assessment and measurement methods used to determine and evaluate the QoS of the multimedia applications.

QoS Assessment Methods: State of Art

3.1 Introduction

The use of networking systems is becoming a dominant factor in bringing information to users. As a result, the user requirements and attitudes have changed, demanding QoS levels other than the conventional Internet best-effort service. Implementing communication service levels that are higher than the best-effort level requires the measurement of network characteristics before any new transmission. Measurement techniques are traditionally used in telecommunications networks to support a wide range of activities including network planning and design, network operation and research (Pasztor and Veitch, 2001). Measuring packet-switched network performance is a new research area where the first considerable work was performed by Paxson (1997) in the mid 1990s (Michaut and Lepage, 2005).

In this chapter, the theoretical background and related work relevant to the network performance assessment and measurement techniques are provided. The organisation of this chapter is as follows: Section 3.2 outlines the classification of these techniques. Section 3.3 describes theses assessment methods and the related previous work. Lastly, in Section 3.4, a summary of this chapter is provided.

3.2 Assessment Methods Classification

Many real-time multimedia applications over the Internet have appeared today. These include audio, video phones, videoconferencing, video streaming, telemedical applications, distance learning, etc., with diverse requirements for their perceived quality. This gives rise to a need for assessing the quality of the transmitted applications in real time. The need to measure and assess the QoS is a fundamental requirement in modern communications systems for technical and commercial reasons (Sun, 2004). There has been a surge in the efforts for concentrating on QoS issues of these applications. The interests and emphasis have been on the QoS at the network level and on the end-user's point of view. Currently, there is no standard for the QoS performance measurement; hence, various methods are used. These measurement and assessment techniques may be classified in different ways.

One type of classification is the distinction between direct and indirect measurements. Indirect measurement methods are based on network models and assumptions, where direct measurement methods do not rely on any models or expected behaviours but only on direct traffic observations at several points within the architecture.

Another classification of measurement methods is by the distinction between real time and non-real time methods. Real time methods collect traffic data and packet events as they happen and some of them may be able to display the traffic information as it happens. In contrast, in non-real time measurement methods, the collected traffic data is analysed off-line (later) and may only be a subset (sample) of the total traffic population.

Multimedia quality measurement may also be classified and carried out using two broad techniques: subjective and objective approaches. Generally, subjective tests of multimedia quality are based on evaluations made by human subjects under well defined and controlled conditions therefore; the reference is the end user judgement which is directly captured using this approach. While, the objective methods measure the quality based on mathematical analysis that compare original and distorted multimedia signals.

In addition, the evaluation methods may also be classified in terms of passive and active measurement methods. Passive measurement methods collect information from the ongoing traffic and the results are taken directly after some calculations without disturbing network operation or interfering with operational network traffic. On the other hand, active measurement methods inject measurement traffic (probe) into the network and use the measurements to determine the performance of the application/network.

Generally, the last two classifications (i.e., subjective/objective and passive/active) are the most popular techniques used for the purpose of QoS/performance of multimedia application evaluation. Moreover, these two classifications may be combined and classified into another categorisation. This categorisation is based on whether it is interfering with the network performance (i.e. intrusive or non-intrusive) as shown in Figure 3.1. This classification is the most suitable for this work, so a detailed description of these methods will be presented in the following sections.

Depending on the measurement method used and the inputs of the measurement unit, there are three categories of measurement as illustrated in Figure 3.1 (Sun, 2004):

- Signal-based methods where the inputs are the single-end degraded signals (like audio signals),
- Parameter-based methods by which the inputs are the measured network QoS parameters of the multimedia application (like delay, jitter... etc).
- Comparison-based which involves comparison of the reference and the degraded signals to obtain a score about the application quality over the network.

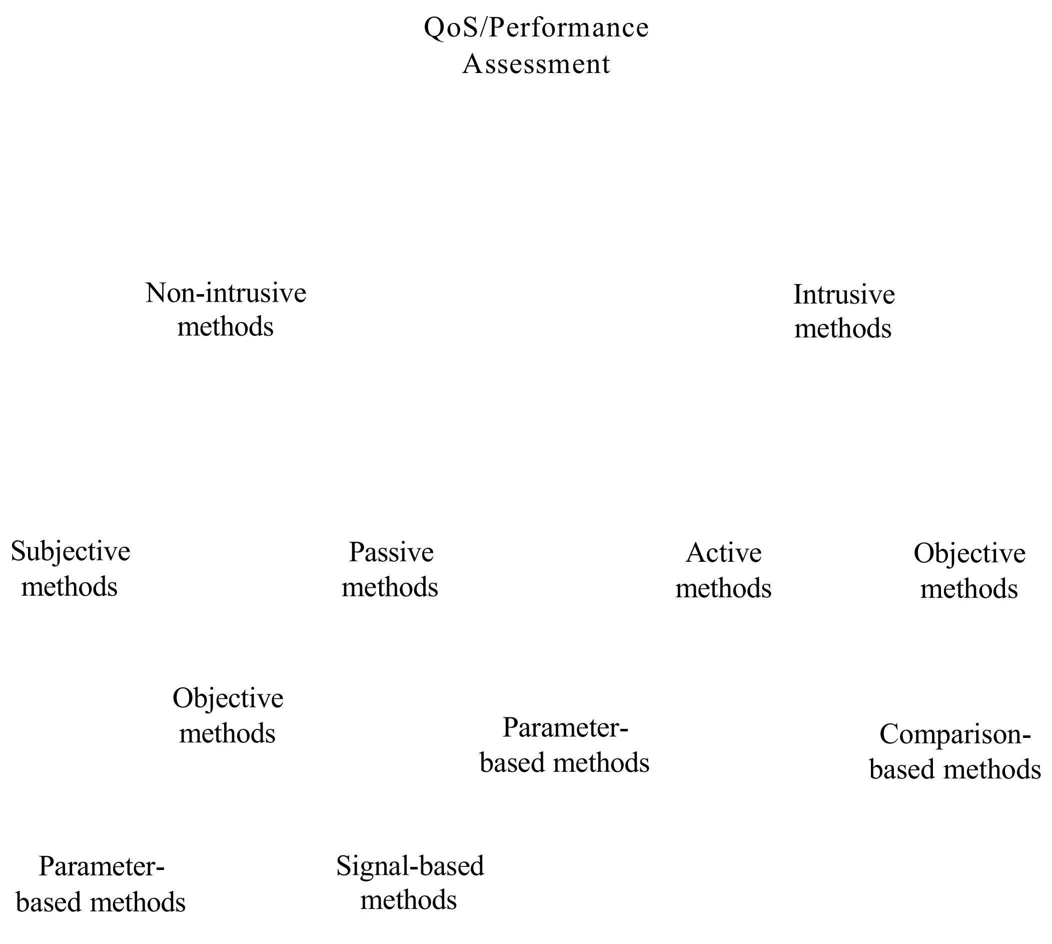


Figure 3.1: Classification of multimedia QoS assessment methods.

3.3 QoS Assessment/Measurement Methods

Before starting and performing any measurement, careful attention and consideration must be taken of the issues that may influence the measurement results. These include: (i) measurement technique; (ii) selection of useful QoS metrics; (iii) monitor placement within the network topology and (iv) measurement period. In this section, the measurement techniques are discussed. As mentioned before in Section 3.2, multimedia quality assessment can be carried out using either subjective or objective methods and passive or active methods. In this section, some details about these approaches are provided.

3.3.1 Subjective and Objective Methods

3.3.1.1 Subjective Methods

Subjective measures, as implied by their name, require human subjects listening/watching to a live or recorded application (audio or video) (ITU, 1996a) (ITU, 1996b) (ITU, 2000) (ITU, 2002). These methods refer to opinion rating and/or measurement of task performance. This rating measures the overall perceived multimedia quality. The applications that are assessed in these tests are generally specific material, recorded or spoken under defined conditions (Watson and Sasse, 1996). Because they use human subjects, subjective measures are often very accurate and useful for evaluating a telephony system (Hall, 2001). In addition, there exist standard methods for conducting subjective quality evaluations for video (ITU, 2000). The most commonly used measure for quality evaluation is the Mean Opinion Score (MOS) (Afifi, et al., 2001) (Hall, 2001) (ITU, 1996a). MOS was the first described method in ITU P.800 recommendations for subjective voice quality evaluation (ITU, 1996a). The ITU recommended the MOS test method for subjective tests is the Absolute Category Rating (ACR) (Kajackas and Anskaitis, 2005). A MOS ACR value is normally obtained as an average opinion of quality based on asking people to grade the quality of the application signals on a five-point scale (5 Excellent, 4 Good, 3 Fair, 2 Poor, 1 Bad) under controlled conditions as set out in the ITU standard. In addition, Degradation Category Rating (DCR) is also used to conduct quality subjective tests which provides Degradation Mean Opinion Score (DMOS) based on an annoying scale and a quality reference (Sun, 2004).

Clearly, a metric such as MOS that uses human subjects can be a good measure of perceived quality and has served as the basis for analysing many aspects of multimedia signal processing (Afifi, et al., 2001). However, subjective metrics have disadvantages, too. In particular, they can be time-consuming and are expensive to repeat frequently due to their human dependent and each test takes a long time to be completed. Therefore, some researchers or organizations may not have the resources to conduct these types of tests. Certainly, such metrics cannot be used in any sort of real-time or online application quality assessment. In addition, it requires very stringent environments and the process of assessment can not be automated (Hall, 2001).

3.3.1.2 Objective Methods

The above mentioned limitations and shortcomings of the subjective tests have led to the development of objective metrics (ITU, 1998) (ITU, 2001c) (Rix, et al., 2000). The objective methods measure the quality based on mathematical analysis (Afifi, et al., 2001) (Wu, et al., 1996). Such measures predict the application perceived quality based, typically, on a computation of distortion between the original (clean) signal and a received (noisy) signal (Hall, 2001). In some algorithms, something other than the difference between the received and original signals is used, such as a quantitative measure of the distortion. Some existing methods are based on Mean Square Error (MSE) or Peak Signal to Noise Ratio (PSNR) which measures the quality by a simple difference between frames. These measures usually depend on functions of measured parameters which are related to the encoder used or to the network.

There are several objective quality algorithms, like Perceptual Speech Quality Measure (PSQM) and Perceptual Evaluation of Speech Quality (PESQ) that provide an objective MOS-equivalent score for a voice call (ITU, 1998) (Pennock, 2002). PSQM was originally designed to evaluate codec quality. PSQM+ is an enhancement of PSQM to cover short duration temporal clipping as often seen in wireless communications. PESQ is an intended replacement of PSQM (ITU, 2001c). It was developed by combining the two advanced speech quality measures PSQM+ and PAMS (Perceptual Analysis Measurement System). PESQ compares an original speech sample $x(t)$ with its transmitted and hence degraded version $y(t)$ (Kajackas and Anskaitis, 2005). After some pre-processing, both the original and degraded speech signals are transformed into a psychoacoustic representation which models the properties of the human auditory system. The output of PESQ is a prediction of the perceived quality that would be given to $y(t)$ by subjects in a subjective listening test and directly produces an objective MOS ACR in the range 1 to 5. PESQ provides a significantly higher correlation with subjective opinion than the PSQM (Rix, et al., 2000).

Although intrusive objective methods have overcome some of the limitations of the subjective approaches, they still present several disadvantages. Afifi, et al., (2001) summarised these drawbacks as: i) these methods do not correlate well with human perception; ii) they require high calculation power; iii) they are time consuming because they usually operate at the pixel level; and iv) it is very hard to adapt them to real-time quality assessment, as they work on both the original video sequence as well as the transmitted/distorted one.

Unlike the intrusive objective methods, in which a reference signal must be injected into the tested network, non-intrusive assessment methods do not need the injection of a reference signal. Non-intrusive approaches are based on predicting the quality directly from varying network impairment QoS parameters or non-network parameters like codec, echo...etc. The goal is to establish a relationship between the perceived QoS and the network or the non-network parameters. A typical method for achieving that is the E-model.

3.3.1.2.1 E-Model

In order to overcome the above described limitations of the intrusive objective methods, ITU recommendation G.107 introduced the E-model (ITU, 2003) (ITU, 1999). The E-model provides a powerful, non-intrusive and repeatable objective technique to assess the multimedia quality. In contrast to the two approaches described above (subjective and objective), the E-model does not compare the original and received signals directly as in objective methods nor depends on humans to assess the quality as in subjective methods. Instead, the E-model allows the obtaining of an approximation of the perceived quality as a function of several ambient, coding and network parameters (Mohamed, et al., 2004). The output of an E-model calculation is a single scalar, called an "R factor," derived from the sum of delays and equipment impairment factors (Assessing, 2005). Once an R factor is obtained, it can be mapped to an estimated MOS using the equations stated in the ITU G.107 (ITU, 2003). Impairment factors include codec used, echo, average packet delay, packet delay variation, and fraction of packets dropped. As an example, in a system with distortion due to the codec, average one-way delay, packet delay variation (jitter) and packet loss, the quality rating R is computed as follows (Assessing, 2005) :

$$* \, 0 \qquad codec \qquad I \; delay \qquad \hat{} jitter \qquad \hat{} \; packetloss \,) \tag{3-1}$$

where *Ro* is the highest possible rating for this system with no distortion and is equal to 100. Each time a test is run; measurements are collected for the one-way delay time, the number of packets lost, and the amount of jitter of the packets. The MOS can range from 5 down to 1 (Mohamed, et al., 2004). In addition to the user satisfaction, an estimate of the MOS can be directly calculated from the E-model R factor, as depicted in Figure 3.2 (Assessing, 2005).

R	*User Satisfaction*	*MOS*
100		I *44*
	Very Satisfied	
90		4.3
	Satisfied	
80		4.0
	Some users Dissatisfied	
70		3.6
	Many Users Dissatisfied	
60		3.1
	Nearly All Users Dissatisfied	
50		2.6
	Not Recommended	
0		1.0

Figure 3.2: Quality classes according to the E-model.

E-model is an attractive and useful non-intrusive quality measure, but it has a number of restrictions. For instance, it is based on a complex set of fixed formulas which are applicable to a limited number of codecs and network conditions. In addition, some subjective tests are required to obtain the model parameters which hinder its application in new and emerging multimedia applications. In addition, E-model is a static model which can not adapt to the dynamic environment of the IP networks (Sun, 2004). This makes the need for devising new models to evaluate the application QoS imperative.

Our study will focus on examining and developing methods to infer and assess the QoS of time-sensitive multimedia applications; audio and videoconferencing directly from the network QoS parameters of these applications. This includes devising and application of new methods and approaches. These methods are based on passive and active measurements techniques. These two approaches will be discussed in the following subsections.

3.3.2 Passive and Active Measurement Methods

Traffic measurements are gradually receiving more and more attention from both network and service operators (Brekne, et al., 2002). The objective of network measurement and monitoring is to provide information about the network/traffic conditions enabling the network managers and operators to characterise the state of the network and to evaluate the traffic requirements, demands and its consumption of the network resources. Monitoring and measurement schemes usually fall into two categories: passive and active methods. The former are those based on (transparently) collecting and analysing the traffic observed at a certain point of the network and the latter, which is based on injecting synthetic traffic flow into a network.

3.3.2.1 Passive Methods

Passive measurement allow the tracking of the behaviour of traffic flow because it allows the properties of carried traffic to be observed (Brekne, et al., 2002). It is a traditional technique used to obtain measurements of QoS parameters related to a certain network element (Paxson, 1999), (Paxson, 1997), (Smotlacha, 2001) and (Johnsson, 2005). This method is based on monitoring the performance of packet streams through a network by tracking the traffic passing by a measurement point without creating or perturbing it. So the packet's statistics can be gathered without adding any new traffic. This can be done by collecting traffic flow data, from routers, switches or end-point hosts. Another method, for traffic collecting, is implemented by adding a stand-alone server at the location of interest (e.g., core or edge) of the network, which acts as a traffic meter or a monitoring device by storing information about the crossing traffic.

Therefore, this type of measurement methods acts as an observer inside a network and usually will not interfere with other traffics. The levels of details and accuracy of the information gathered at the measurement points depend upon how much metrics are being processed and the volume of traffic passing through the monitoring device. There are several projects which are based on passive methods like, NetTraMet (Smotlacha, 2001), NetFlow (Brownlee, et al., 1999), AT&T (Feldmann, 2000), (Fraleigh, 2001), (Johnsson, 2005), Simple Network Management Protocol (SNMP) (Case, 1990), the Waikato Applied Network Dynamics (WAND) (Cleary, 2005) and RMON (NetFlow, 2005).

Figure 3.3 shows the basic principle of a passive measurement (Passive, 2005). It can be seen that it consists of two entities and a monitor. The monitor 'snoops' on all the traffic flowing between these two entities. Furthermore, the monitor also may be located in routers or end hosts to observe the characteristics of the traffic passing through them. Passive measurements can be done on two levels (Michaut and Lepage, 2005):

(i) At a microscopic level, measurements are performed on each packet travelling across the measurement point.

(ii) At a macroscopic level, measurements are performed on flows. In this case, aggregation rules are necessary to match packets into flows. Examples of collected data are the number of flows per unit of time, flow bit rate, etc.

Network Link

Entity 1

Entity 2

Monitor

Figure 3.3: Basic passive measurement setup.

Passive measurements may fall into two major classes. The first class deals with the captured data in real-time (on-line analysis), for example, by examining the packet and counting the number of bytes passing the monitor per unit of time. These statistics are very small when compared to the amount of data that could pass the monitor (Passive, 2005). These outputs can be used, for example, to see if available bandwidth is being fully utilised or if there are peak times where more bandwidth could be required. The second type of passive measurement is to create files, which contain copies of portions, or all of the traffic monitored on the link over a certain period. These files may then be processed and analysed later (non-real time analysis or off-line). This can allow advanced computation to be carried out that would be impossible in real-time, and preserves data for further analysis.

The data extracted from these packets are used to measure the QoS for each user and to determine the network performance. This can be done using one of the following monitoring categories (Ishibanishi, et al., 2004) (Aid, et al., 2002):

(i) Two-point monitoring: this method needs two monitoring devices to be deployed at the ingress and the egress points of the network. The packet data will be taken sequentially and the network performance parameters like delay and losses can be determined directly by comparing the data of the corresponding packets taken at each monitoring point (device). In passive measurements, all devices must be time synchronised.

(ii) One-point monitoring: this method is based on the acknowledgement mechanism of the received packets. By monitoring the acknowledgement-packet pairs at a point in the network, the RTT can be measured and the losses can also be detected.

The types of information that can be obtained based on passive traffic monitoring are (Landfeld, et al., 2000):

(i) Bit or packet rates,

(ii) Packet timing (timestamps of the inter-arrival and inter-departure timing) which can be used to calculate the delay and the jitter, and

(iii) Queue levels in buffers, which may be used as packet loss and delay indicators.

Thus, passive methods provide information on the amount of traffic crossing a measurement point of the network in order to estimate the bit rates, number of bytes or packets that have been sent or received, packet dropped or the queue levels. This can be achieved by maintaining counters in the network nodes. Furthermore, to achieve accurate timestamps to measure the delay, the measurement points must be synchronised by Global Positioning System (GPS), Global Time Base (GTB) or Network Time Protocol (NTP) (Jiang, et al., 2000). Besides, they have the advantage of not adding an extra load to the network, i.e., they are a non-intrusive method, and enable gathering of large amount of detailed information (CoralReef, 2005) (Lindh, 2001). Nevertheless, passive monitoring schemes may occasionally have the concatenation of several monitoring points, which means that they may not able to provide an end-to-end evaluation of the network performance. But if the user terminals (i.e. end-hosts) are employed as monitoring points, the end-to-end performance measurements can be achieved. Another disadvantage is that they require the transfer of the captured data for comparison with the other data and the identification of each packet by its header or content, which is hard when the traffic volume is large. Therefore, passive measurements have the disadvantage of requiring substantial resources for comparison and computation.

3.3.2.2 Active Methods

Another way of measuring the network performance is the active measurement. This method is becoming increasingly important due to its great flexibility, ability to achieve end-to-end measurements, and freedom from the need of accessing the core of network. In this method, QoS and the performance of a network are measured by injecting of some artificial probing packet streams into the network and monitoring them from a source to a destination. Active measurements can determine the QoS experienced by the probe flow for a particular path and then measure the QoS as it is seen by applications. The purpose of these probing packets is to provide some insight into the way the user traffic is treated within the network. The QoS and performance of the probe-packet stream are monitored to infer the performance of the user's packets and the network directly. There are several tools which are based on active methods like, the Internet Control Message Protocol (ICMP) Echo Reply/Request messages (ping) which is defined in RFC 729 (Postel, 1981), traceroute (Traceroute, 2002), Surveyor (Surveyor, 2004), Active Measurement Project (AMP) (NLANR, 2006), Internet Measurement Structure (Matthews and Cottrel, 2000), and Surveyor (McGregor, et al., 2000) and Service Monitoring Management Information Base (SM MIB) (Choi and Hwang, 2005), Cisco Internet Performance Monitor (IPM)(Cisco, 2004), and (Johnsson, 2005).

The basic components of an end-to-end active probing structure are shown in Figure 3.4. In each probing experiment, the sender generates and transmits a probe stream, which traverses some route in the network and terminates at the receiver (the sink). Together with the probe sequence numbers available from the payloads, the packet arrival and departure timestamps define the raw outcome of the experiment (Pasztor and Veitch, 2001). They are recorded by the sender monitor and the receiver monitor, respectively.

By selecting particular properties at the sender (like packet size, departure time, bit rate, etc.), it is potential to compute metrics by analysing the probe flow characteristics (e.g. arrival time) at the destination so, one can determine end-to-end metrics (from the source to the destination) (Michaut and Lepage, 2005). The types of metrics that can be derived from the active measurement methods are (Landfeld, et al., 2000):

- Connectivity,
- Delay,
- Delay variation (Jitter),
- Packet losses,
- Link bandwidth (capacity),
- Bottleneck bandwidth,
- Available bandwidth.

Probe Sender

Network

Sender Monitor

Receiver Monitor

Figure 3.4: The basic components of an active monitoring method.

Connectivity between hosts, routers and end-points can be measured using the ICMP ping (Postel, 1981), traceroute (Traceroute, 2002), or skitter (Skitter, 2002) tools. Ping and traceroute are also employed for the delay and loss measurements. Every probe packet is assigned a timestamp at the sender and the receiver, based on these timestamps delay can be calculated. To achieve this, the cooperation of the sending and the destination hosts is required. Losses can be measured by injecting several probe packets and recording the number of lost packets. In addition, bandwidth can also be measured using the active probing. Several approaches have been used like (Dovrolis, et al., 2001) (Pathchar, 2002) (Pchar, 2002) (Clink, 2002) (Lai and Baker, 2000). Generally, these methods estimate the bandwidth based on the distance between the arrival times of the injected probe packets at the destination. The probe packets may be inserted by single, two or more back-to-back, or train fashions as discussed below.

Active measurement approaches can be classified into the following categories (Michaut and Lepage, 2005):

(i) Cooperative approaches, which consist of separate source and destination programs that are respectively, installed on the source and destination hosts.

(ii) Non-cooperative approaches, which consist of only one program that includes the sending and receiving tasks. As an example of these approaches is the measurement of the RTT.

In addition to the above, the active approach affords explicit control on the generation of packets for measurement scenarios. This includes control on the nature of traffic generation, the generation techniques, the timing, frequency, packet sizes and types (to emulate various applications), statistical quality, the path and function chosen to be monitored (Cottrell, 2001). Emulation of scenarios is easy and checking if QoS or Service Level Agreements (SLAs) are met is relatively straightforward based on the active schemes.

It is implicitly assumed that the QoS and performance of the user/network is the same as the values measured from the active probe packets. Sometimes, the measurements of the probing packets do not accurately represent and estimate the performance experienced by the actual traffic (Brekne, et al., 2002). This accuracy depends on the specifications of both the probe traffic and the actual user traffic. Therefore, in order to produce accurate results, the active probe traffic pattern must have the same pattern of the user traffic pattern being measured (Heegaard, 2002). The accuracy of the measurements depends on many factors: packet size of the probe packet, generation rate (i.e. number of injected probe packets), and its packet type. Excessive probe packets generation produce a significant load which can disturb the operation of the network. On the other hand, low probing rates can not reveal the performance accurately (Brekne, et al., 2002). So, underestimation or overestimation of the user performance and application QoS will occur if probe packet properties are very different than the user packet properties under estimation. Therefore, the active monitoring schemes may suffer from the following problems (Aid, et al., 2002):

- If a probe packet stream is used to simulate an actual user traffic:
  - (i) The probe packet incurs non-negligible extra traffic into the network and it affects QoS and the performance of user's traffic, and
  - (ii) The QoS and performance obtained from the probe packets will not be equal to the unbiased one i.e. the results obtained without the presence of the probe packet stream.

- If probe packets of small length have been used and sent periodically, the extra traffic may be negligible, but the QoS and performance results obtained from the probe packets are not exactly equal to the QoS and performance experienced by the user.

For accurate results, certain active measurement procedures need a strong time-related constraint to be achieved (Michaut and Lepage, 2005), these are:

- Timing accuracy in probes injecting.
- Accurate time-stamping of probes upon arrival to the destination point.
- Accurate time synchronisation of the source and the destination to allow clock comparison between the two hosts.

As indicated before, active measurements are based upon probing packets. There are many forms of how to probe the network with these packets. The most common methods of packet probing are (Hu and Steenkiste, 2003):

(i) Packet Pair Probing (PPP): the source sends multiple packet pairs to the receiver. Each packet pair consists of two packets of the same size sent back-to-back. The dispersion (separation) is the time distance between the last bit of each packet. Figure 3.5 shows the dispersion of a packet pair before and after the packet pair goes through a link. Measuring *Aout* and *Ain* is known, both of them are used to calculate the delay, jitter, link capacity, and cross traffic.

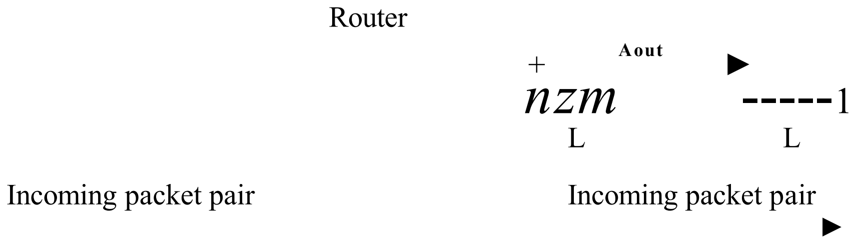


Figure 3.5: Packet pair dispersion.

(ii) Packet Train Probing (PTP): this is an extension of the PPP by using multiple back-to-back packets. The dispersion of a packet train at a link is the amount of time between the last bit of the first and last packets. After measuring the end-to-end dispersion for a packet train of length *N,* it can be used to calculate the delay, jitter, link capacity, and cross traffic.

(iii) The IP Performance Metric (IPPM) Working Group recommends Poisson and periodic sending processes of probe packets (Raisanen, et al., 2002). Periodic scheme is the process of generating probe packets based on a pre-determined function, like sending one packet every t time. It is quite attractive because of its simplicity and ease of implementation and it appears better adapted to measuring continuous multimedia streams (Michaut and Lepage, 2005).

Our study concerns with the QoS assessment and measurement over wireless ad hoc networks and because of the available resource scarcity properties of these kinds of networks and due to the overwhelming characteristics of the PPP and PTP approaches; the periodic generation method will be adopted in this research.

Active measurements have several advantages. Among these is the flexibility to create probe flow with specific features to match measurement needs. These features include the packet sizes, types, and inter-departure times. Moreover, active measurements include reduction in the quantity of resulted measurements compared with the passive measurements. However, the main disadvantage of active measurements is their invasive nature (Pasztor and Veitch, 2001). The probe packets used for the measurements will perturb the network and the user traffic QoS metrics. Another important issue is that both the source and the destination of the probing packets must be timely synchronised. This means that to obtain accurate timing information measurements and to minimise the measurement errors, the sender must forward probe packets at the specific times, while the receiver must produce accurate timestamps for the arriving packets.

Finally, passive measurements overcome the disadvantages of active measurements with regard to the overheads and delay by monitoring (probe) streams. In addition, it can provide more precise performance evaluation of user traffic than active measurements. That is because in passive monitoring, the actual user traffic packets themselves are measured rather than depending on results of probe packets.

To overcome some of the disadvantages of both active and passive approaches, several studies were carried out. These studies were based on combination of active and passive methods. One of these methods is the Change-of-measure based active/passive monitoring (CoMPACT) (Aida, et al., 2003), (Ishibanishi, et al., 2004). This is a light active measurement method transformed by using passively monitored data to correct the probe results to be closer to the actual user performance. This method was only applied to estimate the actual user delay. Another technique has been proposed which combines passive and active ways (Lindh, 2002), (Lindh, 2001) from a probe report. In this technique, a router sends active probe packets at regular intervals. The passive monitoring method is used to count the number of user packets passing through the router. This approach has been used to estimate the QoS parameters only (i.e. delay, packet loss... etc.) over wired networks.

3.3.2.3 Examples of Tools for Passive and Active Measurement of QoS Parameters

Numerous groups of the Internet Engineering Task Force (IETF) are working to enhance the standard best effort service to guarantee or, at least, to improve the QoS of data transmissions (Breslau and Shenker, 1998). The IPPM working group mainly searches for mechanisms to describe and measure the QoS of unicast connections. The working group defined several methods and metrics describing QoS parameters. Individual tools and algorithms are available to measure QoS parameters in an IP network based on these definitions using the passive and active approaches:

3.3.2.3.1 Surveyor

The Surveyor tool (Surveyor, 2004) based on an active measurement, periodically measures the performance of wide-area network. This is done by measuring the end-to-end delay, loss, and routing among a diverse set of measurement probes through the network. Delay and loss are measured using the same stream of active probing traffic. Each probe packet is of minimal size: 12 bytes and essentially with a sequence number and a timestamp. These packets are sent using UDP, so the actual packet size, excluding any MAC header, is 40 bytes.

There are three major components used in the Surveyor infrastructure: measurement machine, the database, and the analysis server. The Surveyor measurement machines collect performance data and buffer them to local disk (database). Once every few minutes the measurement machine is polled for new performance data; if there is data, it is uploaded to the central database. Finally, analysis is performed by and made available through an analysis server (Zseby and Scheiner, 2002).

3.3.2.3.2 Cisco Internet Performance Meter / Service Assurance Agent

The Cisco Internet Performance Monitor (IPM) (Cisco, 2004) is another active measurement tool used for monitoring the performance of multi-protocol networks. It is used to achieve many tasks including, monitoring latency, availability, jitter, packet loss, and errors between two network points. To fulfil all these tasks, the IPM solution consists of three parts: the IPM server, the IPM client applications, and the Service Assurance (SA) Agents. The IPM server provides central services and functions as a measurement database. It manages the exchange of data between the measurement devices and its central database (Zseby and Scheiner, 2002).

Both the client and the server do not perform measurements. They are only used to organize the SA Agent, which execute the measurements on a Cisco router. This SA Agent is the only source for all measurements. SA Agent is capable of performing probing measurements at the network (IP), the transport (TCP, UDP) and the application layer.

3.3.2.3.3 WAND

The WAND (Waikato Applied Network Dynamics) is a passive tool, which has been used to perform some latency and loss measurements (Cleary, 2005) (Graham, et.al., 1998). In particular, it was designed to capture ATM cells passively, recording a timestamp and signature of each cell. These signatures can be correlated off-line to find one-way delays accurate to 10 nanoseconds. WAND has also developed an Ethernet interface that uses the same technique.

3.3.2.3.4 IPMON

Another example of a passive measurement tool is the IPMON system (Fraleigh, et al., 2003). It is, among other things, able to collect packet traces at several points in the network. A packet trace gives a detailed picture of what happens on the monitored links. The packet traces are then used in analysis of traffic behaviour. Using IPMON it is possible to study packet size distributions and the protocol type distribution (e.g. mail, http) etc.

3.3.2.3.5 AQUILA

Adaptive Resource Control for QoS Using an IP-based Layered Architecture (AQUILA) project defines and implements QoS architecture for dynamic end-to-end service provisioning in IP networks (Engel, et al., 2003). Its architecture guarantees QoS parameters for end-user applications, like low delay, low packet loss and a specific amount of bandwidth. A compromised methodology of active as well as passive measurement approaches was used when AQUILA evaluates QoS measurements. Active measurement is performed by synthetic application-like flows and by probing flows. While application-like flows are emulating real end user applications, probing flows are thin measurement flows for monitoring the network behaviour. Passive measurement in AQUILA relies on data gathered from different network elements (Hofmann, et al., 2002), (Hofmann and Miloucheva, 2001). In AQUILA, measurements are also used to support QoS mechanisms like resource control and admission control. For more information about the active and passive measurements studies, the reader is advised to refer to the Cooperative Association for Internet Data Analysis (CAIDA) (CoralReef, 2005) and to the National Laboratory for Applied Network Research (NLANR) (NLANR, 2006).

3.3.3 Previous Work

Based on the growing importance of multimedia applications in the Internet, different measurement and evaluation approaches have been proposed to monitor and assess network QoS parameters which affect the quality of these applications. One of the distinct methods that carry out traffic monitoring is EdgeMeter (Molina-Jimenez, et al., 2004) (Pias and Wilbur, 2001). EdgeMeter is a distributed meter system designed to monitor QoS of traffic over IP networks. Its architecture is distributed in the sense that it can be deployed to collect metric in the provider's enterprise and in the service consumer's. Metrics collected by EdgeMeter can readily be used for billing; likewise, they can be useful for network planning and QoS monitoring of applications.

H.323 is an umbrella standard that defines how real-time multimedia communications, such as audio and videoconferencing, can be exchanged on packet-switched networks (Internet) (ITU, 1999). This has led to the need to identify the behaviour of these applications as well as its impact on the end user perceived quality of the H.323 applications over the Internet. Several studies and many approaches have been proposed to determine the performance quality measures of H.323 applications (Calyam, et al., 2004) (Markopoulou, et al., 2002) (Marsh and Li, 2003) (Mullin, et al., 2001). Many of these studies used pre-recorded audio and video streams and aimed at obtaining quality measures either based solely on network variations or on various audiovisual quality assessment methods like subjective and objective approaches.

Bolot (1993) measured the round trip delays of small UDP probe packets sent at regular time intervals to analyse the end-to-end packet delay and loss behaviour for VoIP. Papagiannaki, et al., (2002) provided an analysis of the measured single-hop delay from an operational backbone network and its impact on the VoIP quality. Moreover, an approach to derive an exact metric for numerical evaluation of the QoS of Internet connections was discussed in (Dressler, 2003). In this work, the quality of the connection was estimated as a vector of single weighted metrics, and the numerical representation of the overall connection quality is the product of the single values of the weighted metrics. These metrics, especially developed for verification of SLA of multimedia services, consider parameters like throughput, delay, and jitter and packet loss ratio. Besides, Miloucheva, et al., (2004) presented a technique for monitoring of network QoS parameter for VoIP application in inter-domain environment. This approach was designed for the monitoring of the connection characteristics for VoIP applications based on active QoS measurement of emulated VoIP traffic and detection of delay and packet loss patterns for network connections characterising the impact of the network delay and packet loss on the quality of VoIP based on the E-Model objective method. Cole and Rosenbluth (2001) investigated the use of the E-Model as a tool to relate the level of several metrics to an estimate of conversational voice quality. In their work, the reduction of the existing E-model in terms of quality metrics for the purpose of monitoring of conversational voice quality was also analysed.

As quality assessment is a subjective concept, the best way to evaluate it is to have real people do the assessment. The key problem with subjective methods is that they are very costly (in terms of both time and manpower) to perform, which makes them hard to repeat often. And, of course, they cannot be a part of an automatic process and to be carried out on-line. Therefore, methods for quantitative evaluation of audio and video quality over packet networks have been proposed. Mohamed, et al. (2000), Mohamed, et al. (2001) and Mohamed and Rubino (2002a) outlined several Artificial Neural Network (ANN) models which were used to predict voice or video quality from network or non-network parameters. Mohamed, et al., (2004) proposed a method which is a hybrid between subjective and objective evaluation methods. The idea is to have many distorted samples evaluated subjectively, and then use the results of this evaluation to teach a Random Neural Network (RNN) the relation between the parameters that cause the distortion and the perceived quality. In order for it to work, the author argued for the need to consider a set of parameters (selected a priori) which may have an effect on the perceived quality. Another approach has been presented by Rubino, et al. (2006) for objective quality assessment to substitute the subjective methods. In this paper, an assessment mechanism recently developed and used based on results obtained by Mohamed and Rubino (2002a) and Mohamed, et al. (2004). The idea is to train a RNN to behave like a 'typical' human evaluating the streams. This is done by identifying an appropriate set of input variables related to the source and to the network, which affect the quality, and mapping their combined values into quality scores. In addition, a stochastic model for the wireless network that allows simulating its variations in performance, and seeing how they affect the perceived quality of the streaming applications was used.

All of the above proposed ANN approaches have the same drawback. This is represented by the fact that they are relying on subjective tests to create the training sets. As a result the training sets are limited and cannot cover all the possible scenarios in dynamic and evolving networks, such as the wireless networks. Therefore, the impact of a variety of network parameters (e.g. delay variation, and loss rate) on perceived quality remains unclear based on the ANN approaches.

Generally, in wireless networks, the majority of the related work concentrates on routing and analysing or optimising mechanisms for the regular 802.11 medium access layer. Relatively few give attention to the measurements themselves and make conclusions about the quality of the multimedia applications over these networks. However, some recent studies which are devoted to wireless measurements have used wireless sniffers to obtain passive and active characterisations of the network (Portoles-Comeras, et al., 2006a). Wireless sniffers are packet capture engines that passively monitor the wireless medium capturing (non-intrusively) passing traffic (Portoles-Comeras, et al., 2006b). A sniffing system can easily be set up and put into operation without any interference to existing infrastructure, including end-hosts or network routers. In fact sniffing can be performed without any interaction with the existing network, and hence is completely independent of the operational network (Yeo, et al., 2002). Despite these advantages, wireless sniffing has the following challenges (Yeo, et al., 2004):

(i) Limited capability of each sniffer: each sniffer has the limitations, e.g. on signal receiving range, disk space, processing power, etc.
(ii) Placement: finding the best location for each sniffer is difficult.
(iii) Data collection: it is difficult to collect and synchronise a large volume of data from multiple sniffers.

In this research, wireless monitoring (sniffing) is utilised. This is based on passive monitoring approach. While passive measurements serve to characterise the traffic and other operational parameters (e.g. loads) at any particular point of the network, our measurements are attached and performed at the destinations. Most of the above mentioned sniffing challenges and limitations have been overcome using sampling techniques. Sampling methodologies, properties, and implementation will be discussed in Chapters 7 and 8.

To date, two threads of research have examined the property or performance of the IEEE 802.11 for multimedia transmissions: performance analysis, and performance and/or QoS enhancements (Zhai, et al., 2005). Many studies developed to study the performance of the IEEE 802.11 based on analytical models to assess its capability for supporting major QoS metrics, i.e., throughput, delay, delay variation, and packet loss rate (Zhai, et al., 2005) (Bianchi, 2000) (Zhai, et al., 2004). Based on the IEEE 802.11, both DCF and PCF modes provide inadequate performance (Visser and El Zarki, 1995) and are considered to be insufficient for achieving a reasonable quality in scenarios with high background load (Koepsel, et al., 2000), therefore various performance improvements have been proposed and evaluated (Lindgren, et al., 2003).

Koepsel, et al. (2000) Koepsel and Wolisz (2001) simulations were conducted to identify the performance and whether the DCF and PCF MAC mechanisms can fulfil real-time traffic requirements. In the DCF mode, stringent delay requirements were fulfilled only in low load scenarios. In a high load scenario or in a scenario with a high number of nodes, DCF fails to provide low delay and jitter. Therefore, the authors suggest switching from DCF to the PCF mode in those cases. Koepsel, et al. (2000) showed that the audio flows are transmitted over a 2 Mbps wireless channel. In the case of an audio stream with 64 kbps rate, the capacity is 12 stations in the DCF mode and 15 in the PCF mode. As a minimal quality level, the authors have chosen a maximal transmission delay of 250 ms and maximal 5% packet loss. The usage of PCF, however, decreases the overall throughput due to unsuccessful polling attempts. Therefore, many manufacturers are choosing not to implement the optional PCF mode, claiming that it inhibits interoperability with other access points and does not, in fact, always allocate bandwidth better than DCF (Anjum, et al., 2003).

Garg and Kappes (2002) experimentally studied the capacity of IEEE 802.11b to determine the maximal number of VoIP calls. The maximal number of stations depends on the transmission rate of VoIP, the geographic distribution of the wireless clients, and the distance between the wireless clients and the base station. The authors determined the quality of VoIP calls by measuring packet delay, jitter and loss rate. Using G.711 and 10 ms interval six simultaneous calls were possible. Starting the seventh call, only the wired to wireless streams failed. The authors concluded that lowering the packet frequency is the most efficient solution to increase the number of VoIP calls in a WLAN cell.

Masala, et al. (2003) evaluated a number of "QoS indices" of a real-time video transmission over an 802.11 ad hoc wireless network by means of the NS-2 network simulator. The quality perceived by the video user at the receiver is objectively evaluated, using the PSNR as a distortion measure. Moreover, the impact of background interfering traffic was studied. From this study, it was found that in the presence of interactive video services; the number of TCP sources (background traffic) that can be admitted in the network should be limited in order to meet the QoS requirements.

Gao, et al. (2005) experimentally assessed the MPEG-4 video streaming performance over 802.11e. In particular, they discussed in details how the human satisfaction of streaming video is affected by the main QoS parameters in IEEE 802.11e WLANs. In addition, they measured the level of end user satisfaction objectively using the PSNR together with the network performance.

Cranley and Davis (2005) investigated the effect the background traffic load on unicast streaming video sessions in a WLAN environment and to monitor the resource utilisation for the video streaming application under loaded conditions. The performance of the system was measured using a WLAN probe. The probe was used to monitor WLAN resource utilisation in terms of its MAC bandwidth components. In particular, they monitored the load throughput component that is associated with the transport of data packets. As the load is increased, the throughput reaches a maximum and the AP becomes saturated and so the quality of the video deteriorates.

From the above discussion, assessing the quality of multimedia services transmitted over wireless networks has not been widely addressed by the network research community and remained a rather difficult problem to be addressed, comprehensively. This is due to the fact that, most of the available measurement approaches and methods rely and concentrate on measuring individual parameters that influence the multimedia quality rather than focusing on the overall QoS. One of the major concerns of the multimedia applications quality assessment is to maximize the QoS of these applications over a given network state. Traditionally, this is done by measuring, tracking and keeping some of the network parameters (e.g., packet loss rate and delay variation) within certain limits. However, the current Internet infrastructure provides basically a best effort service, with no provisions for QoS. Therefore, in order to deliver the best achievable QoS, a continuous tracking and monitoring of the network/application performance and a better understanding of the combined effects of all of the parameters that impact the QoS of these applications is necessary. In addition, representing the QoS in a single measure introduces many aspects to the networking communities like facilitating the process of monitoring the application/network performance because monitoring single value is much easier than observing several metrics at the same time. As well, single QoS measure will ease the process of issuing the SLAs between the user and the network operator because this will summarise the agreement to be for one item rather than several items (i.e. QoS parameters).

The main contribution of this research is in the development of a light, scalable and efficient quality assessment mechanisms to study and analyse the influence of certain quality-affecting parameters on time-sensitive multimedia applications performance. The effects of some of these parameters have been the subject of previous research like (Paxson, 1997) (Lindh, 2001) (Lindh, 2002) (Ishibanishi, et al., 2004) (Choi and Hwang, 2005). However, those studies normally consider only one or two parameters at once, thus neglecting the effects of their interaction as a whole. Regarding the objective techniques, there were no formerly published objective QoS evaluation methods that consider the direct impact of the whole set of the effective QoS parameters, considered simultaneously, on the perceived QoS of the multimedia applications over wireless networks. The approach presented allows better understanding of the influence of all parameters considered at the same time. The techniques present advantages over the other available objective evaluation methods, since they do not need to access the original signal, and they are not computationally intensive. Therefore, they can be used and implemented in real-time QoS applications assessment.

3.4 Summary

The purpose of this chapter was to present the state of art of the assessment and measurement methods used to determine the QoS/performance of the multimedia applications. In addition to the subjective assessment methods, the objective methods have been discussed. From the available objective speech quality measures in the literature, only the ITU E-model does not need the access to the original signal to compute the quality (Sun, 2004). The E-model offers the ways to consider several quality effects when designing a telephony network. Most of these effects are related to the field of signal processing rather than to the field of computer networks. Moreover, the active and the passive measurement approaches features and applications have also been described. Additionally, some recent selected prior work based on the above mentioned techniques related to network performance evaluation and time-sensitive multimedia applications assessment have been considered. Furthermore, the differences between the approaches proposed in the literature and the proposed techniques are also outlined. The next chapter focuses on the description of the experimental approach that was followed to evaluate and validate the proposed QoS assessment mechanisms.

Experimental Procedure

4.1 Introduction

Methods of network and proposed protocols performance can be investigated, evaluated and analysed using simulation tools, analytical models or practical real-networks. This research was based on network simulations. Generally, and for primary investigations, simulation is more flexible than the real network implementation and has fewer complications than the analytical modelling approaches. Simulation allows changes to the network topologies, protocols and parameters to be carried out easily and in realistic time. In addition, by using simulations, more control over the network conditions could be achieved. The research method of this study was simply based on data collection from simulation runs using the NS-2 simulation tool. This data were, subsequently, quantified and analysed through fuzzy logic-based and Distance measure-based assessment systems.

This chapter provides an explanation of the general experimental approach followed in this thesis. The structure of this chapter is as follows: Section 4.2 covers the simulation model used which includes description of the simulation tool, simulation environment and protocols. Then, Section 4.3 presents the audio and video traffic characteristics, QoS metrics and requirements. After that, Section 4.4 overviews the simulation methodology. Finally, Section 4.5 provides a summary of this chapter.

4.2 Simulation Model

4.2.1 Network Simulation Tool

The proposed assessment methods performance is evaluated via computer simulations. Fortunately, computer simulation is a particularly powerful and flexible tool and becoming at the stage where it plays an important part in performance analysis. Discrete-event simulation is the main tool used to study the characteristics and predict the behaviour of communication networks and, more in general to study complex stochastic dynamic systems modelling real-world situations of practical interest (Di Caro, 2003). In a discrete-event simulation, state variables only change according to the sequence of events which are happening at discrete points in time. All the simulations of this work were carried out using the NS-2 simulator (NS-2, 2005) (Fall and Varadhan, 2005). NS-2 is an open-source simulation tool developed primarily by the University of California at Berkeley. It is an object oriented discrete-event simulator written in C++ and OTcl, where an OTcl interpreter serves as a front end.

NS-2 is widely used in the networking research community and has found large acceptance as a tool to investigate new ideas, protocols and distributed algorithms (Di Caro, 2003). It has achieved a reputation and popularity among researchers, mainly because of its flexibility. The NS-2 architecture closely follows the OSI model. Its code source is split between C++ for its core engine and OTcl language for configuration and simulation scripts (Di Caro, 2003). Therefore, it allows simulation scripts to be easily written in a script-like programming language (OTcl) and more complex functionality depends on C++ code that either comes with NS-2 or is supplied by the user. This flexibility makes it possible to develop the simulation environment as required, although the main common elements are already built-in, such as wired nodes, wireless and mobile nodes, protocols, queues, links, agents, and applications. In addition, the researchers at CMU have developed support for simulating multi-hop wireless networks complete with physical, data link and MAC layer modules. Simulations in NS-2 can be logged to files called trace files, which include detailed information about transmitted and received packets and allow for post processing using many analysis tools. For details about NS-2, refer to (NS-2, 2005) and (Fall and Varadhan, 2005).

4.2.2 Simulation Environment and Protocols

NS-2 is well-suited to packet switched networks and the most used simulator for studies on wireless networks (ad hoc, local and satellite) which allows it to be as a sort of reference simulator (Di Caro, 2003). As a part of the traffic engineering approach of the wireless QoS framework, in this work the focus will be on QoS analysis in IEEE 802.11. The rest of the section presents the details of the simulation environment, the algorithms and protocols used, and the metrics used in the performance evaluation.

4.2.2.1 Physical Layer

In NS-2, the physical layer consists of a combination of the free space propagation model and the two-ray ground reflection model (Broch, et al., 1998). Free space model for short distances andground reflection model for long distances above 100 meters are usually used. In our experiments, the radio model employed was the most commonly used model in the literature which is similar to the commercial radio interface, Lucent's WaveLAN (Fall and Varadhan, 2005). The nominal bit rate and the nominal radio range of WaveLAN is 2 Mbps and 250 meters, respectively. In addition, all nodes broadcast their transmissions omni-directionally. Some physical layer specifications are shown in Table 4.1 (Fall and Varadhan, 2005).

Table 4.1: The specifications of IEEE 802.11 standard used in NS-2.

Mac/802_11 set CWMin	31
Mac/802_11 set CWMax	1023
Mac/802_11 set SlotTime	20us
Mac/802_11 set SIFS	10us
Mac/802_11 set PreambleLength	144 bit
Mac/802_11 set PLCPHeaderLength	48 bits
Mac/802_11 set PLCPDataRate	1Mbps
Mac/802_11 set RTSThreshold	0
Mac/802_11 set ShortRetryLimit	7
Mac/802_11 set LongRetryLimit	4
Antenna/OmniAntenna set X	0
Antenna/OmniAntenna set Y	0
Antenna/OmniAntenna set Z	1.5
Antenna/OmniAntenna set Gt	1.0
Antenna/OmniAntenna set Gr	1.0
Phy/WirelessPhy set CPThresh	10.0
Phy/WirelessPhy set CSThresh	1.559e-11
Phy/WirelessPhy set RXThresh	3.652e-10
Phy/WirelessPhy set bandwidth	2e6
Phy/WirelessPhy set Pt	0.28183815
Phy/WirelessPhy set frequency	914e+6

4.2.2.2 MAC Layer

IEEE 802.11 DCF is the most popular MAC protocol used in both wireless LANs and ad hoc networks (Xu, et al., 2002). In this work, the DCF mode is used. A reason for using DCF could be that DCF is a technology that has been well tested and proven to be robust in the field. For example, when there are two overlapping WLANs where both use the same frequency channel, DCF will continue to work while PCF will not, since collisions between stations of the two WLANs may occur during their supposedly contention-free periods (Wang, et al., 2005). There are two schemes for the DCF protocol, namely, two way handshaking scheme (basic mechanism) and four-way handshaking scheme (RTS/CTS). The DCF mode basic mechanism of IEEE 802.11 MAC layer protocol (Institute, 1999) is used. The values of the parameters of 802.11 DCF which have been used in the simulations are listed in Table 4.1. These specifications initialise the shared media interface with parameters to make it works like the 914MHz Lucent WaveLAN DSSS radio interface card.

4.2.2.3 Routing Protocol

Many different protocols have been proposed to solve the multihop routing problem in wireless ad hoc networks; each protocol is based on different assumptions and intuitions. These protocols include, Destination-Sequenced Distance Vector (DSDV), Temporally-Ordered Routing Algorithm (TORA), Dynamic Source Routing (DSR) and Ad Hoc On-Demand Distance Vector (AODV) (Ayyash, 2005). AODV is essentially a combination of both DSR and DSDV (Perkins, et al., 2003). It borrows the basic on-demand mechanism of route discovery and route maintenance from DSR, plus the use of hop-by-hop routing, sequence numbers, and periodic beacons from DSDV (Broch, et al., 1998). Due to that, in this work AODV routing protocol was adopted.

4.2.2.4 First-In First-Out (FIFO) Queuing

FIFO is used as the Inter-Frame Queuing (IFQ) along the whole experiments. This type is the simplest management queuing method and the most commonly used for queuing control. Very large numbers of network devices employ this type of queuing due to its simple and cheap implementation. In this case, there is no differential dealing given to any packet. This means that, the order in which the packet arrives is maintained and there is no preference given to any packet with a strict first come, first served basis. In our experiments, because we are not going to explore the effect of queuing type or queue size on the performance or the QoS of the multimedia applications, FIFO type was used with the default queue size value used in the NS-2 which is 50 packets.

4.2.2.5 Topologies and Scenarios Characteristics

The proposed approaches of QoS assessment were evaluated through simulating single and multihop wireless ad hoc networks. We assumed that the transmission range for a node is 250m. In a single-hop ad hoc network, all the nodes are in the same BSS and they can hear each other, which means that it is a fully connected network. Throughout this thesis, every experiment has its own simulation characteristics in terms of simulation topologies, scenarios and traffic features but they are common in the network specifications, settings and protocols (i.e., IFQ, MAC and routing protocols,).

4.3 Audio and Video Applications

4.3.1 Applications Traffic Characteristics

Time-sensitive applications like audio or videoconferencing require specified bandwidth, low delay and jitter but can tolerate some losses. Voice connections generate a stream of small packets of similar sizes at relatively low bit rates. Typical voice stream generation rates range from 5 Kbps to 64 Kbps, which mean that these rates remain in the tens of Kbps order. For example, the G.711 voice encoding scheme which was used through our experiments and simulations, generates 160 byte at 20ms intervals, resulting in 64 Kbps stream (Tobagi, et al., 2001). Whereas, videoconferencing connections generate streams of moderate packets of similar sizes (e.g. 512 byte) at bit rates within the wideband range, 64-2,048 Kbps rate. A common generation rate is 384 Kbps. This rate was employed to implement the videoconferencing traffic generation model.

4.3.2 Applications QoS Metrics and Requirements

Audio and videoconferencing qualities are directly affected by three QoS parameters: packet loss, delay, and jitter. Packet loss causes voice clipping and skipping. Delay can cause quality degradation if it is excessive. Jitter can cause a display monitor to flicker and will introduce clicks or other undesired effects in audio signals. The obtained performance bounds are mapped to end-users perceptions of the overall audiovisual quality and are then categorised into grades such as good, acceptable and poor. The goal commonly used in designing networks to support audio applications is the target specified by ITU. This states that 150 ms of one-way, end-to-end delay ensures user satisfaction for telephony applications. The ITU states that a 150 ms one-way delay budget is acceptable for good quality and not more than 400 ms for acceptable quality (Tobagi, et al., 2001). In addition to this, average one-way jitter should be targeted at less than 1 ms and loss should be not more than 3 percent for good voice quality. On the other hand, videoconferencing average one-way jitter and loss should be not more than 30ms and 1%, respectively in order to provide a good quality. The performance targets for conversational audio and videoconferencing applications are summarised in Table 4.2 (Tee, et al., 2005).

Table 4.2: QoS parameters range for audio and videoconferencing traffics.

Range	Low Range(good quality)		Medium Range (acceptable quality)		High Range(poor quality)	
Param eters^\^	Audio	Video	Audio	Video	Audio	Video
One-way delay [ms]	< 150	< 150	150 <& <400	150 <& <400	>400	>400
Jitter [ms]	< 1	<30	-	-	-	-
Packet loss ratio [%]	<3	< 1	-	-	-	-

4.4 Simulation Approach and Output Analysis

Generally, simulation under NS-2 consists of three steps: (i) describing the simulation in an OTcl script; (ii) running the simulation and (iii) analysing the generated trace files. Given the network topology defined by its type, operation characteristics, and the number of applications, Figure 4.1 demonstrates the basic stages and components used in the simulation experiments. Each run of the simulation accepts a scenario file as input. This file describes the sequence of packets generated by every node, together with the exact time at which each packet generation is to occur (i.e. generation rate), packet size, definition of the source and destination nodes, transport protocol, queuing type, queue size, and application type.

Scenarios File

NS Machine

Trace File

Parameters Analysis

Matlab

Summary Files

Figure 4.1: Overview of the simulation approach.

After simulating the scenario file using the NS-2, a detailed trace file is created by each run. This trace file contains detailed information about every individual packet as it arrives, departs or dropped at a link or queue of the whole traffic applications included in that scenario. This information includes: node ID, node coordinates, source address, destination address, packet type, packet size, flow ID, unique ID, sequence number, timestamps at every node that every packet passed through, and some other information which are not important for our analysis. This information will be analysed using a variety of scripts to calculate the main QoS parameters (delay, jitter, and packet loss) which impact the overall QoS of the audio and videoconferencing applications. To achieve that, the packets were differentiated by their flow ID, their sequence number within this flow, and some other relevant information such as sender and receiver node IDs. To calculate the end-to-end one-way packet delay, the difference between the values of the sending and receiving timestamps of every packet was calculated. Then the difference between two consecutive packets delays was computed to calculate the delay variation (jitter) of the flow. A segmentation procedure was used to divide the traffic packets of each flow into blocks. Each block contained a certain number of packets. Then, the average delay and average jitter was calculated for the packets contained in each block. Packet loss ratio was calculated by tracking the packet sequence number and the number of lost packets in each block.

After getting the QoS parameters for each block, these values were saved in summary files in order to be further analysed using Matlab (Matlab, 2006) to assess the overall QoS of each multimedia application. This assessment was based on developing evaluation systems to map these parameters and combining them to produce individual values which were representative of the QoS/performance of the multimedia application as will be discussed later in the following chapters. Finally, after feeding the assessment systems by the QoS parameters for each block, a vector of output values was produced.

The output QoS vector values were between zero and 100%. These values were categorised symmetrically into three regions to represent the QoS level. These regions were poor, average and good QoS regions. The categorisation process was based on two thresholds, which are 33% and 67%. These thresholds will be used as follows:

- If the QoS value was less than or equal to 33% $\Rightarrow$ the QoS was in the poor region,
- If the QoS value was greater than 33% and less than or equal 67% $\Rightarrow$ the QoS was in the average region, and
- If the QoS value was greater than 67% $\Rightarrow$ the QoS was in the good region.

Moreover, from the output QoS vector of each of the assessment systems, the overall QoS for each application was calculated (i.e., one value to represent the overall QoS). This was achieved using two procedures. One of these was to calculate the average and the standard deviation for each flow, which represented the mean and the variation in the output measured QoS. However, because there may be a high variation in the values of the QoS of some flows, a normalisation technique was applied to ensure that all the values are within the range between one and zero. Then, the average of the normalised values was calculated and multiplied by 100%. This was done using equation 4.1 as follows:

$$Normalised\,QoS_i = \frac{QoS_i - QoS_{\min}}{QoS_{\max} - QoS_{\min}}$$

$$Overall\,QoS = \left(\frac{\sum_{i=1}^{n} Normalised\,QoS_i}{n} \right) * 100\% \qquad (4.1)$$

where QoS_i is each current entry in the QoS output vector, $QoS_{\min}$ and $QoS_{\max}$ are the minimum and the maximum QoS values in the QoS vector respectively, and n is the number of values in the output QoS vector.

In order to produce a more specific picture about each application QoS, the estimation of the distribution of the QoS values was used. A cumulative distribution plot was used to determine the percentage of the measured QoS, which is less than a threshold (a). In other words, it is the probability for the QoS being less than (a). Suppose a network under consideration is shared by K applications and let $X_k(n)$ denotes the measurement objective (in this work, the QoS) of the *nth* block of application k. X has the distribution function of P. Then, the distribution of X is written as:

$$\begin{aligned} \Pr(X < \mathrm{a}) &= \int 1_{\{X<\mathrm{a}\}} dP(x) \\ &= E_P\left[1_{\{X<\mathrm{a}\}}\right] \end{aligned} \qquad (4.2)$$

where $1_{\{.\}}$ denotes the indicator function,

$$1_{\{x<\mathrm{a}\}} = \begin{cases} 1 & if\ x < \mathrm{a} \\ 0 & otherwise \end{cases} \qquad (4.3)$$

where (a) is an arbitrary real number and $E[.]$ is the expected value.

If there are n QoS measurements, $X(i)$ denotes the i-th value of X. Then the estimator $Z_X(n, \mathrm{a})$ of the distribution of X (Ishibashi, et al., 2004) , is given by:

$$Z_X(n,\mathrm{a}) = \frac{1}{n}\sum_{i=1}^{n} 1_{\{X(i)<\mathrm{a}\}} \tag{4.4}$$

4.5 Summary

In this chapter, the experimental approach that was used to evaluate and validate the proposed QoS assessment systems is discussed. This chapter defined the simulation tool used to create the scenarios employed to simulate the wireless ad hoc network topologies. In addition, the experimental environment and protocols have also been described. As well as, the characteristics, QoS metrics and requirements of the audio and videoconferencing multimedia applications transmitted over the simulated networks are explained. The following chapters will depend on the outlined experimental approach to test, validate and evaluate the proposed QoS assessment and estimation systems.

Quality of Service Assessment of Multimedia Traffic Using Fuzzy Logic and Distance Measure Approaches

5.1 Introduction

The aim of this chapter is to describe the approaches used to evaluate the performance of wireless ad hoc networks by considering the QoS requirements of multimedia applications in a simulated set up. Audio and videoconferencing applications were considered for this purpose because of the time-sensitive nature of their QoS requirements. In this work, the objective of QoS monitoring and measurement was to evaluate the performance of the wireless networks to establish whether they satisfy the requirements of different applications that were sharing the same infrastructure. This involved devising QoS assessment techniques that combine and summarise these parameters in a single value. This value represents the QoS level provided to the applications based upon the network conditions compared to the QoS level expected for those applications. Two assessment systems were devised; one based on fuzzy logic approach and other used distance measures.

The organisation of this chapter is as follows: Section 5.2 outlines the related studies. Section 5.3 describes the reasons for using the: Fuzzy and Distance approaches. Section 5.4 presents the experimental procedures, which includes the description of both the approaches and the simulation set up. In Section 5.5, the experimental results are presented. Section 5.6 summarises the advantages and disadvantages of using each approach. Lastly, in Section 5.7, a summary of this chapter is provided.

5.2 Related Work

With the rapid increase in the number of individuals in industry and academia using audio and videoconferencing, the need for assessing and monitoring the transmission quality of these applications has risen significantly. This has led to the need to understand the behaviour of audio and video traffic as it affects the end-user's perception quality. There are several methods for assessing and evaluating the quality of such applications. These studies are categorised into two approaches: subjective and objective methods. These methods were discussed in Chapter 3.

A number of studies have used fuzzy logic for network analysis problems, for example (Saraireh, et al., 2004), (We and Chen, 1999), (Oliveira and Braum, 2004), (Pitsillides and Sekercioglu, 1999), and (Fernandez, et al., 2003). To our best knowledge, the only one work on evaluating the QoS using fuzzy logic prior to our work has been by (Saraireh, et al., 2004). In the study, a fuzzy logic approach was used to evaluate the QoS for image transmission over a network and the frame rate was considered as a reference for assessing the received quality of the image.

Distance measure approach is usually used in multimedia processing as a similarity measure tool between two patterns that could be related to speech, image, graph, or signature (Li, et al., 2002), (Eidenberger, 2003), (Wu and Pols, 1996) and (Daoudi, 2006).

5.3 Why Fuzzy Logic and Distance Approaches?

Fuzzy logic is a powerful tool that uses human reasoning as an important part of system design process. A major advantage of this feature is that it allows a natural description, in linguistic terms, of problems that should be solved rather than in terms of relationships rather than precise numerical values (Nedeljkovic, 2004). Another advantage of the fuzzy system is that for some complex problems, it tends to be less computational intensive than other intelligent methodologies such as neural networks (Oliveira and Braum, 2004).

An alternative QoS assessment system was proposed relying on the principle of quantified distance evaluation between two vectors. This approach is based on the concept of Euclidean and Minkowski distance measures (Teknomo, 2006). The distance system was proposed, as a non-intelligent system to be used as a baseline to compare with the effectiveness of the fuzzy assessment system.

In this research, the use of fuzzy logic and distance approaches is justified by the absence of simple mathematical models or formulas to estimate the overall QoS. In addition, QoS assessment is a domain, which may meet the general conditions where the application of these approaches may be considered appropriate. That is because QoS is a field where the value and ranges of the important QoS parameters can be represented numerically, i.e., the QoS parameters requirements of multimedia applications. Moreover, QoS assessment is a domain where the relationship between the input parameters and the output QoS exist but may be complicated. Fuzzy logic, in addition to the distance approach, simplifies this complexity in the input-output relationship.

In addition, QoS evaluation is a problem that needs logic of reasoning which may be an approximate rather than an exact solution. Therefore, fuzzy logic is quite suitable for evaluating the QoS where the uncertainties and requirement of combination of more than one parameter (input) are present. Additionally, fuzzy logic processing is not intensive; hence, it can be executed in each node without interfering its router performance role (Fernandez, et al., 2003). Finally, fuzzy logic has the advantage of dealing with the complicated systems in a relatively simple way, which is the main reason why fuzzy logic theory is widely applied in this study. Similarly, the distance approach is uncomplicated and mathematically very straightforward, which includes one equation and a simple mapping process.

5.4 Assessment Approaches

The use of intelligent and non-intelligent methods for measurements and evaluation of overall QoS are described in this section. A performance measurement method for estimating the actual network QoS experienced by the network users has been proposed based on a fuzzy logic approach. The results obtained using this approach were compared with those obtained using distance measure approach. These approaches have been designed based on the information and background provided in Chapter 2.

5.4.1 Proposed QoS Assessment Fuzzy Logic Approach

5.4.1.1 Fuzzy System Input

As mentioned before, for audio and videoconferencing applications, the main QoS parameters are delay, jitter and packet losses. These parameters will be quantified and used as inputs to the fuzzy inference system. The fuzzy input variables were represented by three fuzzy sets to create the input membership functions for the audio depending on the requirements (Table 4.2, Chapter 4) of each input variable as shown in Figure 5.1a.

The same procedure was carried out to produce the membership functions for the videoconferencing application, which are depicted in Figure 5.2a. The fuzzy linguistic variables used were Low, Medium and High. Each input parameter was mapped to these fuzzy sets according to its value.

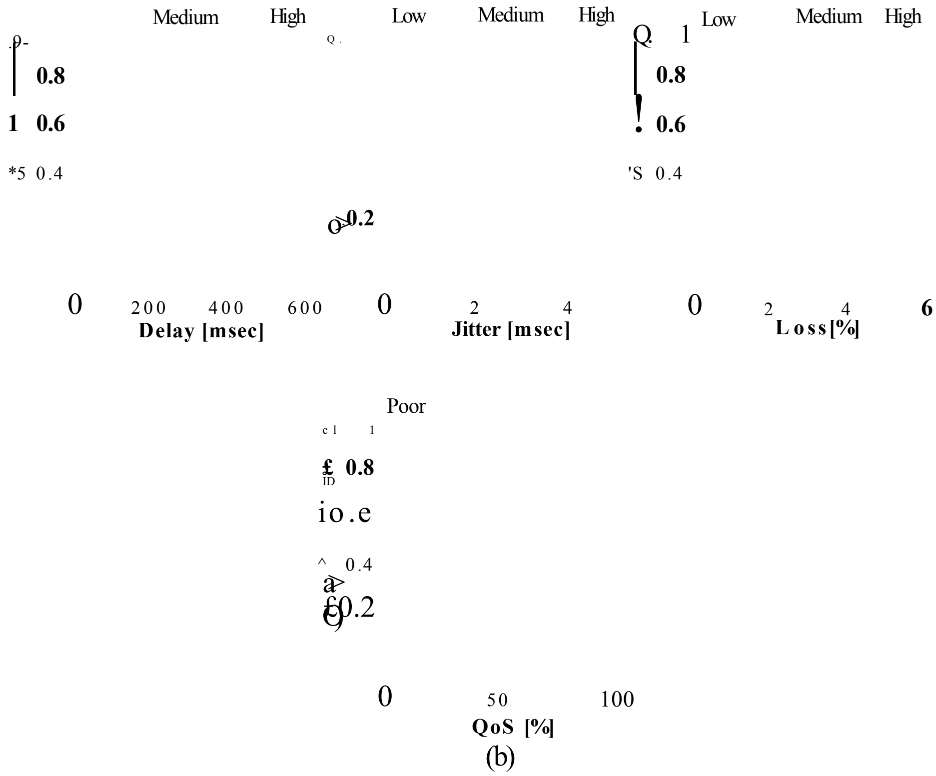


Figure 5.1: Audio fuzzy membership functions: (a) Inputs and (b) Output.

Gaussian type membership functions were used for the input variables of the fuzzy system. This type of membership functions were chosen because of its smoothness, computing simplicity and concise notation. In addition, it is the most widely used membership function in the literature (Oliveira and Braum, 2004) and it is popular method for specifying fuzzy sets.

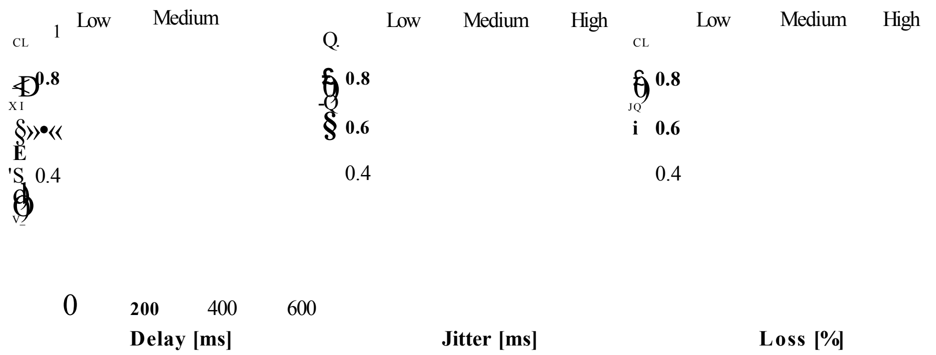
Gaussian membership function requires the mean and the standard deviation values to be defined. These parameters values are given in Tables 5.1 and 5.2 for both audio and videoconferencing, respectively. These values were selected based on the QoS requirements of each QoS parameter to provide reasonable outputs to reflect the overall QoS of each application. The selection of these values was based on the QoS thresholds defined in Table 4.2 in order to define the input regions of the QoS parameters.

Table 5.1: Mean and standard deviation values of audio input and output fuzzy membership functions.

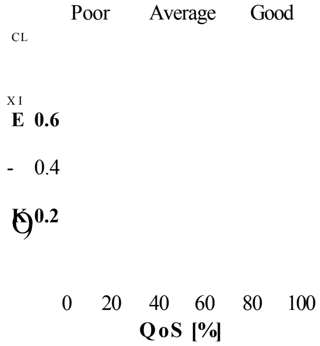
Membership	Delay [msec]		Jitter [msec]		Loss [%]		QoS [%]	
functions	Mean	St. dev.	Mean	St. dev.	Mean	St. dev.	Mean	St. dev.
Low MF	0	156	0	1	0	1.1	0	12
Medium MF	273	71	2.58	0.56	3.28	0.83	50	12
High MF	600	146	5	1.14	6	1.62	100	12

Table 5.2: Mean and standard deviation values of videoconferencing input and output fuzzy membership functions.

Membership	Delay [msec]		Jitter [msec]		Loss [%]		QoS [%]	
functions	Mean	St. dev.	Mean	St. dev.	Mean	St. dev.	Mean	St. dev.
Low MF	0	156	0	6	0	0.55	0	12
Medium MF	273	71	15.51	3.36	1.64	0.42	50	12
High MF	600	146	30	7.17	3	0.81	100	12



(a)



(b)

Figure 5.2: Videoconferencing fuzzy membership functions: (a) Inputs and (b) Output.

5.4.1.2 Fuzzy System Output

In this work, a single fuzzy output provided the assessed QoS. Hence, the output of the fuzzy system was set as the indicator of how the network dealt with the applications. In addition, the fuzzy output variable was split into three singleton fuzzy sets as depicted in Figure 5.1b and 5.2b for both audio and videoconferencing applications, respectively. The corresponding fuzzy linguistics variables were Poor (for poor QoS), Average (for average QoS) and Good (for good QoS). The Gaussian membership type function was also used for the output.

5.4.1.3 Fuzzy Rules

The number of rules depends on both the number of input variables and the number of fuzzy sets associated with each input variables. In this fuzzy system nine rules were used resulting from the combination of three inputs (delay, jitter and packet losses) each having three fuzzy sets. The specific fuzzy rules used in the evaluation process are shown in Figure 5.3.

(If delay is Low) and (Jitter is Low) and (Loss is Low) then (QoS is Good)
(If delay is Low) and (Jitter is Low) and (Loss is Medium) then (QoS is Good)
(If delay is Low) and (Jitter is Medium) and (Loss is Low) then (QoS is Good)
(If delay is Medium) and (Jitter is Low) and (Loss is Low) then (QoS is Good)
(If delay is Low) and (Jitter is Medium) and (Loss is Medium) then (QoS is Average)
(If delay is Medium) and (Jitter is Low) and (Loss is Medium) then (QoS is Average)
(If delay is Medium) and (Jitter is Medium) and (Loss is Low) then (QoS is Average)
(If delay is Medium) and (Jitter is Medium) and (Loss is Medium) then (QoS is Average)
(If delay is High) or (Jitter is High) or (Loss is High) then (QoS is Poor)

Figure 5.3: Fuzzy rules output.

From this Figure, for instance, if the three input variables have low values, this indicates that the QoS is Good. Likewise, any high value of the input variables, regardless of the other variables values results in a Poor QoS.

5.4.1.4 Fuzzy Reasoning and Defuzzification

The fuzzy reasoning was based on the minimum-maximum (min-max) inference method, where the crisp input values were mapped into the membership functions (fuzzification) and assessed according to the rules in the place. Each rule was applied to the corresponding membership functions and the minimum (min) of them was mapped into the associated output membership function. Then the output of each rule was aggregated (max) into the defuzzifier that gave the final crisp value that indicated to which output fuzzy set the outcome was to be assigned. For the defuzzification of output, the centroid was employed as illustrated in equation 2.8 (Ross, 2004). Both audio and videoconferencing used the same defuzzification method.

5.4.2 Proposed QoS Assessment Distance Approach

5.4.2.1 Distance System Description

A general measurement system is shown in Figure 5.4. It comprises four main processes: windowing, normalisation, distance measurement and mapping. The functionality and role of each component will be briefly described as follows.

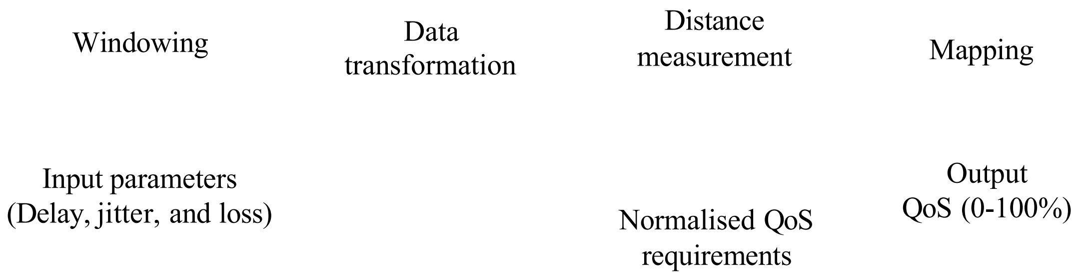


Figure 5.4: Block diagram of the distance measurement system.

As mentioned earlier in Section 4.3, for audio and videoconferencing applications, the main parameters, which affect the overall QoS, are delay, jitter and losses. After measuring these parameters, they will be processed using a windowing technique, which means gathering every *m* consecutive packets in one window (block) and calculating their average delay, jitter and packet loss. These parameters will be used as an input to the data transformation step of Figure 5.4. One weakness of the Minkowski distance function is that if an input element has relatively large values, then this value will dominate the other elements. Therefore, in this step, the distances were normalised by dividing the distance for each input attribute by specific numbers. These numbers represent the limits where the QoS will be poor. For videoconferencing, these limits were 600 msec for the delay, 30 msec for the jitter, and 3% for the packet loss. Similarly, for the audio, they were 600 msec for the delay, 5 msec for the jitter, and 6% for the loss. This was done in order to transform input data into a range which spans from 0 to 1. In this case, all the elements under the root will have the same contribution in the evaluation process, which will prevent large values from dominating the distances of the small-range elements.

5.4.2.2 Distance Assessment Mathematical Approach

The mathematical procedure followed to compute the distance between the required and the measured QoS parameters is explained in this section. After transforming (i.e., normalising) the input data (the required and the measured), the Minkowski distance calculations (distance measurement step in Figure 5.4) are carried out as illustrated in equations 5.1 and 5.2. X values represent the actual measurements (measured delay, measured jitter, and measured loss) and the Y values represent required (desired) values (delay, jitter, and packet loss). The Y values are application dependent.

$$d_{XY} = \sqrt[\lambda]{\sum_{i=1}^{n} (X_i - Y_i)^{\lambda}}$$

where $X_i = [D_m, J_m, L_m]$ *and* $Y_i = [D_r, J_r, L_r]$

Therefore; $d_{XY} = \sqrt[\lambda]{(D_m - D_r)^{\lambda} + (J_m - J_r)^{\lambda} + (L_m - L_r)^{\lambda}}$

$$d_{XY_{nor}} = \sqrt[\lambda]{\left(\frac{D_m - D_r}{600}\right)^{\lambda} + \left(\frac{J_m - J_r}{30}\right)^{\lambda} + \left(\frac{L_m - L_r}{3}\right)^{\lambda}} \tag{5.1}$$

Where d_{XY} and $d_{XY_{nor}}$ are the regular and normalised distances respectively. D_m, J_m, and L_m are the measured delays, jitter and loss, respectively. D_r, J_r, and L_r are the required delays, jitter and loss, respectively.

The distance calculations of the measured values against the required values were carried out based on the Good QoS requirements (i.e., delay $\leq$ 150 msec, jitter $\leq$ 10 msec, and loss $\leq$ 1%). This means that the normalised QoS requirement are $\{D_r = 150$ msec, $J_r = 10$ msec, and $L_r = 1\%\}$. Therefore, equation 5.1 becomes:

$$d_{XY_{nor}} = \sqrt[\lambda]{\left(\frac{D_m}{600} - 0.25\right)^{\lambda} + \left(\frac{J_m}{10} - 0.33\right)^{\lambda} + (L_m - 0.33)^{\lambda}} \tag{5.2}$$

As mentioned before if λ is selected to be equal to 2, the equations correspond to the Euclidean Distance. The Euclidean distance (i.e., λ = 2) has a problem if used in the evaluation system. From the equations above, it is obvious that the higher the distance ($d_{XY_{nor}}$), the poorer the network during that transmission period. Initially, this method will provide a value for a network based on how far the measured QoS metrics deviated from the desired values regardless of the network actually performing better than desired. As an example of this is the case in which one or all the normalised measured values of the QoS metrics were less than the required values. The resulted Euclidean value would be a value, which reflects that the network has performed poorly but actually, the network has performed better than the desired requirement. The method presumes that the network has performed poorly because of the distance between the two values. That is because due to the square (i.e., λ = 2) in the Euclidean distance formula, it does not take into account the sign between the parentheses. This also results in making the method unable to assess how "good" or how "poor" the network is performing. Therefore, λ should be an integer odd number greater than one. In this case, the method will be able to presume the performance of the network. That is because if the output of the distance measurement system block in Figure 5.4 was less than zero, this implies that the network has performed better than the requirements, while if it was equal to zero, this means it has met the requirements. On the other hand, if the output was greater than zero, the network performed worse than desired.

In order to convert the output of the distance measurement step value to a quantity that reflect the QoS or to an indictor of how the network dealt with the application, a transformation of the output calculated distance is required to a value in the range [0, 100]%. This was carried out in the mapping step of the Figure 5.4. Suppose that λ is selected to be 3, the situation at which the distance $d_{XY_{nor}}$ is minimum is when the measured QoS metrics are zeros (i.e., D_m = 0 msec, J_m = 0 msec, and L_m = 0%). Substituting this in equation 7, this produces a distance $d_{XY_{nor}}$ = -0.444. This case represents the best case of network performance (i.e., QoS = 100%). The worst network performance is when the measured metrics are equal or greater than the poor values, i.e. when $D_m \geq 600$ msec, $J_m \geq 30$ msec, and $L_m \geq 3\%$. This gives $d_{XY_{nor}}$ = 1.01 which corresponds to minimum poor QoS (i.e., QoS = 0). Therefore, we have two pairs of $d_{XY_{nor}}$ and QoS as (-0.444, 100%) and (1.01, 0%). From this information, we can determine the equation of a straight line. Given that the line passes through the two points $P_1 = (x_1, y_1)$ (i.e., (-0.444, 100%)) and $P_2 = (x_2, y_2)$ (i.e., (1.01, 0%)), then the slope of the line is:

$$m = \frac{y_2 - y_1}{x_2 - x_1} \tag{5.3}$$

Given the slope m and a point $P_1 = (x_1, y_1)$ through which the line passes, the relationship generally gets simplified algebraically to:

$$y = m(x - x_1) + y_1 \tag{5.3.1}$$

If y is replaced by QoS and x is replaced by the $d_{XY_{nor}}$, equation 5.3.1 can be rewritten as follows:

(5.4)

$$QoS = m * d_{XY\,nor} + c$$

where c is constant equal to $(y_1 - mx_1)$.

After calculating the slope ($m = -68.75$), equation 5.4 becomes:

$$QoS = 69.75 - 68.75 * d_{XY\,nor} \tag{5.5}$$

Similarly, when λ is selected to be 5 and following the same previous steps, the final equation will be:

$$QoS = 69.19 - 78.98 * d_{XY\,nor} \tag{5.6}$$

5.4.3 Topology and Traffic Scenarios Characteristics

In order to demonstrate the application of fuzzy logic and the distance assessment approache, different simulation scenarios, protocols, settings and traffic characteristics as discussed in Chapter 4 were simulated using NS-2 (Network Simulator, 2005). In this chapter, the proposed approaches were evaluated through simulating single and multihop wireless ad hoc networks. The single hop network topology used for the simulations is shown in Figure 5.5, which consists of 10 nodes. This network had five pairs of fixed source/destination hosts and all the sources (*0, 2, 4, 6, and 8*) and the destinations (*1, 3, 5, 7* and 9) were in the same basic service set with an area of (250m X 250m). Audio and videoconferencing application sources and destinations were (*0, 2, and 4*) and (*1, 3, and 5*), respectively. Cross-traffic sources and destinations were (*6 and 8*) and (*7 and 9*), respectively. This traffic was traffic in the network that corresponds to non-audio or videoconferencing usage, which intervenes between consecutive packets of a significant flow. This traffic was used to make the network busy during some selected times.

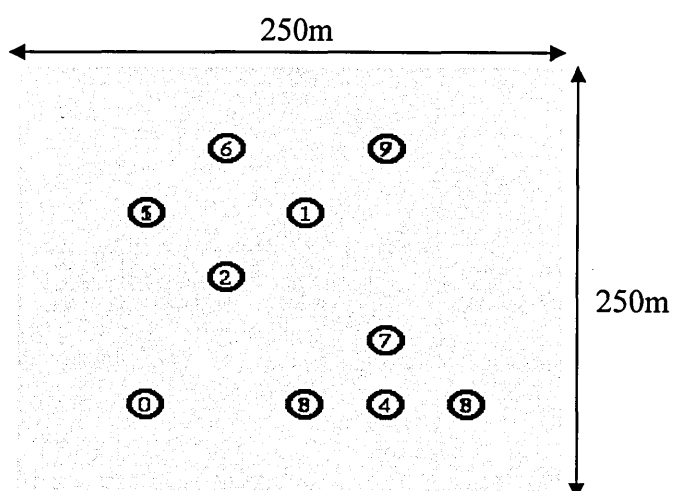


Figure 5.5: Network topology.

The first experiment was executed for the audio application with the applications characteristics as shown in Table 5.3. For the videoconferencing experiment, the same topology was used with the applications characteristics as illustrated in Table 5.4.

Table 5.3: Audio applications characteristics.

Traffic	Packet Size [byte]	Generation Rate [Kbps]
Audio 1	160	64
Audio2	160	64
Audio3	160	64
Cross-traffic 1	500	800
Cross-traffic2	600	500

Table 5.4: Videoconferencing applications characteristics.

Traffic	Packet Size [byte]	Generation Rate [Kbps]
Videoconf. 1	512	384
Videoconf.2	512	384
Videoconf. 3	512	384
Cross-traffic 1	300	150
Cross-traffic2	500	200

Another application for the proposed QoS evaluation systems is the assessment of the audio and videoconferencing applications through multihop wireless paths and the study of the capacity of a mesh network. The network topology and simulations were done for single, two, three, and four hops. The distance between two neighbouring nodes was 200 meters. The simulations were done by varying the load rate by increasing the number of connections (sources) from 1 to 8 and all sources started and finished simultaneously for every hop experiment.

5.4.4 Analysis Steps

Once the topology was selected and the traffic was configured, the main QoS parameters (metrics) that were important for the application under consideration (i.e. evaluation) were quantitatively evaluated and analysed based on the widowing technique mentioned earlier. The procedures was continued as illustrated in Figure 5.6 to get the QoS value for each window to assess the QoS value for a multimedia application using the fuzzy logic and distance approaches.

Finally, after processing the QoS parameters by the assessment systems, a vector of output values was produced. This vector represented the evaluated QoS of each application. This output characterised how the network dealt with the applications. This vector was further processed and analysed as discussed in Section 4.4 of Chapter 4. Then, to determine how the network treated the application as a whole, the QoS output vectors of each application running over the network were gathered in one vector. These vectors were also analysed using the same procedure of Section 4.4.

Network simulation and
transmission of the applications

Data file output (trace file
generation)

QoS parameters measurement
& calculations

Windowing and processing of
the required QoS parameters

Distance System
(Figure 5.4)

Fuzzy System
(Figure 2.7)

Analysis & processing

Overall assessed QoS

Figure 5.6: Flow chart of QoS assessment procedures using fuzzy and distance approaches.

5.5 Results and Discussions

In this work, an assessment of two important multimedia applications (time and loss sensitive) was carried out in the presence of cross-traffic. For each application; delay, jitter and packet loss were measured and processed using the windowing technique mentioned before to get the average value of each window. The instantaneous and the average delay obtained by blocking every 10 successive packets are shown in Figure 5.7a and 5.7b, respectively.

0 500 1000 1500 2000 2500 3000
Packet Sequence Number

(a)

0 50 100 150 200 250 300
Block Nunber

(b)

Figure 5.7: (a) Instantaneous and (b) Average delay of audio 1 application using windowing technique.

5.5.1 Audio Application

Once the measured average QoS parameters (delay, jitter and loss) for each of the three audio applications were obtained, they were fed to the fuzzy and distance systems to produce the QoS of each application. In order to test the output accuracy of the two assessment systems, combinations of samples of the input parameters were taken and processed by them. These samples and their corresponding outputs are illustrated in Tables 5.5 and 5.6 using fuzzy and distance systems, respectively. From these Tables, it can be seen that the output QoS values are a reflection of the input parameters based on the fuzzy rules shown in Figure 5.3 and the proposed procedure for the distance system. The outputs of both systems for audio applications are shown in Figures 5.8-5.13. From Tables 5.5 and 5.6 and Figures 5.8-5.13, both assessment systems provided results, which are comparable to each others. Some of these outputs are different they are in the same QoS region (i.e. Good, Average, or Poor). The discrepancies between the two methods were due to the different procedure followed by them. From the figures, distance system showed a higher variation and transitions than the fuzzy system. That was due to the fact that the fuzzy system is intelligent and governed by membership functions, Gaussian in our case, which may provide smooth transitions between the system states. On the other hand, the distance evaluation system is a non-intelligent approach, which mainly depends on the difference between the measured parameters values and the required thresholds and then combining (adding) the differences that will produce direct crisp values without any fuzzification.

Table 5.5: Sampled input QoS parameters with their expected QoS (Audio QoS fuzzy system evaluation).

Delay [msec]	Jitter [msec]	Loss [%]	Evaluated QoS [%]	QoS Level
20	0.65	0.98	87.9	Good
60	0.75	2.79	72.6	Good
50	2.2	0.88	79.4	Good
200	0.85	0.95	83.2	Good
70	2.4	2.8	47.8	Average
300	0.5	2.5	51.2	Average
200	2.3	1.1	58.9	Average
250	3	2.7	44.9	Average
480	0.75	0.85	13.3	Poor
75	4	0.98	18.3	Poor
400	1.8	5.3	19	Poor
550	4.3	5.5	9.73	Poor

Table 5.6: Sampled input QoS parameters with their expected QoS (Audio QoS distance system evaluation).

Delay [msec]	Jitter [msec]	Loss [%]	Evaluated QoS [%]	QoS Level
20	0.65	0.98	89.8	Good
60	0.75	2.79	82.2	Good
50	2.2	0.88	83.9	Good
200	0.85	0.95	85.3	Good
70	2.4	2.8	53.4	Average
300	0.5	2.5	52.2	Average
200	2.3	1.1	55.6	Average
250	3	2.7	43.8	Average
480	0.75	0.85	28.2	Poor
75	4	0.98	28.1	Poor
400	1.8	5.3	25.9	Poor
550	4.3	5.5	10	Poor

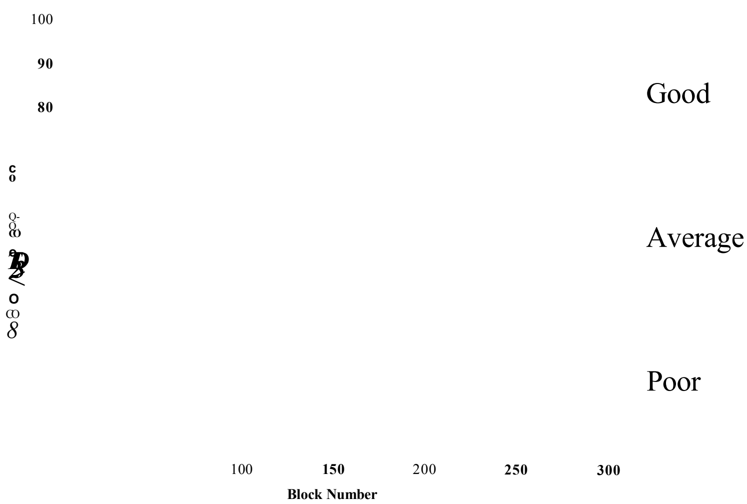


Figure 5.8: The output QoS of Audio 1 application using the fuzzy system.

Good

Average

Poor

0 100 200 300 400 500 600 700 800

Block Number

Figure 5.9: The output QoS of Audio 1 application using distance system.

90

Good

Average

Poor

100 150 200 250 300

Block Number

Figure 5.10: The output QoS of Audio2 application using the fuzzy system.

Good

Average

co 40

Poor

0 100 200 300 400 500 600 700 800

Block Number

Figure 5.11: The output QoS of Audio2 application using the distance system.

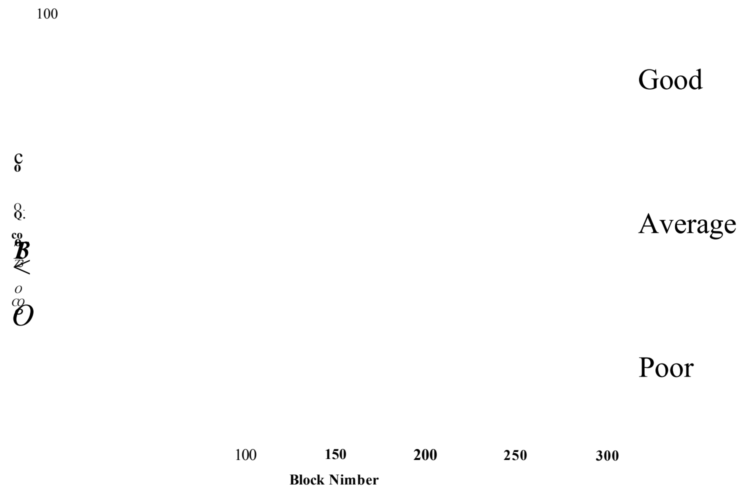


Figure 5.12: The output QoS of Audio3 application using the fuzzy system.

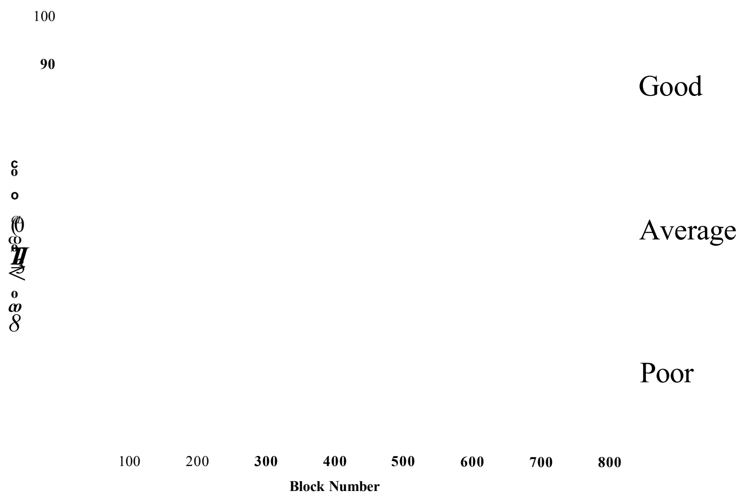


Figure 5.13: The output QoS of Audio3 application using the distance system.

The figures indicate that the fuzzy system provided QoS values in the range of [10%-90%] while the distance system generated QoS in the range of [0%-100%], Therefore, the fuzzy system could not provide a QoS value less than 10% and a maximum value greater than 90%. The cause of this effect was due to the overlaps between the input membership functions and between the output membership functions, which affected on the performance of the fuzzification and the defuzzification processes. On the contrary, the distance system, as mentioned before, relies on combining the differences between the measured and the desired values and therefore could produce an output range of [0-100] based on linear transformation.

Given that the maximum value of QoS is 100%, it can be observed, from these figures and for both assessment systems that the QoS was fluctuating between good, average and poor values. Good QoS was a result of low measured values of the QoS parameters or when two parameters were low and the other was medium. While QoS is considered poor if any of the QoS parameters was high regardless of the other parameter values. Otherwise, the measured QoS was average. The fluctuation between the three regions reflected the availability of the network resources based on the number of the applications which are sharing these resources.

In addition, it can be seen that there is a high variation in the evaluated QoS, especially for the Audio1 application. This variation is a result of high variation in the measured QoS parameters (delay, jitter and loss). In the simulations, the data rate of the audio application was very low (64Kbps), which means that they were not bandwidth-hungry applications that scarcely compete for bandwidth in the network. Moreover, the queue size used was 50 packets. Due to these, zero losses for the three applications were measured. Therefore, the measured losses met the audio losses requirement (<3 percentage) which means that the provided QoS is good with respect to the losses. Therefore, the variation in the assessed QoS by both systems may be due to variation in delay and jitter. However, the maximum measured average delay did not exceed 150 ms, which is the maximum audio delay requirement to get a good (high audio quality) QoS. This includes queuing delay, transmission delay, propagation delay, retransmissions at the MAC layer and processing delay. From this, and as in packet loss case, the measured QoS with regard to delay is also good.

Form the above discussion; it can be deduced that the variation in QoS was mainly due to variations in the measured jitter values, which some times exceed the audio jitter requirements to provide a good QoS. That was because the high quality audio application jitter requirement is very hard to be achieved (< 1ms) in the default DCF since only a best-effort service is provided. The variation of the jitter is due to the contention between the sending nodes for the available resources of the network. This contention will enforce the nodes to defer their transmissions for some times like Short Inter Frame Space and DCF Inter-frame Space (SIFS and DIFS) during the busy times of the network channel because it was occupied by some other nodes. The deferral of transmitting some packets will cause some variations in the delays of the consecutive packets. This will produce excessive delay variation (jitter), or intermittent but noticeable jitter at the receiving side, which will degrade the overall audio quality. In addition to that, due to congestion in the node queue, this leads to a variation in the queuing delay. Moreover, a high collision rate and frequent retransmissions cause unpredictable jitters. This will result in an increased network jitter, which can be very significant. Therefore, the assessed QoS of the application quickly deteriorates.

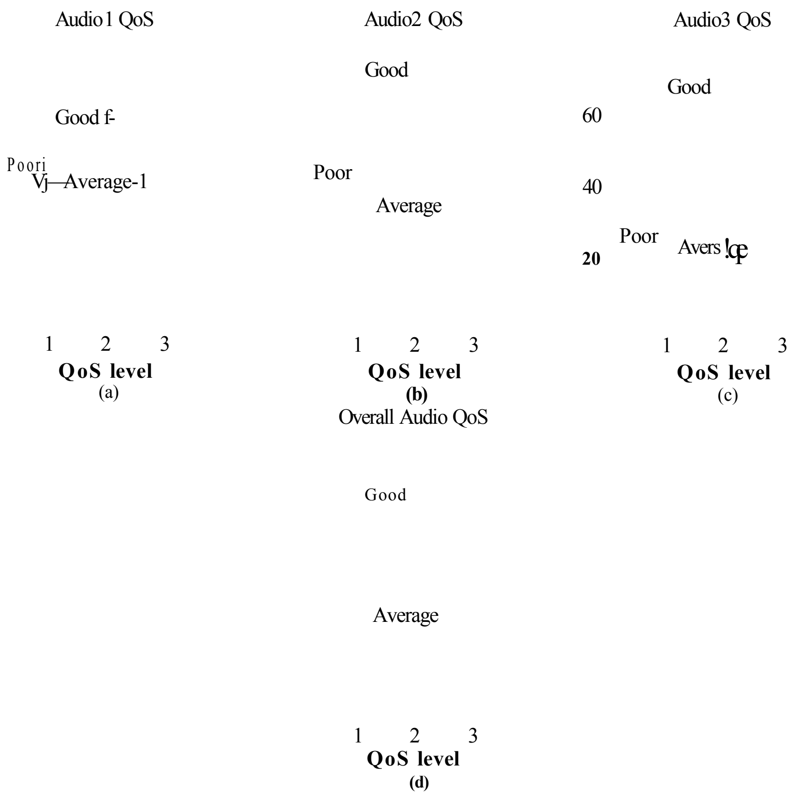


Figure 5.14: The bar chart for: (a) Audio 1, (b) Audio2, (c) Audio3 applications QoS and (d) the overall QoS using the fuzzy assessment system.

To show the extent that the QoS was poor, average, and good, bar chart distribution was used. The length of the bar was representative of the percentage of each QoS case. Figures 5.14 and 5.15 depict the bar charts for the Audio 1, Audio2 and Audio3 applications QoS, and the overall QoS using the fuzzy and the distance approaches, respectively. The overall QoS represents the QoS of the audio applications over the network. In order to identify how much the QoS was poor, average and good and to show the variation of these values, the mean and standard deviation were calculated. Tables 5.7 and 5.8 illustrate the statistics that characterise each QoS region of each application and the overall QoS. From these figures and tables, it can be seen that Audio3 showed the best QoS. Fuzzy logic indicated that about 80% of Audio3 QoS values was in the Good region with average value of 84.48% and less than 11% was in the Poor and Average regions. Similarly, the distance system indicated that 80% of Audio3 QoS was Good but with average value of 98.69% and about 12% and 8% was in the Poor and Average regions, respectively. It can be observed that both assessment systems gave, relatively, similar results regarding the QoS level of the Audio3 application. Nevertheless, the average values were different due to the reasons mentioned earlier. The overall audio QoS was good because around 60% of its values were in the Good region with average values of 82.56% and 98.07% from fuzzy and distance systems, respectively. It can be observed that this method provided a good picture about the measured QoS regions statistics and percentages.

Audiol QoS

Audio2 QoS

Audio3 QoS

Average

Overall Audio QoS

Average

QoS level

(d)

Figure 5.15: The bar chart for: (a) Audiol, (b) Audio2, (c) Audio3 applications QoS and (d) the overall QoS using the distance assessment system.

Table 5.7: Statistics of each audio application region QoS and the overall QoS using fuzzy logic assessment system.

	Audiol QoS			Audio2 QoS			Audio3 QoS			Overall Audio QoS		
	Poor	Average	Good	Poor	Average	Good	Poor	Average	Good	Poor	Average	Good
Mean [%]	12.4	52.2	81.3	12.1	53.3	81.2	11.2	54.2	84.5	12.1	53	82.6
Std. Dev. [%]	4.9	11.7	5.2	5.1	13	3.6	4.8	10.7	4.4	5	11.9	4.7

Table 5.8: Statistics of each audio application region QoS and the overall QoS using distance measure assessment system.

	Audiol QoS			Audio2 QoS			Audio3 QoS			Overall Audio QoS		
	Poor	Average	Good	Poor	Average	Good	Poor	Average	Good	Poor	Average	Good
Mean [%]	21	43.1	97.5	20.2	42.7	97.7	18.4	42.9	98.7	20.1	42.9	98.1
Std. Dev. [%]	7.8	6.1	2.7	8.1	6.1	2.7	8	7.2	1.8	8	6.2	2.5

In order to produce a more specific picture about the QoS of each application and the overall audio QoS without classification of the QoS values into good, average and poor regions, equation (4.4) was used to generate the distributions of each QoS. As an example, the distributions of Audio3 are shown in Figures 5.16 and 5.17. The figures illustrate the cumulative distributions, $Pr\{X < a\}$, where the random variable X denotes the end-to-end QoS. The usefulness of this method stems from the fact that it gives the probability that the QoS is less than any threshold value in the 0 to 100 percentage range. For example, it can be seen from the figures that it is very easy to assess the probability or how many values of the QoS were less than 30%. These are 0.28, 0.24, 0.12 and 0.21 for Audiol, Audio2, Audio3 and overall audio QoS, respectively using the fuzzy assessment system. Moreover, the values are 0.2, 0.2, 0.12 and 0.18, respectively, using the distance system, which are comparable to the fuzzy system results. In addition, it can be observed that the minimum and maximum values of the QoS can be found from these figures. For example, the minimum value for all audio applications was between 9 and 10% based on both systems.

Moreover, to provide more general representation of the QoS of each audio flow over the network and how the network treated the audio application in general, averaging or normalisation method (equation 4.1) can be used. Tables 5.9 and 5.10 summarise the results of using these methods for each audio application and for the overall QoS of the audio performance over the network depending on the fuzzy logic and the distance evaluation systems.

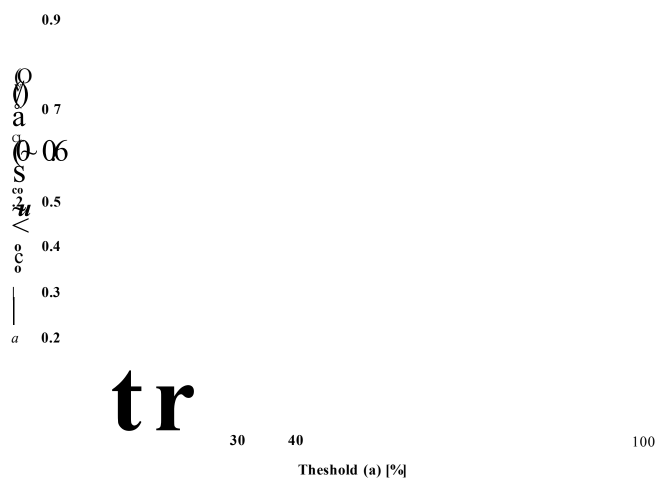


Figure 5.16: Audio3 QoS distribution based on fuzzy approach results.

Figure 5.17: Audio3 QoS distribution based on distance approach results.

From Tables 5.9 and 5.10, it can be seen that there is a small discrepancy, in the results of the QoS of each application and overall between the two methods (i.e. mean and normalisation) and for both assessment systems. However, due to some variations in the QoS output of the assessment systems; the averaging method may be not very suitable in these situations because some high and low values will bias the final result. On the other hand, the normalisation approach might be the most suitable method for the evaluation of the QoS of each application and the overall one. That is because it eliminates the variations in the values and takes these values in account in calculating the overall QoS. Both systems provided comparable results.

Table 5.9: QoS of audio applications and the overall audio QoS using the fuzzy system.

Units [%]	Audiol QoS	Audio2 QoS	Audio3 QoS	Overall Audio QoS
Mean	56.6	60.4	72.9	63.2
Normalisation	58.3	64	78.8	66.4

Table 5.10: QoS of audio applications and the overall audio QoS using the distance system.

Units [%]	Audiol QoS	Audio2 QoS	Audio3 QoS	Overall Audio QoS
Mean	62.9	67.8	81.7	71.2
Normalisation	58.4	64.2	81.7	67.8

In wireless networks, the standard DCF can only support best-effort services without any kind of QoS guaranties. In this mode, all sources in a basic service set compete for the resources and channel with the same priorities. As an application example of the proposed QoS evaluation system, a measurement of the performance, ability, and capacity of the 802.11 standard DCF mode to deliver QoS of audio application (i.e., number of audio connections that the 802.11 DCF mode can provide with a Good and Average QoS) if the network was only used to transmit audio. The simulations were performed by increasing the number of connections (sources) from 1 to 8 and all of them started and finished simultaneously. The G.711 voice encoding scheme was used, which generates 160 byte at 20ms intervals resulting in 64 Kbps stream (i.e., 8KB/s) (ITU, 1988). All the sources and destinations were in the same BSS. The simulation results for the delay, jitter, the overall QoS using fuzzy system and overall QoS using the distance system are shown in Figures 5.18(a)-(d), respectively.

0 2 4 6 8 10

Number of audio streams

(a)

0 2 4 6 8 10

Number of audio streams

(b)

0 1 2 3 4 5 6 7 8 9

Number of audio streams

(c)

10

0 1 2 3 4 5 6 7 8 9

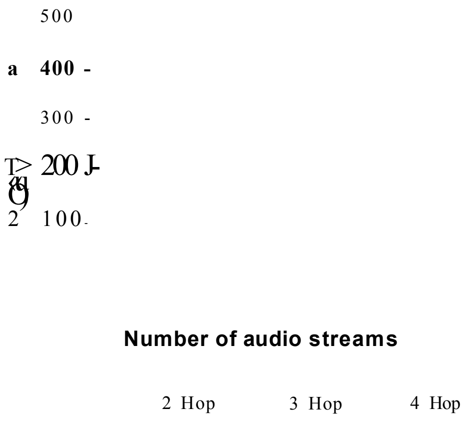
Number of audio streams

(d)

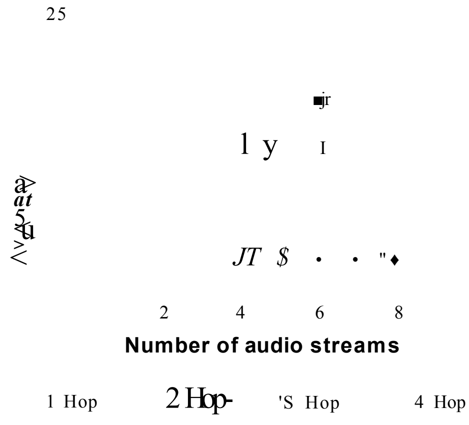
Figure 5.18: Overall average QoS parameters and the overall QoS for audio in the same BSS: (a) delay, (b) jitter (c) QoS using the fuzzy system and (d) QoS using the distance system.

The measured packet loss ratio was zero over the whole simulations. From the above figures, it can be seen that the average delay and loss are just below the "good" QoS requirements for all of the channel load rates. In contrast, the average packet loss ratio was zero due to the low generation rate of the applications. However, for the average jitter, as the number of c onnections (load rate) increased, the average jitter increased rapidly, which means as the number of sources was increased, they experienced higher jitter. In addition, and as a result, the overall average QoS also decreased sharply with increasing number connections. This decrease was mainly due to the jitter because the jitter was the only parameter, which exceeded the audio QoS requirements. Using both assessment systems, it is apparent that, the standard 802.11 DCF mode can only provide just 4 audio sources with Good QoS and 3 sources with Average QoS. In this experiment, any increase in the number of connections would produce a Poor QoS.

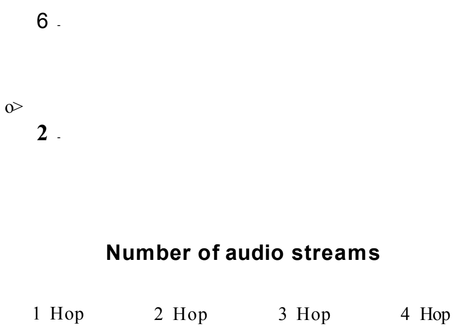
Another application for the proposed QoS evaluation system is the assessment and evaluation of the delivery of audio application through multihop wireless paths and study of the capacity of the mesh network. The network topology and simulations were run for one, two, three, and four hops and the distance between two neighbouring nodes was 200 meters. The simulations were executed by varying the load rate by increasing the number of connections (sources) from 1 to 8 and all of them started and finished simultaneously for every hop experiment. The results of these experiments are illustrated in Figure 5.19. From this Figure, it can be observed that as the length of the transmission path increases (i.e., number of hops), the performance degrades and the average delay, jitter, and loss increase and the overall QoS decreases as evaluated using the fuzzy and distance systems. For the single hop experiment, it is apparent that all the measured QoS values were always in the Good region. This was because all the measured QoS parameters were very small and within the Good audio application QoS requirements. The two hop experiments showed that the measured QoS was distributed in the Good and the Average regions. As the number of audio sources increases, the QoS decreases but it is mostly in the Average region. The three hop experiments illustrates that the QoS was in the Poor region except when the network had one or two sources, it was in the Good region. This was mainly due to the high values of the jitter as increasing the number of connections. For the four hops, it is clear that all the measured QoS were in the Poor region except for one source it was in the Good region. The poor QoS was because, mostly, all the parameters have experienced high values, which exceeded the Good and the Average QoS requirements.



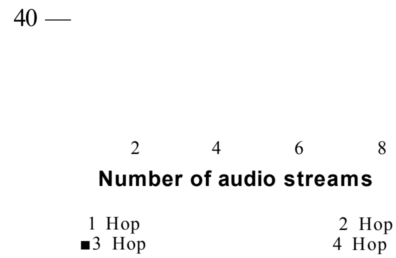
(a)



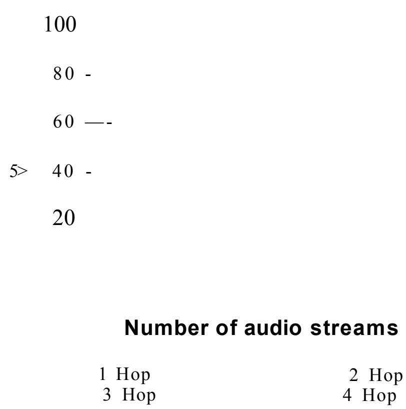
(b)



(c)



(d)



(e)

Figure 5.19: Overall average QoS parameters and the overall QoS for audio in multihop Ad hoc network: (a) delay, (b) jitter c) losses, (d) QoS using fuzzy system and (e) QoS using distance system.

As in the BSS experiments, the jitter was the main parameter which degraded the audio QoS. Here, in the multihop network, the high jitter values were mainly due to packet collision which occurred mainly due to the hidden nodes which are located within the transmission range of the receiver but not of the sender. In addition, jitter was also high due to high competition between the sending nodes for the available resources which caused collision and packet dropping in the IFQ. Also, additional delay and jitter incurred due to the retransmission process of the lost packets because the node did not consider the packet loss until the retry transmission limit was expired.

5.5.2 Videoconferencing Application

The procedure, which was adopted to evaluate the QoS of the audio application, was also used to assess the overall QoS of the videoconferencing multimedia application. After measuring and calculating the average QoS parameters (delay, jitter and loss), these parameters were input to the videoconferencing evaluation systems (i.e. fuzzy and distance systems) to get the QoS for each application. To verify that this system was efficient and complied with the fuzzy rules illustrated in Figure 5.3 and distance assessment system expectations, a sample from the averaged input parameters and their assessed QoS output were taken. These samples are shown in Tables 5.11 and 5.12. The assessed overall QoS of the three videoconferencing applications are depicted in Figures 5.20-5.25 based on both proposed assessment systems.

Table 5.11: Sampled input QoS parameters with their expected QoS (Videoconferencing fuzzy system evaluation).

Delay [msec]	Jitter [msec]	Loss [%]	Evaluated QoS [%]	QoS Level
20	5	0.60	85.9	Good
60	6	1.20	74.9	Good
80	11	0.67	74.3	Good
200	5	0.60	79.9	Good
70	18	1.2	45.4	Average
300	4.5	1.3	51.1	Average
200	15	0.8	57.6	Average
250	17	1.5	45.5	Average
530	8	0.5	10	Poor
100	23	0.8	27.9	Poor
400	20	2.6	18.5	Poor
550	23.3	2.2	9.8	Poor

Similar to the results obtained for audio application, both systems performed well in assessing the QoS and generally, they produced comparable outputs. However, there are some differences between the results attained using the fuzzy system and those attained by the distance system which are mainly represented in QoS values in the ranges [10%-90%] for fuzzy and [0%-100%] for distance. Nevertheless, these differences are not high, which will not lead to different QoS assessment of the multimedia application (i.e.

at least they gave QoS values which are in the same region or level). The reasons behind these discrepancies in the videoconferencing QoS assessment are the same reasons discussed earlier in Section (5.5.1) for the audio QoS evaluation.

Table 5.12: Sampled input QoS parameters with their expected QoS (Videoconferencing distance system evaluation).

Delay [msec]	Jitter [msec]	Loss [%]	Evaluated QoS [%]	QoS Level
20	5	0.60	89.1	Good
60	6	1.20	83.5	Good
80	11	0.67	80.1	Good
200	5	0.60	82	Good
70	18	1.20	44	Average
300	4.5	1.25	51.9	Average
200	15	0.80	51.9	Average
250	17	1.5	46.2	Average
530	8	0.50	21.6	Poor
100	23	0.75	30.8	Poor
400	20	2.60	25.2	Poor
550	23.3	2.22	15.3	Poor

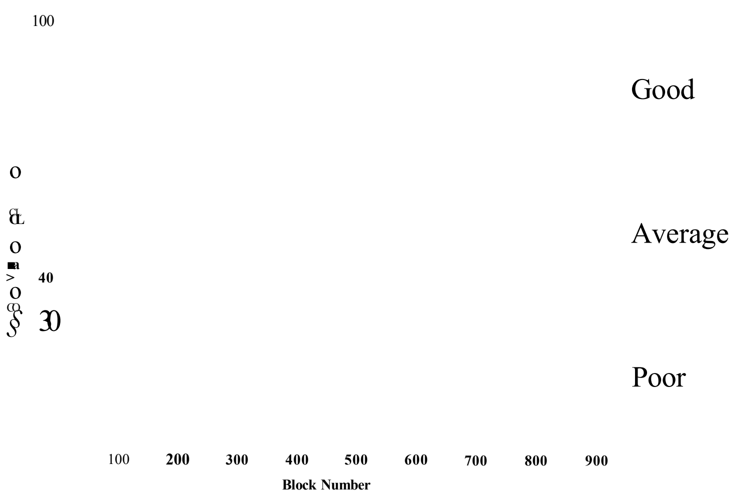


Figure 5.20: The output QoS of videoconferencing 1 application using fuzzy system.

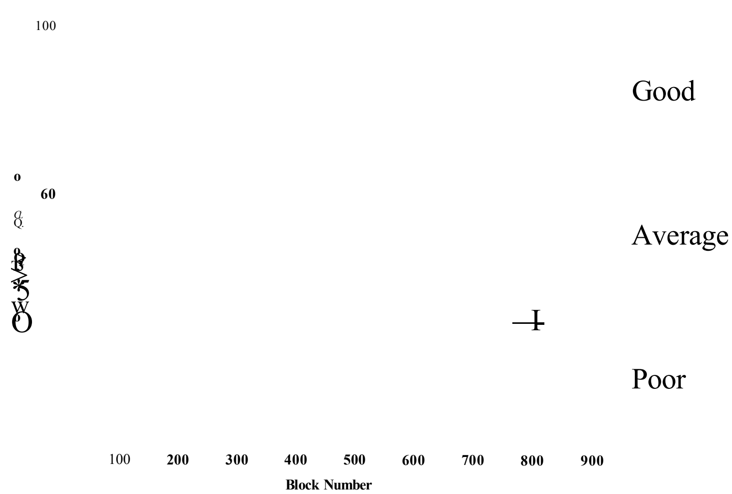


Figure 5.21: The output QoS of videoconferencing 1 application using distance system.

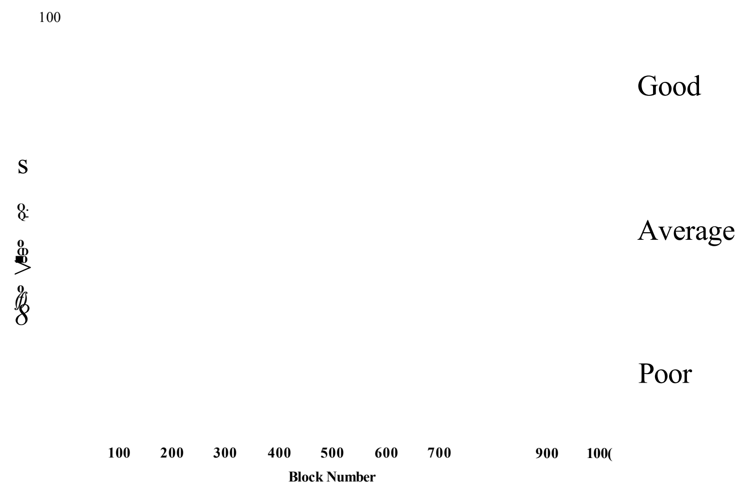


Figure 5.22: The output QoS of videoconferencing2 application using fuzzy system.

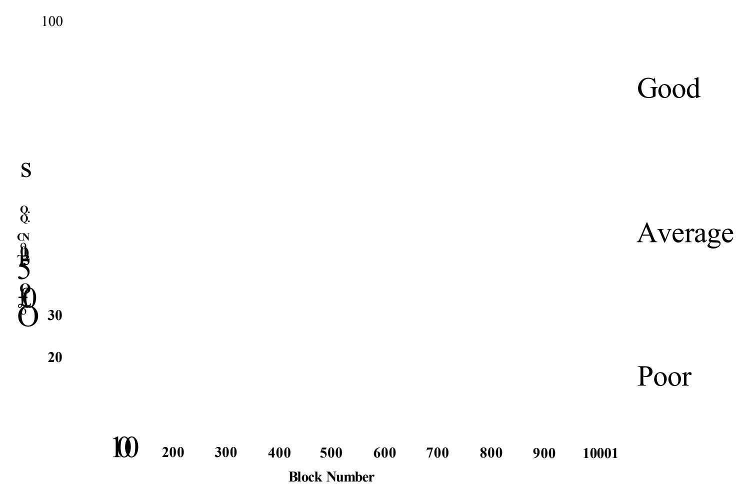


Figure 5.23: The output QoS of videoconferencing2 application using distance system.

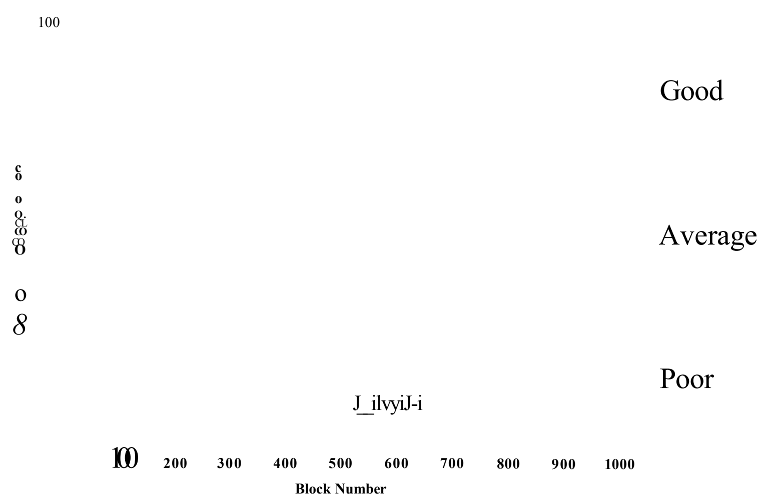


Figure 5.24: The output QoS of videoconferencing3 application using fuzzy system.

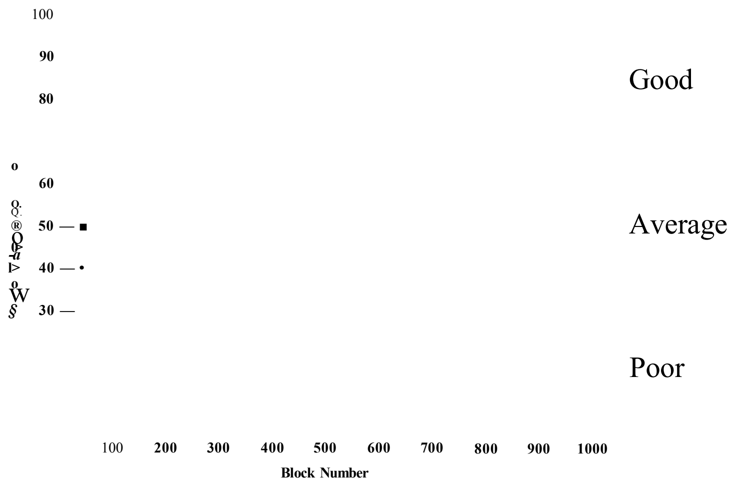


Figure 5.25: The output QoS of videoconferencing3 application using distance system.

As for audio traffic, it can be seen from the figures that there are fluctuations in the measured QoSs due to wireless networks characteristics like links and resources with variations over time. Therefore, the measured QoS will provide a measure of the network resources availability. The variation in the availability of these resources contributed directly for the variation of the output QoS values.

In addition, from the figures; it can be observed that there was a very high competition among the five flows in the network (three videoconferencing and two background traffics). This competition was due to the fact that the default DCF does not support any QoS guarantees. In DCF all the flows compete for the channel with the same priority without any differentiation mechanism. Due to this, during the majority of the simulation period, the measured parameters had high values, which caused degradation in the output QoS. In order to justify and determine the reasons behind this degradation, the measured parameters need to be carefully examined. Firstly, most of the average jitter values were less than the good videoconferencing jitter requirement. This means that the degradation was due to delay and losses. It was seen that most delay values varied between medium and high values, whereas the loss values were low or high. From this and using the two assessment systems, it can be deduced that all the average QoS values were due to medium delay and loss values and poor QoS was due to high delay, high losses or high delay and losses. Packet loss may occur at both network and the MAC layers due to transmission errors, broken link, congestion, or collisions. These are associated with the network conditions (e.g., number of connections, traffic load, and application type). Network layer losses are usually due to routing problems, but in our case a single hop network was used, so all the losses were at the MAC layer. As

CSMA/CA was used in the simulation, a packet may be dropped due to congestion for two reasons. Firstly, if the wireless channel was too busy, the back-off time might exceed the limit when the demands surpassed the maximum capacity of the communication link. Alternatively, when the channel was associated with the queue, which buffered all the packets waiting to be sent, all the incoming packets were dropped when the queue was full. In addition, some of the high packet losses were due to collisions because every node had to wait for a random amount of time before trying to send a packet. The collisions occurred when two nodes started transmitting simultaneously.

The high values of delay were due to the contention between the sending nodes for the available resources of the network. This contention will enforce the nodes to defer their transmissions for some times like Short Inter Frame Space (SIFS) and DCF Inter-frame Space (DIFS) during the busy times of the network channel because some other nodes occupied it. The deferral of transmitting some packets will cause excessive delay at the receiving side, which will degrade the overall quality. In addition, congestion in node queue, leads to an increase in the queuing delay. Losses due to congestion and collisions introduce another delay due to retransmissions of the lost packets at the MAC layer. The result is that both loss and delay can be very significant. Therefore, the assessed QoS of the application quickly deteriorates.

In order to determine the QoS level or grade of the videoconferencing, the bar chart distributions were utilised. Figures 5.26 and 5.27 show the QoS bar chart of videoconferencing1, videoconferencing2, videoconferencing3 applications, and the overall QoS based on the fuzzy and distance approaches, respectively. The overall QoS represents the average QoS of the three videoconferencing applications over the network. In order to recognise how much the QoS of each application was poor, average and good and the variation of these values, the mean and standard deviation were calculated. Tables 5.13 and 5.14 illustrate the statistics for each region of QoS and the overall QoS. In addition, from figures 5.26 and 5.27, it can be, seen that both assessment systems provided similar outputs. The three videoconferencing applications had nearly the same QoS for every region. In general, the overall QoS was poor because 60% of the QoS values were in the poor region with an average value of 14.5%, revealed by the fuzzy system. While the distance system showed that the overall QoS was also poor but with a percentage of 54% of the QoS values and with an average of

20.4%. Tables 5.13 and 5.14 showed that there are some differences also in the assessed average QoS values by the fuzzy and the distance systems. These differences were due to the procedure followed by the systems in the assessment approach as discussed earlier. It can be observed that this method provided a good representation of the measured QoS statistics and percentages.

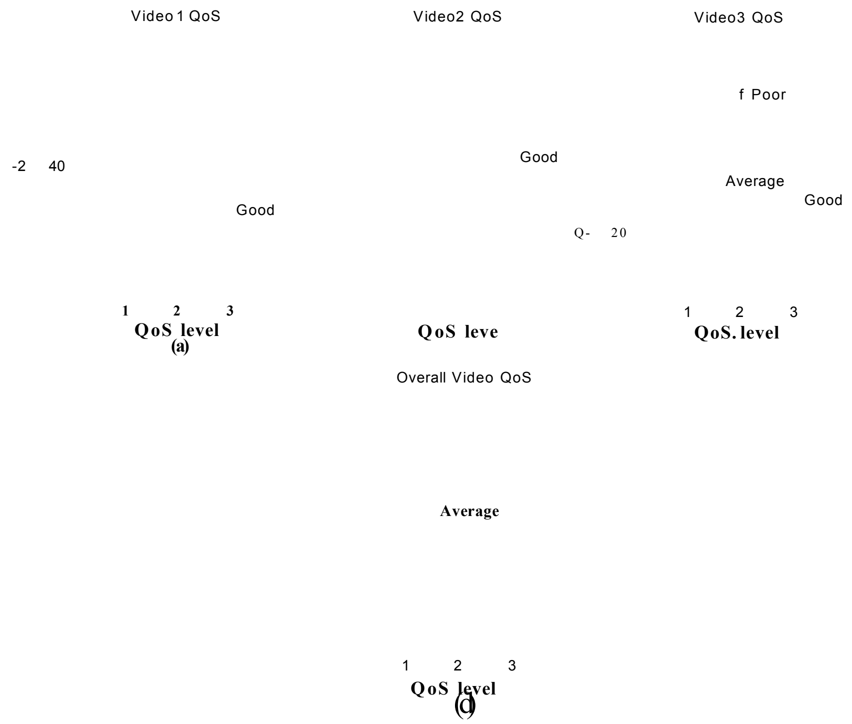


Figure 5.26: The bar chart for: (a) videoconferencing l, (b) videoconferencing2, (c) videoconferencing3 QoS and (d) the overall QoS using the fuzzy system.

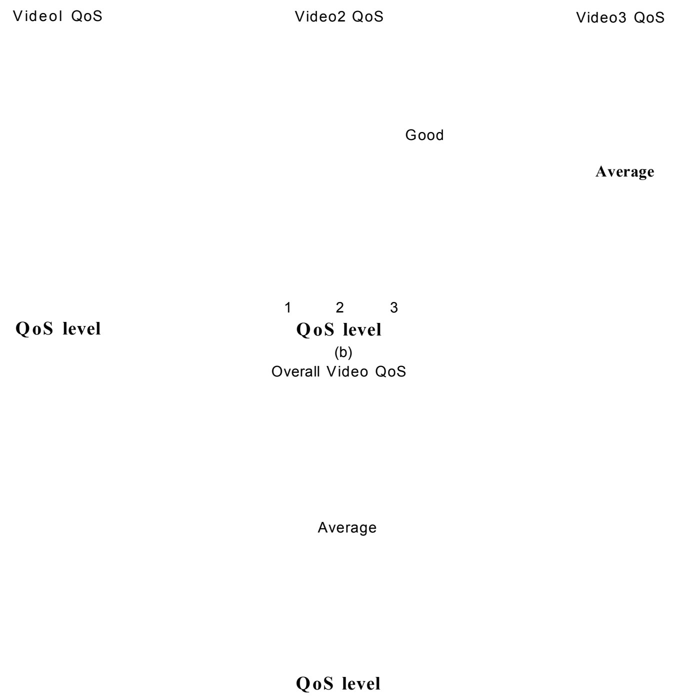


Figure 5.27: The bar chart for: (a) videoconferencing l, (b) videoconferencing2, (c) videoconferencing3 applications QoS and (d) the overall QoS using the distance system.

Table 5.13: Statistics of each videoconferencing application region QoS and the overall QoS using the fuzzy logic assessment system.

	Videoconfl QoS			Videoconf2 QoS			VideoconD QoS			Overall Videoconf QoS		
	Poor	Average	Good	Poor	Average	Good	Poor	Average	Good	Poor	Average	Good
Mean [%]	14.5	50.6	77.2	13.8	50.2	81.3	15.2	49.5	83.8	14.5	50	81
Std. Dev. [%]	6.4	9.7	6	5.9	9.9	7.6	6.8	10	7.5	6.4	9.9	7.6

Table 5.14: Statistics of each videoconferencing application region QoS and the overall QoS using the distance measure assessment system.

	Videoconfl QoS			Videoconf2 QoS			Videoconf3 QoS			Overall Videoconf QoS		
	Poor	Average	Good	Poor	Average	Good	Poor	Average	Good	Poor	Average	Good
Mean [%]	20.7	40.2	93.3	19.6	40.4	94.6	20.8	39.4	94.7	20.4	40	94.3
Std. Dev. [%]	7.9	5.6	3.1	8.4	5.4	3	8.5	5.4	3.1	8.3	5.5	3.1

Moreover, to provide a good representation of the QoS for each videoconferencing flow and to determine how the network treated each videoconferencing application, averaging and normalisation methods (equation 4.1) were used. Tables 5.15 and 5.16 summarise the results of using the QoS assessments for each videoconferencing application and for the overall QoS of the videoconferencing performance over the network with the fuzzy and distance evaluation systems, respectively.

Table 5.15: QoS of each videoconferencing application and the overall videoconferencing QoS using fuzzy system.

Units [%]	Videoconfl QoS	Videoconf2 QoS	VideoconB QoS	Overall Videoconf QoS
Mean	31.5	38.1	35	34.9
Normalisation	27.3	35.4	32	31.6

Table 5.16: QoS of each videoconferencing application and the overall videoconferencing QoS using distance system.

Units [%]	Videoconfl QoS	Videoconf2 QoS	Videoconfl QoS	Overall Videoconf QoS
Mean	39.6	46.3	40.4	42.8
Normalisation	38.1	44.5	42.3	40.9

From Tables 5.15 and 5.16, it can be seen that there is a difference in the achieved results of the QoS of each application and the overall one using the two QoS assessment approaches (i.e. mean and normalisation). The evaluation using the averaging and normalisation methods gave close results. Nevertheless, averaging provided useful information but was not accurate enough due to some variations in the QoS output results, the averaging method is not very suitable in these situations because some high values and low values may bias the final result. On the other hand, the normalisation method might be the most appropriate method for the evaluation of the QoS of each application and the overall one. That is because it reduces the deviations in the values and takes these values into account when calculating the overall QoS. Both assessment systems offered nearly similar results. However, the distance method provided values higher than those obtained by the fuzzy system. That is because distance system output values in Good and Average regions have values higher than the fuzzy system values in the same regions.

As mentioned earlier, because the standard DCF can only support best-effort services without any kind of QoS guarantees, all sources in the same BSS will compete for the network resources with the same priorities. Another application example of the proposed QoS evaluation system is a measurement of the performance, ability, and capacity of the 802.11 standard DCF mode to deliver QoS of videoconferencing application if the network was only used to transmit this application. Simulations were carried out by increasing the number of connections from 1 to 8 in which sources started and finished their transmissions simultaneously. All the sources and destinations were in the same BSS. The simulation results for the overall average delay, jitter, packet loss, assessed QoS using fuzzy and the distance systems are shown in Figures 5.28(a)-(e), respectively.

Form these results, it can be seen that for only three streams, the average delay, jitter and losses are nearly below the good QoS requirements. In addition, as the number of streams increases to more than 3, a drastic decrease in the overall average QoS was observed as illustrated in Figures 5.28 (d) and (e). This sudden decrease was mainly due to a sharp increase in the delay and losses from 15 to 600 msec and from 0 to 10.8 %, respectively, which overtook the videoconferencing QoS requirements. These high values of delay and loss were because of the increase of the offered load in the network from 1152 kbps for three streams to 1536 kbps for four streams, which the standard DCF 802.11 mode cannot afford. Therefore, a high competition between the video sources will result in high collisions and so high losses and congestion due to retransmissions.

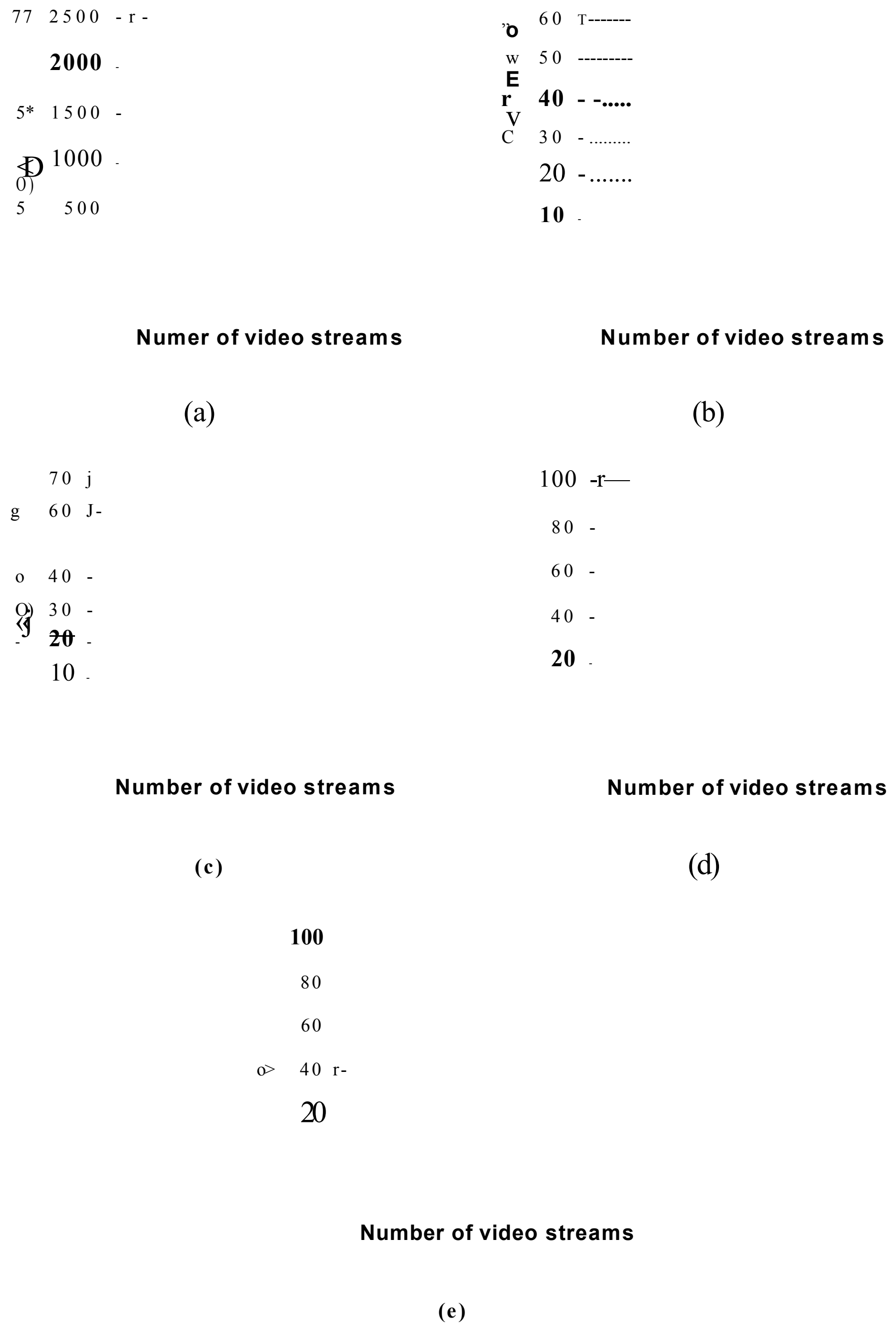


Figure 5.28: Overall average QoS parameters and the overall QoS for videoconferencing in the same BSS: a) delay, b) jitter c) losses, d) QoS using fuzzy system and (e) QoS using distance system.

In order to test how many hops that the DCF 80.211 can support for the videoconferencing application, a number of simulation experiments were conducted for multihop ad hoc network with distance between nodes set to 200 meters. The results of these simulations are depicted in Figures 5.29 (a)-(e). From these figures, the fuzzy and distance assessment systems revealed that the 802.11 can afford a good QoS for just three hops. If the destination needs more than three hops, the 802.11 will not provide a good QoS without using any kind of service differentiation mechanisms. This degradation in QoS was mainly due to a severe increase in the losses. For more than four hops, the drastic increase in the delay in addition to the losses caused additional decrease in overall QoS as reported by the evaluation systems. All of these problems were due to the hidden node problem in the ad hoc multihop network, which causes a high packets drop due to collisions, this also results in a high contention and so, a higher delay and so poor QoS.

^ 1200
w 1000
® 400

Number of hops

30 T-
SI 25 -
§ 2 0 -
| 15 —
& 10 !-

Number of hops

(b)

2 4 6

Number of hops

(c)

100

i
2 4 6 8

Number of hops

(d)

100 —♦—f
80 I ...j
60 -!-
40 ---------I-
20 -

Number of hops

(e)

Figure 5.29: Average QoS parameters and the overall QoS for videoconferencing in multihop Ad hoc network: a) delay, b) jitter c) losses, d) QoS using fuzzy system and (e) QoS using distance system.

5.6 Comparison between the Two Assessment Approaches

The two QoS assessment approaches demonstrated advantages and disadvantages during the evaluation process. In this section, the advantages and disadvantages of using them are summarised.

5.6.1 Fuzzy Assessment System

5.6.1.1 Advantages

Fuzzy assessment system has the following advantages:

- It provided a degree of membership for each QoS parameters and for the overall QoS (i.e. Good, Average and Poor),
- It provided smooth transitions between QoS values and regions,
- It is an intelligent system where the fuzzy rules are easy to write and to modify,
- It could be implemented in hardware.

5.6.1.2 Disadvantages

On the other hand, the fuzzy assessment system has the following disadvantages:

- Its parameters needed to be decided and designed accurately like the membership function parameters,
- It provided output QoS values in the range [10%-90%] rather than [0%-100%] range,
- It needed more processing time, and
- It required more memory requirements.

5.6.2 Distance Assessment System

5.6.2.1 Advantages

Distance assessment system has the following advantages:

- Simpler than the fuzzy system,
- It provided complete range of measured QoS [0%-100%],
- It required processing time less than the required for fuzzy logic system, and
- Different similarity measure types can be examined to identify the most suitable one.

5.6.2.2 Disadvantages

The distance assessment system has the following disadvantages:

- It had crisp transitions rather than smooth,
- It did not include or provide degree of memberships of the QoS parameters and the overall QoS, and
- It is not intelligent.

5.7 Summary

This chapter presented two methods to assess the QoS of multimedia applications: the Fuzzy assessment system and the Distance assessment system. The methods showed how the QoS could be measured without the necessity for analytical models. The measured QoS has been classified into Good, Average, and Poor categories. In addition, for each application, based on the proposed systems, the distributions and the overall QoS have also been obtained. The measured QoS using the two proposed evaluation systems was a good indication of the network conditions and resources availability.

In this chapter, the QoS measurement was continuously performed for the whole application traffic in which it is a resource, effort and time consuming. Therefore, it is essential to develop approaches, which can infer and deduce the network and the application performance to improve the efficiency of the measurement process. These are the objectives of the next chapters.

Combined Active-Passive QoS Monitoring Approach

6.1 Introduction

The performance of a network is of vital importance for both the service provider and the customer. Therefore, the QoS measurement process must be simple and accurate. In addition, this process must be fast enough so that it can reflect the QoS and the network performance in a timely manner. In general, methods for monitoring and measuring QoS and network performance are classified as either active or passive monitoring techniques. These techniques were deeply discussed in Chapter 3.

In order to overcome some of the disadvantages of both active and passive approaches, several studies were carried out by researchers. Some of these studies were based on a combination of active and passive methods. Change-of-measure based active/passive monitoring (CoMPACT) has been devised (Aida, et al., 2003), (Ishibashi, et al., 2004). This method was only used to estimate the actual user delay. Another technique has been proposed which combines passive and active methods (Lindh, 2002), (Lindh, 2001). In this technique, a router sends active probe packets at regular intervals. The passive monitoring method is used to count the number of user packets passing through the router. This approach has been used to estimate the QoS parameters only over wired networks.

The aim of this chapter is to describe the techniques that were devised to infer the performance of wireless ad hoc networks by considering the QoS requirements of multimedia applications based on the ideas of both active and passive methods. In this study, the aim of QoS monitoring and measurement was to assess the network performance for satisfying the requirements of user's applications. This approach uses an in-service measurement method in which the QoS of the actual application (user) is estimated by means of dedicated monitoring packets (probes) (Choi and Hwang, 2005). Afterwards, these parameters are combined to produce and assess the application's overall QoS using the fuzzy logic assessment and based on the measured QoS parameters estimated using the probe traffic. Therefore, the contribution of this chapter is represented by adding the process of the overall QoS assessment to the system utilised in (Lindh, 2002).

This chapter is organised as follows: Section 6.2 describes the monitoring approach description and the experimental simulation set up and settings. Section 6.3 presents the experimental results. Section 6.4 provides a summary of this chapter.

6.2 Monitoring Approach

6.2.1 Approach Description

The purpose of this work is to design a single monitoring system that can indirectly monitor and estimate the main actual user QoS parameters (delay, delay variation (jitter), packet loss and throughput) and the overall QoS/performance based on an artificial probe packet stream (monitoring packet stream). This approach combines both active and passive monitoring methods (Lindh, 2002). The active scheme is used to generate monitoring probe packets which are inserted between blocks of target application packets at regular intervals as shown in Figure 6.1. Based on these monitoring packets, the actual user delay and the jitter are estimated. While the passive monitoring is utilised to act as a traffic meter which performs as a counter of user packets (and bytes) that belong to the application (user) traffic flow that is subjected to monitoring. The combination between active and passive is utilised to infer the actual packet loss ratio and the throughput of the multimedia application. Active methods are not reliable for these measurements due to two drawbacks. Firstly, active methods inject a large number of probes to detect packet losses in the network which has a non-negligible load on the network. Secondly, the estimated packet losses based on probe packets may not be identical to that occurred to user packets. As a result, packet loss and throughput are passively measured depending on the active probes position. The method introduces the monitoring block, as can be seen in Figure 6.1, as a concept to attain higher resolution than the long term averages over the measurement period.

The probe packets are generated by a periodic single packet generation process. Periodic generation is quite attractive because of its simplicity and ease of implementation. The sending monitoring node generates a monitoring packet after every *M* number of user packets on average or within specific time intervals as depicted in

Figure 6.1 (Lindh, 2001). M is the number of actual traffic packets monitored between two successive monitoring packets. The generation process of the monitoring packet is a function of the selected monitoring block size or duration.

Monitoring packet ***n-1*** < -- ▶ Monitoring packet ***n***
Monitoring block of ***M*** packets (user data)

Figure 6.1: Two monitoring packets enclose a monitoring block that consists of M user packets.

In this work, in addition to the generation of the multimedia application, the transmitting node will be used to inject the monitoring packets. Thus, monitoring packets are intermittently dispatched and circulated on the user flow. These packets are interspersed with the user packets regularly to gather QoS information. For every monitoring packet generated, the sending node counts and then inserts the number of user packets sent so far and the timestamp at which this monitoring packet was generated. At the receiving end, the receiver node needs to maintain a counter for the number of the received user packets. In addition, it should: detect the monitoring packets, place a timestamp in every monitoring packet which shows the current time at the receiving end, and insert the current value of the counter that keeps track of the cumulative number of the received user packets. To achieve accurate timing between the sender and the receiver ends, clocks needs to be synchronised. This work is based on a simulation study, so all nodes are already synchronised. But, in reality, a synchronisation tool may be used to keep the nodes synchronised. Current solutions are to synchronise nodes to a specific reference time like the Coordinate Universal Time (UTC) using the GPS receivers, or Global Time Base (GTB) (Jiang, et al., 2000).

At the end, every monitoring packet should have, a sequence number, sending and receiving timestamps and the number (cumulative) of sent and received user traffic packets. The difference between the number of user sent packets on monitoring packet *n* and monitoring packet *n-1* gives the number of packets sent in the *nth* monitoring block and correspondingly for the number of user received packets as illustrated in Figure 6.1. Consequently, the difference between the sent and the received packets in the same monitoring block is the number of lost packets in that block. Lost monitoring packets are detected by the missing sequence number. If a monitoring packet is lost, the monitoring block will be extended up to the next monitoring packet that succeeds to arrive at the receiving node. In addition, a sample of the packet delay between the sending and the receiving nodes is given by the difference between the sending and the receiving timestamps of the monitoring packets. Jitter is calculated from the delay results. After measuring these parameters, they are fed to the fuzzy system to inform the user application QoS using the same procedures discussed in Sections 4.4 and sub-sections 5.4.1 and 5.4.4.

Based on the proposed approach, it is expected to obtain the following measures:

- Samples of the packet delay and jitter between the sending and receiving nodes.
- If the packet size is known, it is possible to estimate and monitor the throughput of the user application between monitoring packets rather than the long-term total average.
- The packet loss ratio of the user application between the sending and receiving nodes for each monitoring block.
- The length of the loss free periods and loss periods expressed in terms of the number of consecutive monitoring blocks that does not contain lost packets and the number of monitoring blocks that contain lost packets, respectively.
- Samples of the estimated QoS values of the user application based on the QoS parameters resulted from the probe measurements of each monitoring block.
- The length of the Good, Average and Poor QoS periods expressed in terms of the number of consecutive monitoring blocks that contains Good, Average and Poor QoS values.

6.2.2 Network Topology and Traffic Characteristics

To demonstrate the effectiveness of the proposed system, NS-2 was used to simulate the wireless ad-hoc network. The nodes were arranged in random positions and the arrangement was made in such a way that it satisfied the single hop condition with an area of (250m x 250m) using the same simulation protocols and settings discussed in Chapter 4. The traffic characteristics are illustrated in Table 6.1 with 500 second simulation time. The proposed approach is applied to approximate the QoS/performance of multimedia applications. As an example of multimedia applications videoconferencing was used. The network used in the simulation had six pairs of fixed source/destination. One of the pairs is used for videoconferencing application transmission and the others were used for the cross-traffic.

Table 6.1: Network traffic characteristics.

Traffic type	Packet Size [byte]	Generation Rate [Kbps]
Videoconferencing	512	384
Background traffic 1	400	300
Background traffic2	370	360
Background traffic3	420	330
Background traffic4	350	300
Background traffic5	600	450

The monitoring packets were CBR packets transmitted using the UDP protocol with a packet size of 64 bytes. The rate at which monitoring packets were sent is important. Too few packets result in inaccurate results and too many result in the network traffic being disturbed. Therefore, in order to examine the effect of probe rate on the QoS assessment, several probing rates were used ranging from low to high probe rates. Probe packets were transmitted periodically with monitoring block sizes *(M)* between the probe packets. *M* was selected to be 375, 186, 93, 47, 31 and 25 packets (i.e. ratio between probe and traffic packets is 1/375, 1/186, 1/93, 1/47, 1/31 and 1/25).

Over the simulation time and in order to examine the probe measurement results with different network conditions, the network was subjected to three different situations: light load (0-170 sec), medium load (171-330 sec) and fully loaded (331-500 sec).

All simulation experiments were repeated several times by using different seed values for the random number generator of the NS-2 simulator. Changing the seed random number essentially runs the same traffic, but will produce different timing for the simulation. The resulted values of the different runs of the same simulation have been averaged to get the actual values. In addition, each simulation was run twice for each seed; once with probe switched on and once with probes switched off. This allowed for testing the effect of the probe presence on the user and network behaviour.

6.3 Experimental Results

The performance of the monitoring procedure using the concepts of monitoring packets has been evaluated. This evaluation has been done for various probe rates and distances between the monitoring packets (i.e. the length of monitoring block).

The service quality was evaluated in terms of one-way delay and delay variation, packet loss rate, throughput and finally the overall assessed QoS. There are two comparisons that needed to be considered when assessing how good the probes are performing. Firstly, to assess how accurate the probe results are and secondly, to know how much the traffic is being affected by these probes (biasness).

6.3.1 Accuracy

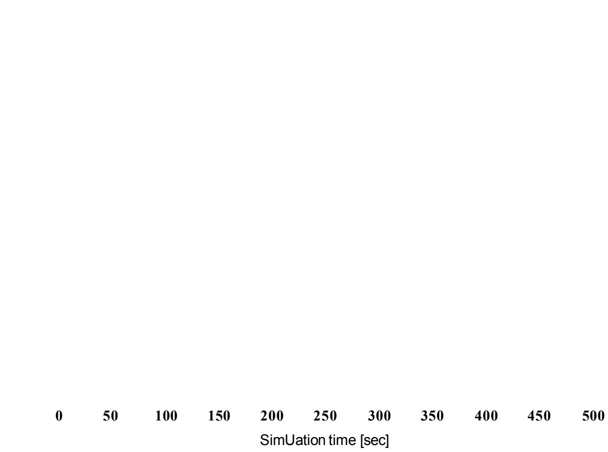
6.3.1.1 One-way Delay and Delay Variation

Figures 6.2(a)-(c) illustrate how the one-way delay varies during the measurement period for both user and probe traffics. In addition, Table 6.2 summarises the long-term actual and the estimated values (mean, maximum, minimum and standard deviation) for the delay and jitter for two different monitoring block sizes. As examples, two monitoring block sizes (i.e. probe rates) were used to compare the results of both traffics: 25 packets and 375 packets block sizes.

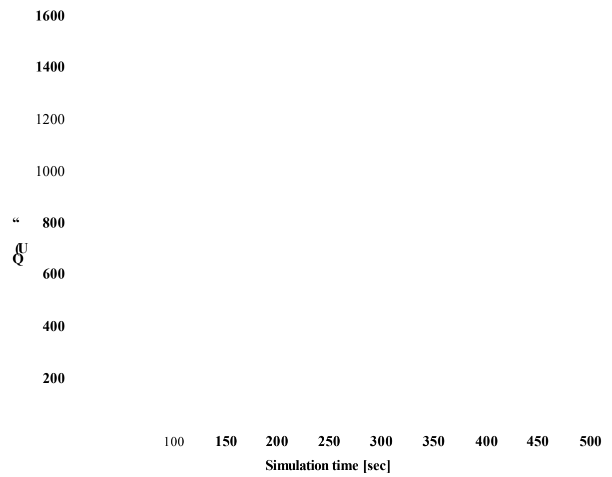
Table 6.2: The actual values for one-way delays and delay variations and the estimated values for block sizes using M - 25 and 375 packets.

Units: [msecj	Actual values	$M = 375$	$M = 25$
Mean delay	335.7	328.8	334
Absolute error		6.9	1.7
Delay St. Dev.	331.7	327.3	315.6
Maximum delay	1915.7	1274.6	1586.9
Minimum delay	2.5	0.78	0.72
Mean jitter	6.4	100.2	42.3
Absolute error		93.8	35.9
Jitter St. Dev.	15.6	136.5	73.5
Maximum jitter	727.9	622.5	912.3
Minimum jitter	0	0.04	0.004

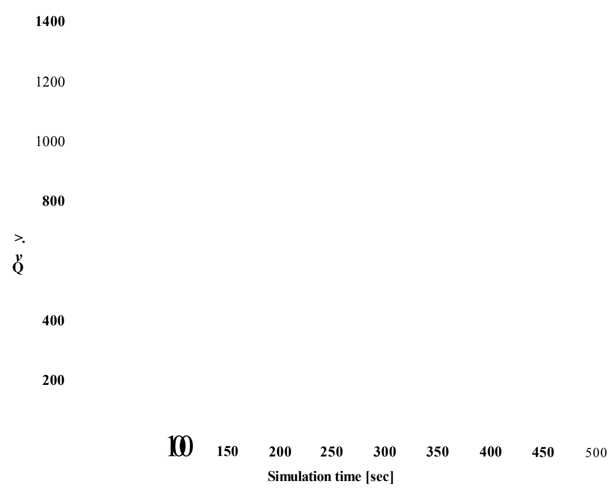
From Figures 6.2(a)-(c), it can be seen that the probe result of the one-way delay samples the user delay with an acceptable accuracy over the three network situations. As can be seen from the Figure, delay values increase when a high background traffic load is offered. That is because both probe and user traffic packets experienced the same network conditions and increasing the probe rate will produce high number of samples which will provide higher precision. So, increasing the probe rate has resulted in reducing the absolute error as can be observed in Table 6.2. These samples indicate that the measurements based on the monitoring packets can give fairly good estimates of the average delay and its variation.



(a)



(b)

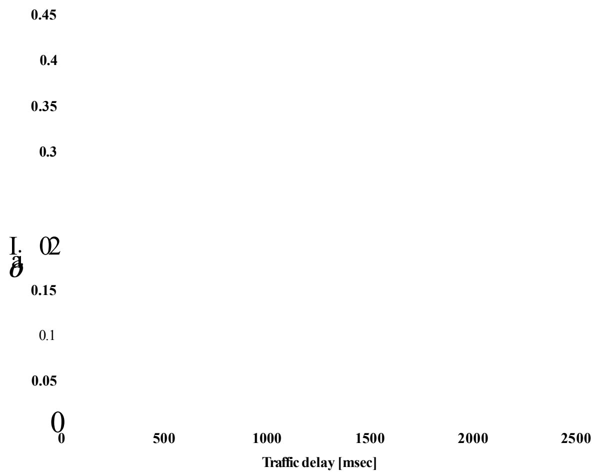


(c)

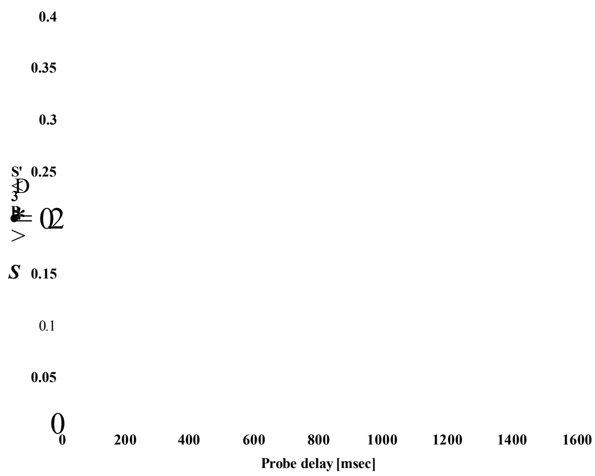
Figure 6.2: One-way delay of the: (a) actual traffic, (b) Monitoring traffic of M = 25 packets and (c) Monitoring traffic of M = 375 packets.

Figures 6.3(a)-(c) show the distributions of the one-way delays for the actual traffic and for the probe traffics of M = 25 and M = 375 block sizes during the measurement period. From these histograms, it is clear that the one-way delay distribution of the M —

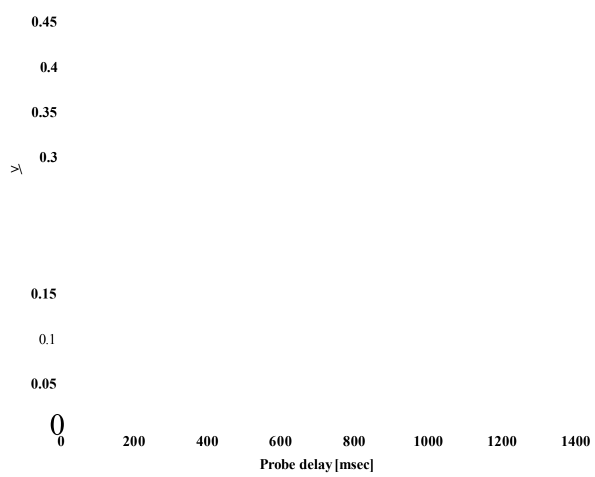
25 is quite similar to the actual user delay distribution. This means that both delays have similar measurement results which is more accurate than the M = 375 results. Nevertheless, for both monitoring blocks, about 40% of the measured delays were less than 40msec which is also identical to the actual delay.



(a)



(b)



(c)

Figure 6.3: One-way delay distribution of the: (a) actual traffic, (b) Monitoring traffic of M - 25 packets and (c) Monitoring traffic of Af= 375 packets.

Simulator!fme [sec]

(a)

800

700 -

Simulation tme [sec]

Simulation time [sec]

(c)

Figure 6.4: One-way delay variation (jitter) of the: (a) actual traffic, (b) Monitoring traffic of M = 25 packets and (c) Monitoring traffic of $M = 375$ packets.

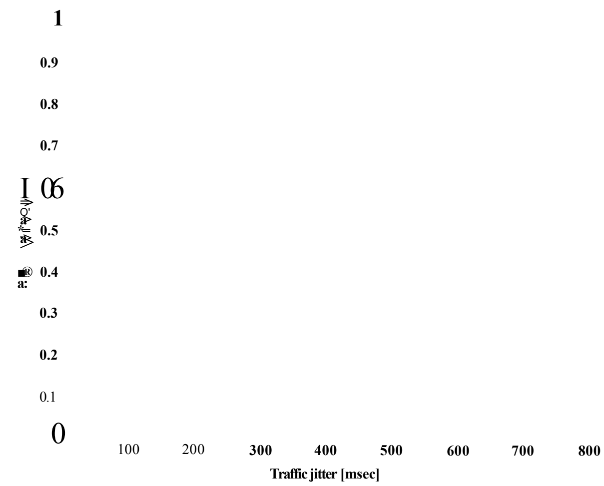
Also noticeable from Figures 6.4(a)-(c) is that the probe jitter during the lightly loaded network situation gives a reasonable representation of the user traffic jitter. On the other hand, as the network load is increased, the probe result overestimates the user delay variation. So, the probe jitter is higher than the traffic jitter over a congested or partially congested network. That is because the more loaded the network, the higher the contention between the nodes. A partially or fully loaded network will increase the probe delay in a significant amount compared to the delay a probe experiences when it encounters an empty network situation.

The variation of the jitter is due to the contention between the sending nodes for the available resources of the network. This contention will enforce the nodes to defer their transmissions for some time like Short Inter Frame Space and DCF Inter-frame Space (SIFS and DIFS) therefore, these packets will be queuing during the busy times of the network channel because it was occupied by some other nodes. The deferral of transmitting some packets will cause some variations in the delays of the consecutive probe packets. A probe packet that goes through a less busy condition may be followed by a high contention period which is met by the next probe which will experience more delay. The extreme difference in delay experienced by these probes will result in a higher jitter. The user traffic does not have this problem as the probe traffic because the next packet is more then likely to be in the same burst. Therefore the difference in delay between subsequent user packets is minimal, resulting in a lower jitter for the user traffic.

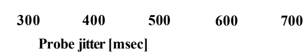
From Table 6.2, the estimated jitter measurement values of the probe traffics are higher than the actual user values. However, increasing the probe rate reduced the difference between the two measurements. This is because increasing the probe rate increases the samples number that is in the same network condition which will provide more reasonable results for the probe traffic.

Figures 6.5(a)-(c) show histogram distributions of the delay variation for the actual user traffic and for the monitoring packets using M = 25 and 375 block sizes. These diagrams reveal that there are some discrepancies between the actual and the estimated delay variation measurements. It is apparent that the actual user traffic jitter is lower than the probe jitter. These discrepancies decrease as the monitoring block decreases. For the actual user, more than 90% of the jitter values were less than 20 msec. Whereas, 58% and 46% of monitoring packets of M = 25 and 375 blocks had jitter less than

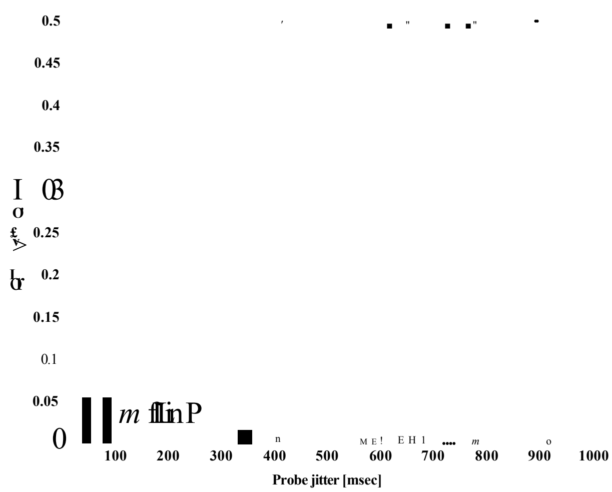
20msec, respectively. The reasons behind these discrepancies have been discussed earlier.



(a)



(b)

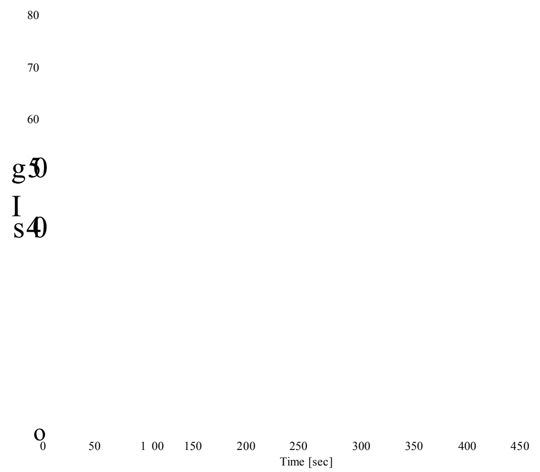


(c)

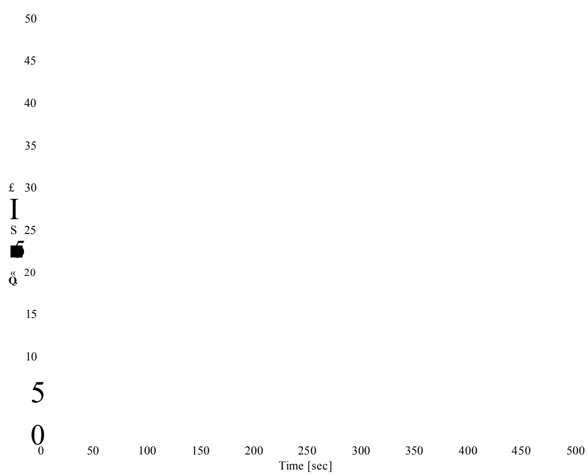
Figure 6.5: One-way delay variation distribution of the: (a) actual traffic, (b) Monitoring traffic of M - 25 packets and (c) Monitoring traffic of M = 375 packets.

6.3.1.2 Packet Loss

Unlike one-way delay or delay variation, packet loss estimation does not rely on sampling techniques (monitoring packets) directly. Packet loss is estimated based on providing a loss ratio for each monitoring block since the number of sent and received packets are counted and sent in the monitoring packets. One advantage of using monitoring packets is that the loss process calculation can be expressed with a higher resolution rather than the long-term average for the total measurement period. The resolution of these results depends on the ratio of the monitoring packets and the user traffic packets *(M)*. In addition, this feature can be used to define periods that contain lost packets (loss periods) and those without losses (loss-free periods) and their lengths.



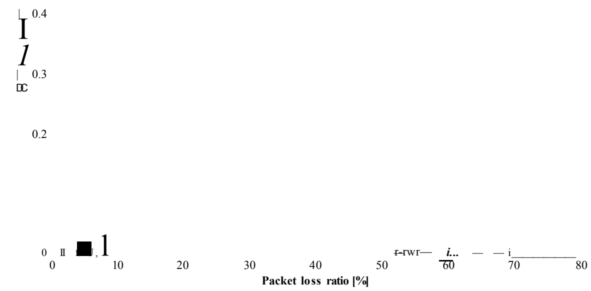
(a)



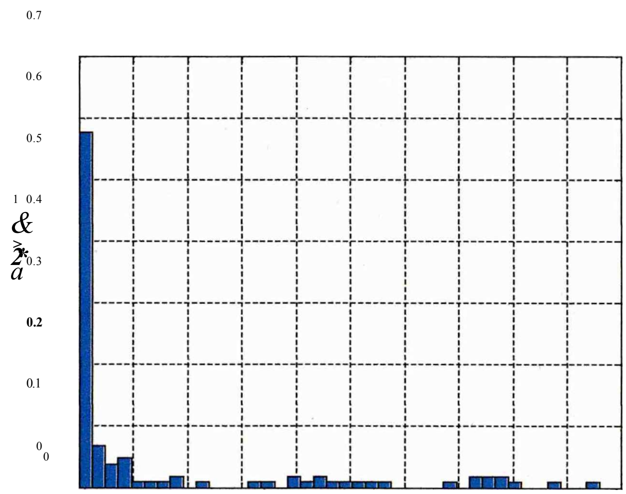
(b)

Figure 6.6: The packet loss ratio using: (a) M - 25 and (b) $M = 375$.

The estimated packet loss ratios using monitoring blocks of M = 25 and 375 are shown in Figures 6.6(a) and (b). These figures exhibit the degree of accuracy of the achieved resolutions in losses estimation over the measurement period. The smaller the monitoring block size the higher the loss resolution. The distributions of the loss ratio in the monitoring blocks are shown in Figures 6.7(a) and (b). The Figures confirm that the required resolution depends on the monitoring block size.



(a)



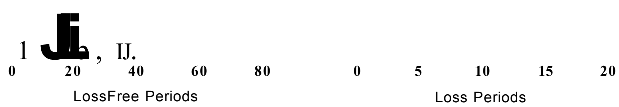
(b)

Figure 6.7: The packet loss ratio distributions using: (a) $M = 25$ and (b) 375.

Table 6.3 summarises the mean, minimum and maximum lengths of the loss and loss-free periods expressed in time units. The loss rate may not be sufficient enough to signify bursty losses. This type of representation provides information about the length of consecutive packet loss period distribution and about the bursty nature of the packet losses. This length is determined by the difference between the arrival timestamps of the monitoring packets. Loss-free period is computed in terms of the number of successive monitoring blocks that do not contain lost packets. This period is the time difference between the first monitoring packet and the last monitoring packet. The same principle is applied to calculate the loss periods. From Table 6.3, it is noticeable that as the block size increases, the mean, maximum and minimum of the loss and loss-free periods increases. Generally, it can be noticed that during the measurement period and over all the network situations, the network was lossless because the loss periods were very short compared to the loss-free periods for the whole monitoring blocks.

Table 6.3: Loss and loss-free period's length measurements based on two different monitoring blocks.

Monitoring Block	Loss-free period length [msec]			Loss period lengt l [msec]		
	Mean	Minimum	Maximum	Mean	Minimum	Maximum
$M=375$	94.3	7.8	283.4	10.7	5	16.3
$M=186$	46.8	4.3	259.9	3.8	2.4	6
$M=93$	30.6	1.2	251.7	2	0.8	4
$M=47$	12.3	0.5	200.8	1.3	0.4	2.9
$M=25$	6.3	0.2	182.4	0.72	0.13	3



(a)

(b)

Figure 6.8: the length of loss and loss-free periods versus time during the measurement period: (a) M = 25 and (b) M = 375.

Another powerful representation can be obtained using the loss periods. This is illustrated in Figures 6.8(a) and (b), which characterise the length of the loss and loss-free periods (in seconds) for $M = 25$ and 375. It is clear that the $M = 25$ monitoring block provides more details of the loss and loss-free periods variations than the $M = 375$ monitoring block. In addition, this representation is capable of showing how many loss and loss-free periods have taken place during the measurement period. Monitoring block ($M = 25$) shows that there were 76 loss-free periods and 18 loss periods over the monitoring period. While the monitoring block ($M = 375$) shows that there were 6 loss-free periods and 2 loss periods over the same measurement period. The ratio between the loss-free time and the total measurement period is 83.3% for the $M = 25$ and 80.8% for the $M = 375$ monitoring block. Whilst the ratio between the loss time and the total measurement period is 2.8% for the $M = 25$ and 7.3% for the $M = 375$ monitoring block.

6.3.1.3 Throughput

Using the monitoring block concept and in addition to the long-term average of the utilised capacity (throughput) for an application, it is often useful to obtain the maximum and the minimum values as well as the variation during the measurement period. This can be calculated since the packet size, the number of the sent and received packets along with the timestamps are available for each monitoring block. This throughput is calculated between two monitoring packets in Kbps using the following equation:

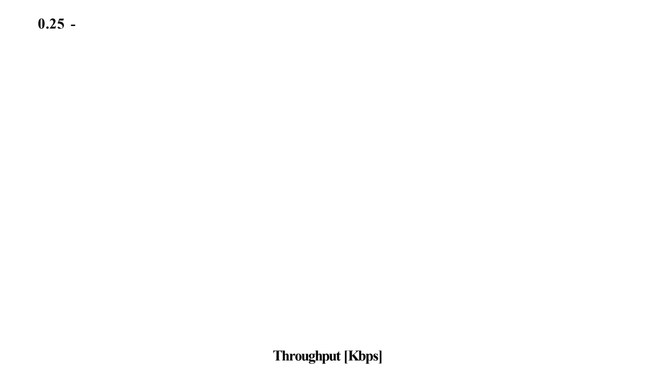
$$Throughput = \frac{8 * PS * N}{1024 * (Timestamp(i) - Timestamp(i-1))} \tag{6.1}$$

Where PS is the actual traffic packet size in byte, N is the number of packets between two monitoring packets and i is the current monitoring packet.

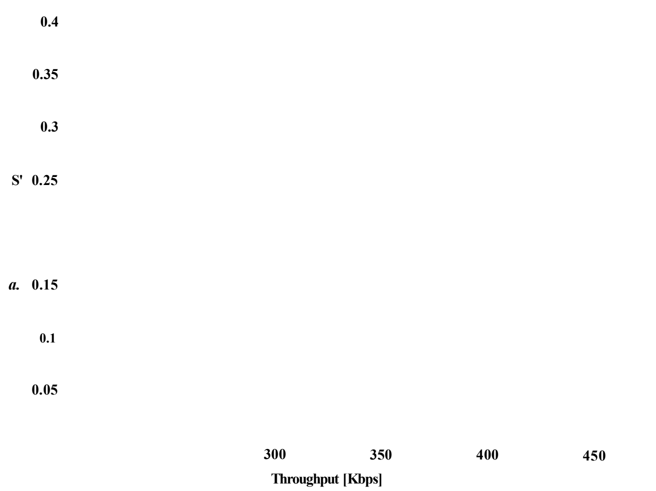
In Table 6.4, the average, maximum, minimum and standard deviation of the throughput per monitoring block are presented for several monitoring block sizes. The estimated average throughput is in the range of 345-355 Kbps for all values of monitoring block sizes. In this case, the estimated maximum throughput increases when the monitoring block size decreases. On the other hand, the minimum throughput decreases as the monitoring block size decreases. Moreover, the standard deviation increases when the block size decreases. The reason for this is that reducing the block size increases the number of samples. This in turn increases the throughput within the different network load situations over the monitoring period. The estimated throughput values vary between large and small values resulting in an increase in the standard deviation.

Table 6.4. The actual throughput estimations based on different monitoring blocks.

Units: [Kbps]	M= **375**	M = **186**	[illegible]	[illegible]	[illegible]
Average throughput	355.6	355.5	355.6	354	345.9
Maximum throughput	423	460.6	518.1	656	705.5
Minimum throughput	207.4	177.4	140	124	62.3
Throughput St. Dev	53.32	54.5	56.3	61.6	70.7



(a)



(b)

Figure 6.9: The throughput distributions based on monitoring block of: (a) M= 25 and (b) M = 375.

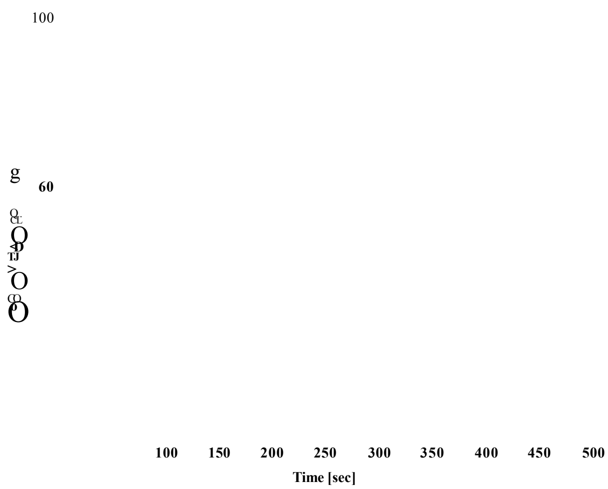
Figures 6.9(a) and (b) depict the distribution of the throughput, per monitoring block for M - 25 and 375. The distributions provide an accurate estimate of the actual throughput (384Kbps) as most of the estimated throughput values are distributed around this value. It is clear that the resolution produced by the M = 25 block size is more than that of M =

375. So, the desired estimated throughput resolution will be dependent on the required accuracy.

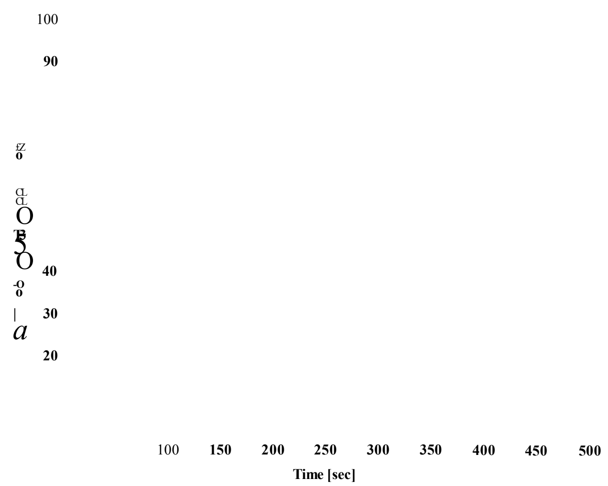
6.3.1.4 Overall QoS

The most important QoS parameters that affect the videoconferencing performance are the delay, delay variation and the packet loss. These parameters can be estimated (as described earlier) by probing the network. Delay and delay variation can be taken (estimated) directly from the probe traffic and packet loss is estimated using the monitoring block concept. After measuring these parameters, they were fed to the fuzzy system to produce the estimated overall QoS of the videoconferencing application based on the results obtained from the monitoring packets. In addition and in order to check the accuracy of the estimated overall QoS result, these parameters were measured for the actual user with the probe traffic switched off. The actual traffic parameters were averaged using the blocking technique for $M = 25$ and $M = 374$ packets. Fuzzy system outputs of the estimated QoS using the probe and the actual user overall QoS are shown in Figures 6.10(a)-(c).

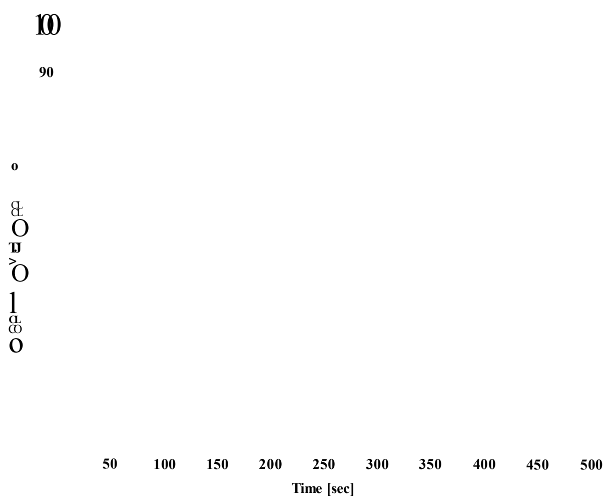
It can be observed from Figures 6.10(a)-(c) that the QoS of the monitoring probe packets can infer the actual user overall QoS during the periods of light and heavy loaded network situations. On the other hand, during the partially loaded state, the probe QoS could not estimate the actual user QoS especially when using the $M = 375$ monitoring block size. However, the probe gave a better estimation of the actual overall QoS using the $M = 25$. This means that the QoS estimation was affected by the probe rate (i.e. number of samples). In addition to that, the poor QoS estimation was, mainly, due to jitter. As the network load is increased, the jitter will increase and in particular the probe jitter as explained earlier. The probe jitter will be higher than the actual traffic jitter. Occasionally the probe jitter will exceed the limits of the required QoS while the actual traffic jitter may stay within these limits. Due to this, the probe QoS will underestimate the actual traffic QoS and especially during the partially loaded situations because during the heavy loaded state periods both the probe and the traffic parameters will go beyond the required values and so the overall measured QoS will be poor.



(a)



(b)



(c)

Figure 6.10: Measured overall QoS of the: (a) actual traffic, (b) Monitoring traffic of $M = 25$ packets and (c) Monitoring traffic of $M = 375$ packets.

Table 6.5 illustrates the long-term statistics (mean, standard deviation, maximum and minimum) that characterise the overall QoS values for the actual user traffic and the estimated values using different monitoring block sizes. This table reveals that as the monitoring block size increases the estimated QoS is enhanced compared with the actual QoS value. Increasing the block size will provide more samples to be evaluated using the fuzzy system which will monitor the network more accurately. The estimated overall QoS standard deviation, maximum and minimum are mostly the same as the actual values. This means that the long-term average QoS estimation using monitoring packets is a good approximation of the actual QoS.

Table 6.5: The actual and the estimated values for overall QoS using different block sizes *M.*

units: [%]	**Actual values**	$M = 375$	[illegible]	$M = 93$	[illegible]	$M = 25$
Evaluated QoS	52.74	40.94	40.51	42.14	43.24	44.51
Absolute error		11.8	12.25	10.6	9.5	8.25
QoS Std. Dev	37.09	38.96	38.49	38.79	38.84	38.66
Maximum QoS	90.48	90.52	90.52	90.52	90.52	90.52
Minimum QoS	9.30	9.27	9.27	9.27	9.27	9.27

To compare the levels when the overall QoS was poor, average and good, for both the actual and the probe traffics *(M* = 25 and 375), a bar charts distribution was used. The length of the bar was representative of the percentage of each QoS case. Figures 6.11(a)-(c) show the bar charts of both application's overall QoS. Monitoring traffic using *M* - 25 was closer to the actual overall QoS regions. That was due to the fact that the network was subjected to more assessments over the measurement period using this rate which will result in a higher precision in the QoS estimation than the *M* = 375 probe rate.

Average

Average

Good

Average

QoS level

(c)

Figure 6.11: The overall QoS bar chart for: (a) Monitoring packets using M = 375 packets, (b) Monitoring packets using M = 25 packets, (c) actual traffic.

In order to quantify how much the overall QoS of each application was; poor, average or good, the variation of these values, mean and standard deviation were calculated. Table 6.6 illustrates these statistics that characterise each region of each the traffic overall QoS values for the actual user traffic and the estimated values of M = 25 and 375 monitoring block sizes. Table 6.6 exhibits that the probe rate of M = 25 had better QoS approximation of the actual overall QoS because all of its estimated statistics are closer to the actual values.

Table 6.6: Statistics of actual and estimated overall QoS region for M = 25 and 375.

Units: [%]	Actual values			M = 375			M = 25		
QoS	Poor	Average	Good	Poor	Average	Good	Poor	Average	Good
Mean	11.8	51.9	88.1	9.8	38.8	89.7	10.5	46.2	89.5
Std. Dev.	5.4	9.9	5.2	2.6	0	2.2	4	7.8	2.5

So as to obtain a more specific picture about the actual and the estimated overall QoS for each application without classification of the QoS values into good, average and poor regions, probability distribution functions have been generated of each QoS. These distributions are shown in Figures 6.12(a) and (b). The Figures illustrate the cumulative distributions, Pr{Jf< a}, where the random variable X denotes the end-to-end QoS. The usefulness of this method stems from the fact that it gives the percentage that the QoS is less than any threshold value (a). Using these types of distributions, for example, it is very easy to assess the probability of the QoS. In addition to that, it can be observed that the minimum and maximum values of the QoS can be found from these figures. It is apparent that the monitoring packets could, to some extent, estimate the actual QoS cumulative distribution. For example, it can be seen from the figures that it is very easy to assess the probability that the QoS was less than 40%. It is from the actual traffic 0.47, 0.57 and 0.55 using the monitoring traffic of M = 25 and 375 respectively. In addition to that, it can be observed that the minimum and maximum values of the QoS can be found from these figures. The minimum value for both traffics (actual and monitoring) was 9.3%. The maximum value for the actual traffic was 90.5% and 90.5% for M —25 and 375 probe traffic.

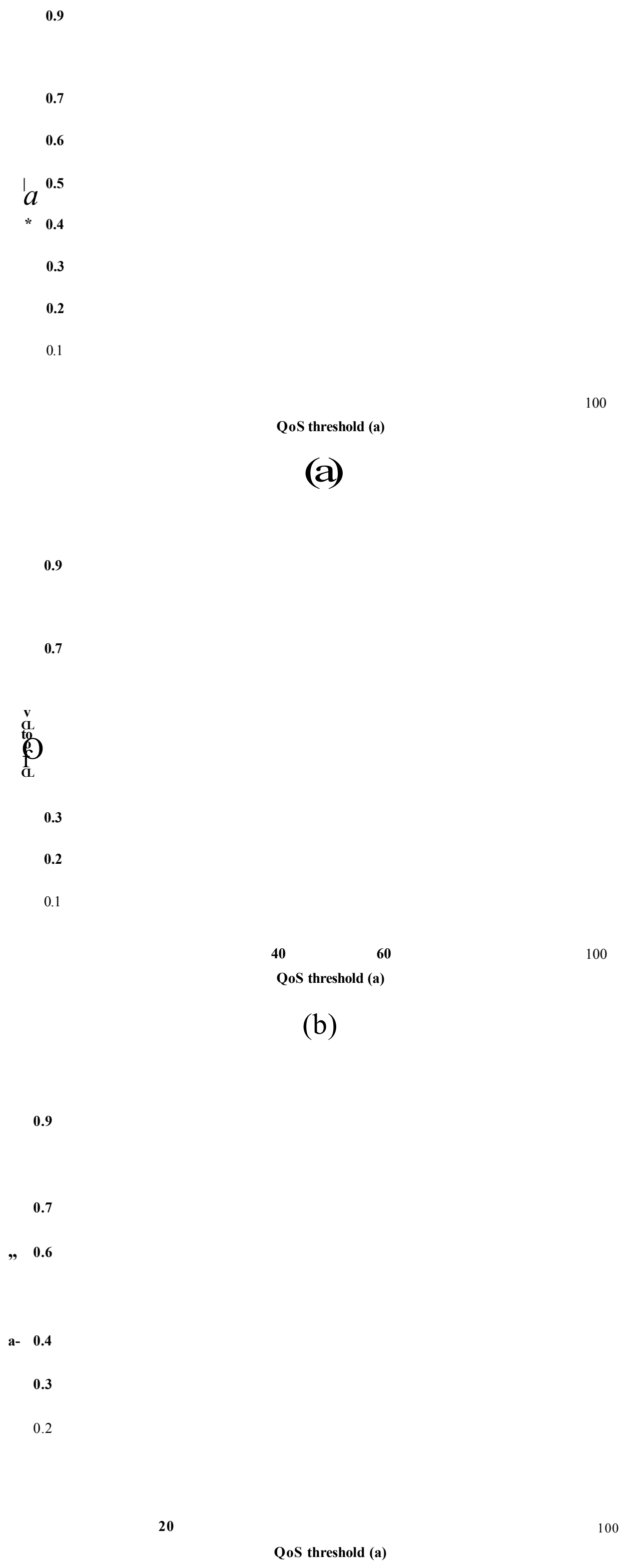
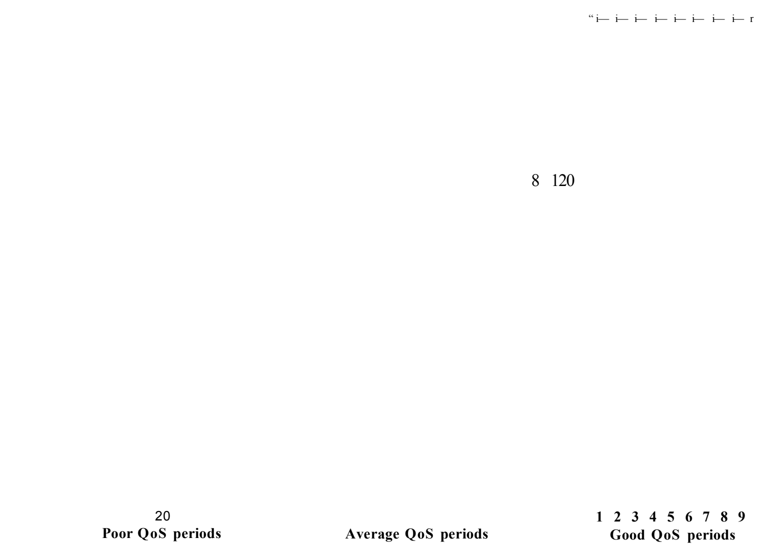
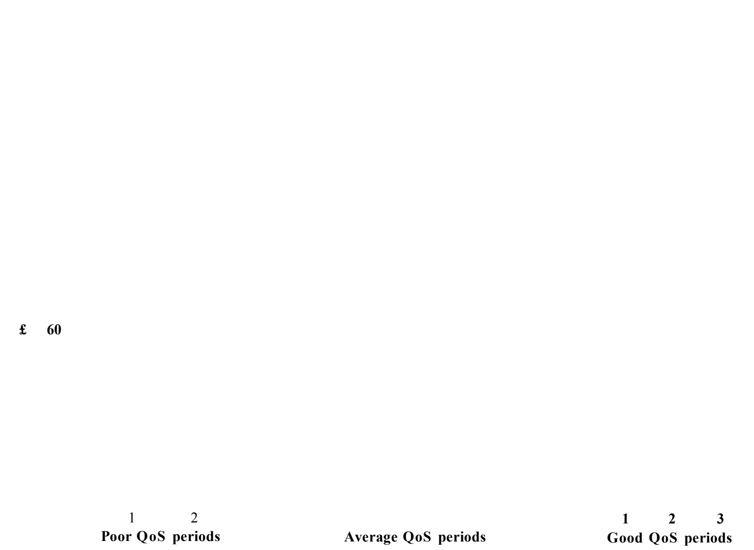


Figure 6.12: The overall QoS distribution for: (a) actual traffic, (b) Monitoring packets using M - 25 packets, (c) Monitoring packets using M= 375 packets.

An additional valuable metric can be achieved using the concept of monitoring blocks. This concept makes it possible to define time intervals in which the QoS was good, average and poor. These intervals are defined as the number of consecutive monitoring blocks which have the same QoS level. The period length is determined by the difference between the timestamps of the monitoring packets. For example, good QoS period is computed in terms of the number of successive monitoring packets that have QoS values larger than 67%. The length of this period is the time difference between the first monitoring packet and the last monitoring packet. The same principle is applied for the determination of poor and average QoS periods. This is illustrated in Figures 6.13(a) and (b) which characterise the length of the periods for Good, Average and Poor QoS (in seconds) for M —25 and 375.



(a)



(b)

Figure 6.13: The length of Poor, Average and Good QoS periods versus time during the measurement period using monitoring block of: (a) M = 25 and (b) 375.

It is clear from the Figure that the M = 25 monitoring block provides more details of variations of these periods than the M = 375 monitoring block. In addition, this representation demonstrates how many Good, Average and Poor QoS periods have taken place during the measurement period. Monitoring block (M = 25) shows that there were 31 Poor QoS periods, 17 Average QoS periods and 9 Good QoS periods over the monitoring period. While the monitoring block (M = 375) exhibits that there were 2 Poor QoS periods, 1 Average QoS period and 3 Good QoS periods over the same measurement period. The ratios between the Poor, Average and Good intervals and the total measurement period are: 48% and 37% for Poor QoS, 2.6% and 1.5% for Average QoS and 35% and 26% for Good QoS using M = 25 and M = 375 monitoring blocks, respectively.

6.3.2 Biasness

The biasness is measured by comparing the actual traffic parameters and the QoS results when the probe traffic is switched on and when it is switched off. This will be done using the long-term results in both cases.

6.3.2.1 One-way Delay and Delay Variation

Table 6.7 illustrates the delay values of both the actual traffic and the monitoring blocks for M = 25 and 375. The results reveal that the higher probe rate affects the actual traffic more than the lower value as can be seen from the calculations of the error in Table 6.7. That was because increasing the probe rate will increase the packets interfering with the actual traffic. This will increase the packets contending to the same resources and so increase the packets in the queue which will result in an increase in the waiting time. Increasing the waiting time and the processing time of every probe packet will result in increasing the delay. The standard deviation of delay for the actual traffic with probe rate (M = 375) is the same as the standard deviation without probe traffic which means that the probe traffic has no influence in the actual traffic.

From Tables 6.2 and 6.7, it is apparent that the results are contradictory. That is because probe rate with M = 25 gave a more accurate delay result, while it has larger effect on the actual traffic. On the other hand, probe traffic with M = 375 provided a smaller effect on the actual traffic but with lower accuracy. This means that the higher the probe rate is, the more precise the result and the more the network is perturbed and vice versa.

Table 6.7: The effectiveness of the probe traffic presence in the actual delay measurement results.

Units: [msec]	Without probe	With probe	
	Actual traffic	Actual traffic with $M = 375$	Actual traffic with $M = 25$
Mean delay	335.7	338.8	355.2
Absolute Error		3.1	19.5
St. Dev.	331.6	331.8	325.7

As can be observed from Table 6.8, jitter values have the same tendency as delay values. The absolute error between the actual traffic (without probe) and the actual traffic with M = 375 is lower than that of M = 25. The reason for that is the same as that discussed for the delay. By comparing the error results in Table 6.2 with those in Table 6.8, they exhibit the same performance of the delay errors. The higher probe rates the closer the results and the more effect on the actual traffic.

Table 6.8: The effectiveness of the probe traffic presence in the actual jitter measurement results.

Units: [msec]	Without probe	With probe	
	Actual traffic	Actual traffic with $M = 375$	Actual traffic with $M = 25$
Mean jitter	6.3	6.4	6.5
Absolute Error		0.05	0.17
Standard deviation	16.1	15.6	15.6

6.3.2.2 Packet Loss Ratio

Probe rate also has an effect on the actual traffic packet loss ratio. For example, the packet loss ratio of the actual traffic (without probe) was 12%. This value was 12% and 14% for the actual traffic with monitoring blocks of $M = 375$ and $M = 25$, respectively. It can be noticed that the M = 375 monitoring block has no effect on the loss ratio of the actual traffic while the M = 25 monitoring block increases the loss ratio by 2%.

6.3.2.3 Overall QoS

To check the effect of adding probe traffic on the actual traffic overall QoS, the difference between the evaluated actual overall QoS values in both cases with and without probe traffic were calculated. Table 6.9 shows the actual overall QoS statistics. It is obvious that increasing the probe rate will increase the distance (error) between the evaluated QoS values (with and without probe traffic). In spite of these differences the QoS is still in the same region (i.e. Average QoS). The standard deviations, maximum and minimum QoS are in the same range for the actual traffic in both cases (with and without probe traffic).

Table 6.9: The effectiveness of the probe traffic presence in the actual overall QoS measurement results.

	Without probe	**With probe**			
Units: [%]	**Actual traffic**	**Actual traffic with *M = 375***	**Actual traffic with *M* = 186**	**Actual traffic with *M* = 93**	**Actual traffic with *M* = 25**
Evaluated QoS	52.7	52.1	51.1	50.2	48.6
Absolute Error		0.7	1.7	2.6	4.2
QoS St. Dev.	37.1	36.8	37.3	37.3	37.8
Maximum QoS	90.5	90.5	90.5	90.5	90.5
Minimum QoS	9.3	9.4	9.4	9.3	9.3

Comparing the results in Table 6.5 and 6.9 confirms that there were no large differences between the probe traffic QoS and actual traffic overall QoS (in both cases).

6.4 Summary

This chapter focused on developing a new approach for estimating the overall application QoS based on the QoS parameters obtained from the probe traffic packets. The simulation results showed that this approach provided a wide range of metrics that can be used to monitor the actual traffic measurements using different probe rates. Furthermore, these measurements were also tested and examined in terms of its accuracy and biasness to be representative of the actual traffic results. The proposed approach provided good accuracy in estimation of the overall QoS and the QoS parameters but showed drawbacks in jitter estimations. In the next chapter, a new estimation method is proposed to overcome some of the shortcomings of probe-based approach based on standard sampling schemes.

Passive QoS Evaluation System based on Sampling Technique

7.1 Introduction

As networks grow in complexity and scale, the importance of network performance monitoring and measurement also increases significantly. The variability of the traffic demands and dynamic network conditions became a challenge for service providers to ensure that the network resources are contested satisfactorily with the traffic/user demands. Passive methods are one of the main schemes which are used in traffic measurements. Passive measurements are based on achieving measurement of the actual traffic currently present in the network where routers or other hosts monitor existing traffic passing through or destined to them. This provides an indication of the treatment of the current traffic in a network section.

Passive methods have the advantage of not adding an extra load to the network, like the active methods, i.e., they are a non-intrusive method, and enable the gathering of a large amount of detailed information (Lindh, 2001). However, they require the transfer of the captured data for comparison with the other data and the identification of each packet by its header or content, which is hard when the traffic volume is large. Therefore, passive measurements have the disadvantage of requiring substantial resources for comparison and computation but they are well suited to investigate the quality of the existing flows without burdening the network with probe traffic as in active measurements.

In order to evaluate the disadvantages of both active and passive schemes, sampling methodologies can be employed. The aim of this chapter is develop a new assessment mechanism based on these methodologies. The developed system will: reduce the amount of data to be processed, reduce the demand on the overhead processing time of the collected data, and hence speed up the performance measurement procedures with reliable results.

In this chapter, Section 7.2 discusses the sampling methods in terms of reasons behind deploying sampling in traffic monitoring and measurements, sampling schemes and their characteristics and some of the related studies. Section 7.3 presents the measurement approach, which includes the description of sampling approaches, methods of analysis and the simulation set up employed. In Section 7.4, the experimental results are presented. Finally, Section 5.7 summarises the chapter.

7.2 Sampling for Passive Measurement

7.2.1 Why Traffic Sampling?

The necessity for detailed measurements stems from the need for identifying the exact situation of the network performance and to move to services beyond the ordinary best effort model. The demand of network measurements has increased due to many reasons (Zseby, 2005). First, the appearance of new multimedia metrics which are in addition to observing some key characteristics of network and data transmission, like link load or round-trip-times, SLAs, sophisticated accounting methods and increasing security threats. These require the measurement of much more and different metrics. Second, the trend towards wireless communication pushes the deployment of mobile devices and wireless networks which usually have very scarce resources (e.g. processing power, storage). Furthermore, transmission resources in wireless networks are often much more limited than resources in fixed networks. Therefore, the trend towards mobile communication increases the demand for resource efficient measurements. Third, increasing data rates which elevate the amount of result data and with this, the resource consumption for processing, storage and result data transmission.

Therefore, the required resources to obtain such detailed measurement information increases with the amount of measured traffics. For many multimedia applications, detailed measurements will result in an enormous amount of measurement data which needs to be transferred to collection points for processing, storage and analysis. These collection points may be routers, switches or the destinations themselves. These nodes may not be able to do that role because (Duffield, 2004):

(i) Processing and storage resources on these nodes are comparatively scarce because they are already employed in the regular work of routing and switching.

(ii) The transmission of measured data to the collection points can consume significant amounts of network bandwidth.

(iii) Sophisticated and costly computing systems are required for analysis and storage of the data.

The above three factors highlight the need for data reduction. The use of sampling techniques in the measurement process allows the resource consumption to be reduced. Two issues affect the decision in choosing a sampling method: reliability and cost. Reliability of the estimated versions increases as the sample size increases but, cost is the restrictive factor. Thus, an appropriate balance between the reliability of the estimate and the cost of obtaining it must be defined. Costs can be expressed in terms of resource consumptions. The deployment of sampling provides information about a specific characteristic of the parent population. With sampling, not all packets are measured by the monitoring node, but only a selected fraction of the packets (Drobisz and Christensen, 1998). Thus, sampling provides the ability to reduce the measurement cost in terms of resource consumption which will limit this cost to a small fraction of the costs of providing the network service itself.

7.2.2 Sampling Schemes

Sampling is applied whenever global characteristics of specific populations are required but the analysis of every element may be too expensive or very time consuming. Sampling methods can be characterised by the sampling algorithm used, the trigger type for starting a sampling interval and the length of the sampling interval (Zseby and Scheiner, 2002).

7.2.2.1 Sampling Algorithms

Sampling algorithm describes the basic procedure for the process of samples selection. There are three basic processes: systematic sampling, random sampling, and stratified sampling.

7.2.2.1.1 Systematic Sampling

Systematic sampling describes the procedure for selecting the starting point and the frequency of the sampling according to a pre-determined function. This includes for example the periodic selection of every n^{th} element of a trace based on counting the arrival packets or selection of the next arriving packet within a time t of the trigger. Figure 7.1 shows the schematic of the systematic sampling (Claffy, et al., 1993).

Figure 7.1: Schematic of systematic sampling (take the 1st member of each n-packet bucket).

7.2.2.1.2 Stratified Sampling

Stratified sampling splits the sampling process into multi-steps. First, the elements (packets) of the parent population are grouped into subsets (sub-populations) in accordance to a given characteristics. Then, the samples are randomly taken from each subset (usually called strata). Stratified sampling is similar to systematic sampling, except that rather than selecting the first packet from each bucket; a packet is selected randomly from each bucket. Figure 7.2 illustrates the schematic of the stratified sampling (Claffy, et al., 1993). For example, if the whole region of interest, A, is spilt into M disjoint sub-regions (i.e. buckets) as in the following equation (Bohdanowicz and Weber, 2002):

$$A=\bigcup_{k=1}^{M} A_k \quad with \;\; A_j \cap A_l = 0 \;\; for \;\; l \neq j \tag{7.1}$$

Where A_k is the k^{th} sub-region.

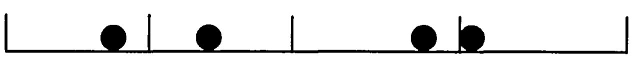


Figure 7.2: Schematic of stratified sampling (take a random member of each n-packet bucket).

7.2.2.1.3 Random Sampling

Random sampling selects the starting points of the sampling interval in accordance with a random process (Zseby and Scheiner, 2002). It is the process of selecting n random units from a population N. The selections of sampled elements are independent and each element has an equal probability of being selected. Figure 7.3 depicts the schematic of the random sampling (Claffy, et al., 1993).

Figure 7.3: Schematic of random sampling (take random members from the population).

7.2.2.1.4 Measurement Interval and Sampling Trigger

Sampling techniques can be differentiated by the event that triggers the sampling process (Zseby and Scheiner, 2002), (Claffy, et al., 1993), and (Paul and Lillian, 1989). The trigger determines what kind of event starts and stops the sampling intervals. With this, the sampling frequency and the length of the sampling interval (measured in packets or time) are determined. Measurement (sampling) interval defines the interval for which the metric of interest should be measured. There are two different ways to define the measurement interval (Zseby, 2005). Count-based definition by which the measurement interval is defined by the number of packets and Time-based definition where the measurement interval is defined as time interval.

The sample selection decision depends on two trigger types (Zseby, 2004); these are:

- Count-based: the packet selection decision is based on the packet count. An example of this is the systematic sampling of every n^{th} packet of a specific type.
- Time-based: Using this method, the samples are selected based on predetermined time interval. For example, the arrival time of a packet at the monitoring point determines whether this packet is captured or not.

If the member of the populations are randomly ordered, it is expected that all three sampling methods (systematic, stratified and random) will be equivalent (Claffy, et al., 1993). The sampling decision of the systematic sampling can either use count-based or time based triggers. Time-based sampling decision can be achieved by periodically enabling/disabling the packet capturing process. But, it is particularly poor for assessing the network performance metrics such as delay and delay variation (Ma, et al., 2004). This is because it tends to miss bursty periods which contain many packets with relatively small inter-arrival times if using a larger timer. Therefore, it is better to use the count-based trigger when deploying systematic sampling using a packet counter; however it is vulnerable to bias due to synchronisation if the metric being measured exhibits periodic behaviour. These potential problems (i.e. synchronisation and periodicity) may be avoided by suitable use of random additive sampling because the intervals between successive triggers are independent random variables with a common distribution (Duffield, 2004) and (Manku and Motwani, 2002). On the other hand, generally, random sampling is less efficient than systematic sampling for populations with linear trend or those with a population variance less than the variance of the samples (Claffy, et al., 1993). However, for these populations (i.e. with linear trend), stratified sampling is more efficient than systematic sampling because, for example, if the sample from the first bucket was too low, the sample from each subsequent bucket would also be too low, therefore, stratified sampling would alleviate this difficulty

(Krishnaiah and Rao, 1988). In addition to that, stratified sampling may produce a gain in precision if the strata can be considered homogenous (Krishnaiah and Rao, 1988).

7.2.3 Related Work

Network performance monitoring and measurement have been the interest to many research groups such as (CAIDA, 2006) and (NLANR, 2006). A comparison of passive and active measurements can be found in (Graham, et al., 1998) and (Zseby, et al., 2001). Passive measurements are widely used for packet counting, capturing over a network section or path (NeTraMet, 1997), (NetFlow, 1999) and (Brownlee, 2000). Sampling methods have been applied to network performance measurements for different purposes (Claffy, et al., 1993), (Duffield and Grossglauser, 2000), (Cozzani and Giordano, 1998) and (Zseby, 2002). The investigations of volume and packet counts using sampling have been presented in (Duffield, et al., 2001) and (Jedwab, et al., 1992). Claffy, et al. (1993) illustrate the deployment of sampling for the estimation of the packet size and inter-arrival time distributions. Sampling approaches for non-intrusive quality measurements are described by Zseby (2003). In addition to that selected sampling methods are standardised in the IETF Packet Sampling Working Group (PSAMP) (PSAMP, 2006).

A good review and explanations for the classical sampling methodologies in the context of the Internet and the presentation of applications and sampling methods for Internet passive measurement can be found in (Duffield, 2004). Zseby (2005) introduced the key challenges for the deployment of sampling techniques for network measurements in terms of estimation of the proportion of packets that violate the SLA contract for the one-way delay metric.

7.3 Measurement Approach

In this work, an evaluation of the network performance by measuring the QoS parameters is carried out. Non-intrusiveness is the main characteristic of the evaluation system. A performance measurement method for estimating the QoS experienced by the users has been proposed based on a sampling technique. The basic procedure is as follows:

(i) Take a suitable number of samples of the ongoing current traffic.

(ii) Measure the QoS parameters (delay, jitter, packet loss and throughput) based on the sampled packets.

(iii) Assess and quantify the application QoS using two evaluation systems (Fuzzy and Distance approaches (Chapter 5) using the selected samples.

7.3.1 Sampling Techniques with One and Two Measurement Points

The following section describes the use of sampling techniques for measurements with one and two monitoring points. These points have only passive capability, because they do not modify or affect the ingoing traffic through the network. The main elements for implementing this for QoS measurement are illustrated in Figures 7.4 and 7.5.

Processing & Analysis

QoS Parameter Calculation

Timestamping

Sampling

Classification

Packet Capturing

Flow of interest

Sending edges

Network

Monitoring point

Figure 7.4: Sampling measurement system with one-monitoring point.

In these figures, the classification means that if the monitoring point is not an end point (i.e. receiving end) and there are several traffic flows through it, just the selected packets are used in the measurement process (i.e. the packets of flow of interest). For the one monitoring point system, a number of samples are selected and then timestamped. The timestamps and the packet ID are carried within the packet from the sending edge. The process of calculations and analysis are then started.

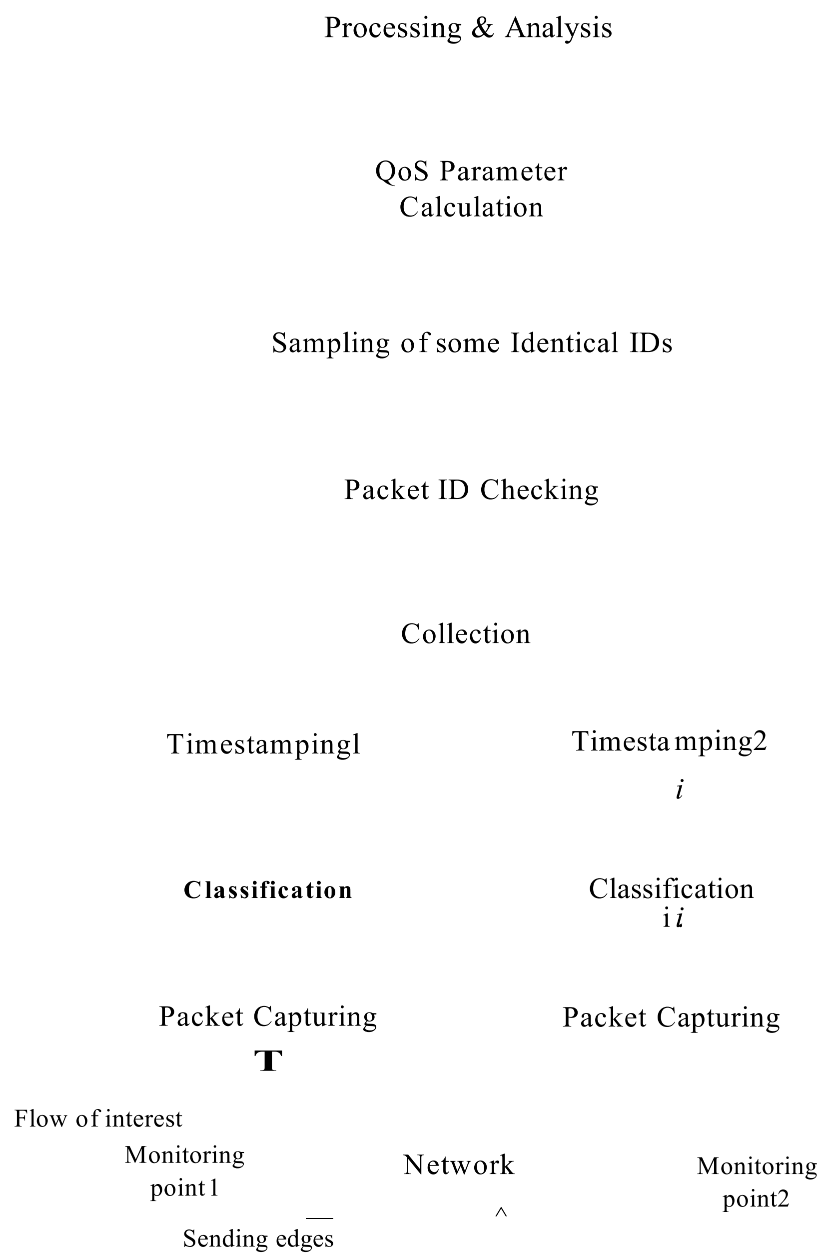


Figure 7.5: Sampling measurement system with two-monitoring points.

For the two-monitoring points system and after classification, every monitoring point timestamps its captured packets and all the filtered packets will be stored in a collection point. Some metrics require the correlation and synchronisation of data from different measurement (monitoring) points like delay and jitter. This work is based on a simulation study, so the monitoring nodes are already synchronised. But, in reality, a synchronisation tool should be used to ensure and maintain the two monitoring nodes are time synchronised. Current solutions are to synchronise nodes to a specific reference time like the UTC using the GPS receivers or GTB (Jiang, et al., 2000). To achieve the correlation of data, a method for recognising the packets at the second monitoring point must be deployed. This can be done by applying packet ID recognition (Mark, et al., 2003). For each packet generated, a timestamp and packet ID are issued at both monitoring points. An ID checking block is used to match the packets, which have the same ID. After this, the process of sampling is performed using one of the sampling methodologies (systematic, random, or stratified sampling). Then, for the sampled packets, the required QoS parameter is calculated.

Due to the problem of correlation between sampled packet and clock synchronisation of the two monitoring points, in practical realisations, it is difficult to perform count-based or time based sampling (Zseby, 2002). Monitoring and sampling at one measurement point can be an alternative solution to deal with the two monitoring points drawbacks. In this case, count-based and time-based sampling triggering methods can be deployed.

Using one monitoring point sampling system, systematic, random and stratified sampling techniques are implemented as follows:

(i) Systematic sampling: in count-based, every n^{th} packet is selected and in time-based every n^{th} time (period), packet is selected. The easiest way to implement that is to set a counter to the n value and decrement it on each packet arrival, then select the packet when the counter reaches zero. After that, the counter is reset and the process is repeated. In order to get different sets of samples of the same size for several experiment runs, the starting point (the 1st selected packet) for the selection is chosen randomly in the experiments.

(ii) Random sampling: this type of sampling can be implemented for count-based sampling and is not reasonable for time-based sampling because the number of packets within a time interval is not known in advance (Zseby, 2002). To implement the count-based of n samples from N population, n random numbers need to be generated in the range [1, N] and then select packets according to their position in the flow. For every run, new random numbers should be generated for the same selected sample size.

(iii) Stratified sampling: Stratified sampling for count-based sampling can be realised using the same way of random sampling implementation. If the number of packets in every bucket is N, then for every bucket, n random numbers in the range [1, N] are generated (where n is the sample size from each bucket and N is the size of the strata). For every run, new n random numbers should be generated for the same selected sample size.

7.3.2 QoS Parameters Calculation and QoS Estimation

(i) Delay calculation:

To calculate the end-to-end one-way delay of a packet, the difference between the values of the timestamps of the packets that arrived at the measurement points, which are selected in accordance to the sampling process is calculated. The correlation between the two timestamps of the same packet is achieved by the packet ID checking block.

(ii) Jitter calculation:

To calculate the end-to-end jitter for a specific flow, the sampling technique must take two consecutive packets in a sample. Then the difference between their delays is calculated after correlating the two timestamps of each packet via the packet ID.

(iii) Throughput and packet loss calculations:

Throughput and packet losses are calculated by making a difference of the counts of the number of packets passing the two measurement points based on the packet sequence number to produce the number of the received packets between every two successive samples. In order to obtain the number of sent packets between the two successive samples, the difference between the sequence numbers of these samples is computed. If the number of sent and received packets between these samples is known, then a packet loss ratio is calculated. Throughput is calculated by multiplying the number of received packets by the packet size and then this is divided by the receiving time difference of the two successive samples. This throughput between two samples is calculated in Kbps using the following equation:

$$Throughput = \frac{8 * PS * N}{1024 * (Timestamp(i) - Timestamp(i-1))} \qquad (7.2)$$

where PS is the packet size, N is the number of packets between two samples and i is the current sample.

After calculating the delay and jitter of the sampled packets and computing the packet loss ratio between every two successive packet samples, the QoS was assessed based on these samples to estimate the actual population QoS. This QoS is obtained using the two evaluation systems: Fuzzy assessment system and Distance assessment system using the procedures discussed in Section 4.4 and sub-sections 5.4.1 and 5.4.4.

Based on the proposed approach, the following outcome measures may be obtained:

- Samples of the packet delay and jitter between the sending and receiving nodes.
- If the packet size is known, it is possible to estimate the throughput of the application over each measurement interval in addition to the long-term average.
- The packet loss ratio of the application between the sending and receiving nodes of each measurement interval.
- The length of the loss free periods and loss periods expressed in terms of the number of consecutive measurement intervals that are loss free (do not contain lost packets) and the number of measurement intervals that contain lost packets.
- Samples of the estimated QoS values of the application based on the QoS parameters resulting from the sampling measurements of each measurement interval.

After generating the estimated QoS populations using the three different sampling methods, a comparison process was carried between these populations and the parent QoS population. The aim is to determine which method represents the parent population most accurately. The mean, the standard deviation, the degree of significance, the Standard Error (SE) of difference and the 95% Confidence Interval Length (CIL) for the estimates by sampling schemes are computed. The results are analysed by carrying out the one-sample *t*-tests (GraphPad, 2004). The *t*-test indicates whether a sample is an accurate representative of population or not.

The degree of significance is performed to check whether the difference between the parent population and the estimated versions is statistically significant or not. SE was used for *t*-tests to compare estimates from the sampling methods to the actual population. The values obtained from the *t*-test are used to calculate a *p*-value for each estimate. This is established depending on the *p* value threshold. If the *p*-value is less than threshold value, the difference is statistically significant and vice versa. In practice, the threshold value is usually set to 0.05 (an arbitrary value that has been widely adopted) in accordance to the 95% Confidence Interval (CI) (GraphPad, 2004).

The size of the discrepancy between the mean of the parent population and the mean of the sampled version depends on the size and variability of the sampled version. Statistical calculations combine sample size and variability to generate standard error of difference and the confidence interval measures. These measures have been performed to examine the degree of difference between the parent population and the sampled version for different sample sizes.

To calculate the SE of difference, it depends on the difference between the population and the sampled versions means which has a standard error. This standard error is calculated using (Ttest, 2005):

$$SE = \frac{SD}{\sqrt{n}} \tag{7.3}$$

where SD is the standard deviation and n is the number of samples.

The 95% confidence interval is the most commonly used. The estimated mean may be very close to the population mean but it possibly lies somewhat above or below the population mean and there is a 95% probability that it is within 1.96 standard errors of it. It produces lower and upper limits for the mean. The 95% CI means that one can be 95% sure that the CI include the true difference between the two means. The interval provides an indication of how much uncertainty there is in the estimate of the true mean. The narrower the CI, the shorter is the CI Length (CIL), the closer is the estimated mean to the true mean and the less variability is in the sampled version.

In addition to the above, a statistical SLA which is based on an estimation of the parameters (delay, jitter, etc) and the assessed QoS instead of population exact measurement is proposed. The purpose is to check if the packets in a specific flow conform to the guarantees given in an SLA using sampling techniques because evaluation of the whole population is, sometimes, difficult and includes more information than needed. Generally, the estimation of the long-term mean and the standard deviation of a given parameter afford some insights about the provided service quality for an application but it not sufficient to examine the SLA conformance. Another valuable parameter for evaluating the application performance is the percentile. It indicates the value below which we can assume the majority of the observed values (Choi, et al., 2003). For example, the 90% percentile gives the information that 90% of all the population members are below the percentile value. Percentile is unsuitable to quantify non-conformance because if it lies above the defined threshold, it does not provide information about what percentage of packets really violated the SLA contract (Zseby, 2005). As an alternative approach of the percentile, an estimate of the proportion (percentage (*P*)) of packets QoS value that violates the SLA contract is used (i.e. above or below a pre-defined threshold (a)). As an example, a packet with QoS value less than the threshold is considered violator (non-conformant), while packets with QoS value greater or equal to the threshold are considered conformant. The number of violators obtained after that classification is a Binomial distributed random variable (Zseby, 2005). After the packets are classified into violators and conformant packets according to the threshold (a), the calculation of the percentage of conformant (*P*) is achieved using equation 4.4.

The quality estimation is performed for all the QoS parameters and for the assessed QoS (obtained from the two evaluation systems). In order to assess the percentage estimation accuracy, there are two important quality criteria (Zseby, 2004):

i. Bias: measures how far the mean of all estimates lies from the exact value (true population) for several runs and different sample sizes using the three sampling schemes. Bias is the averaged difference over all samples of the same size. It is calculated using the following equation:

$$Bias = \frac{1}{N}\sum_{i=1}^{N}\hat{P}_i - P \tag{7.4}$$

Where N is the number of runs, $\hat{P}_i$ is the estimated values from each run and P is the exact value.

ii. Precision: this deals with the precision that can be obtained with the different sampling methods by comparing the estimates from different experiments and for several sample sizes. It measures how much these estimates scatter around the mean and it can be expressed using the Relative Standard Error (RSE) (Zseby, 2004). RSE is a measure of the estimates reliability. It is defined as the standard error of the estimate divided by the true value being estimated as shown in equation (7.5). The standard error is the square root of the variance (i.e. standard deviation) of the all runs of the same sample size.

$$RSE\left[\hat{P}\right]=\frac{\sqrt{V\left[\hat{P}\right]}}{P}=\frac{\sigma_{\hat{P}}}{P} \tag{7.5}$$

Where V and $\sigma_{\hat{P}}$ are the variance and the standard deviation of the estimated P values, respectively.

After generating the sampled populations using the three different sampling methods, a comparison is performed between these sampled versions and the parent populations. The aim is to determine which sampled version represents the parent population accurately. The comparison has been made in terms of calculating the mean, the standard deviation, the minimum, the maximum and the violators and the conformant packets percentages of the SLA contract based on bias and precision. A sampling technique is considered to be better than other techniques in terms of bias and precision if it has a smaller bias (i.e. estimated values are closer to the population values) and a higher precision (i.e. smaller RSE).

7.3.3 Network Topology and Traffic Characteristics

In order to investigate the accuracy of the three sampling methods, a suitable wireless ad hoc network was simulated. This network has the same topology, simulation settings and traffic characteristics as that used in Section 6.2.2.

Over the simulation time and in order to examine the efficiency of the measurement system, the network was subjected to three different load situations: lightly loaded during (0-170 sec), partially loaded during (171-330 sec) and fully loaded during (331-500 sec).

In order to investigate and assess the precision of each sampling method, the simulation experiments were repeated several times using different seed values of the starting points of the systematic sampling and changing the seed for the random number generator for the stratified and random sampling methods to avoid unexpected behaviour due to extreme values. Changing the seed random number essentially runs the same traffic, but will produce different timing and packet counting for the simulation. The resulting values of the different runs of the same simulation have been averaged to get the actual values. Moreover, the sampling process was repeated for several sample sizes. These samples sizes were selected to vary from small to large. In this study, all three sampling methods were implemented using packet count-based sampling triggering.

7.4 Results and Discussion

The aim of this chapter is to estimate the QoS of multimedia applications using non-intrusive measurement process based on sampling techniques. In these experiments, a comparison between the three different sampling methods described earlier for estimating one-way delay, jitter, packet loss ratio, throughput and the overall assessed QoS for a videoconferencing application transmitted over wireless ad hoc network was carried out.

7.4.1 One-way Delay

An application of the proposed approach is to estimate the end-to-end one-way delay of the traffic. Figures 7.6(a)-(c) show the population delay and sampled versions using systematic, random and stratified sampling methods, respectively. In addition, Figures 7.7(a)-(c) illustrate the frequency distributions of the actual and the sampled versions of actual delay. As an example, these versions are generated for a sample size of 200 packets (i.e. sample fraction of 1% of the actual traffic population). Sample size and sample fraction will be used interchangeably.

It can be seen from Figures 7.6(a)-(c) how the sampling approaches represent the actual delay. The degree of discrepancy is different for each sampling method. The delay was calculated using several sampling fractions. It was found that the degree of discrepancy decreases as the sample size increases. The sample fraction of 1% provides a good estimation of the delay distributions especially the distribution of the systematic and random sampling versions.

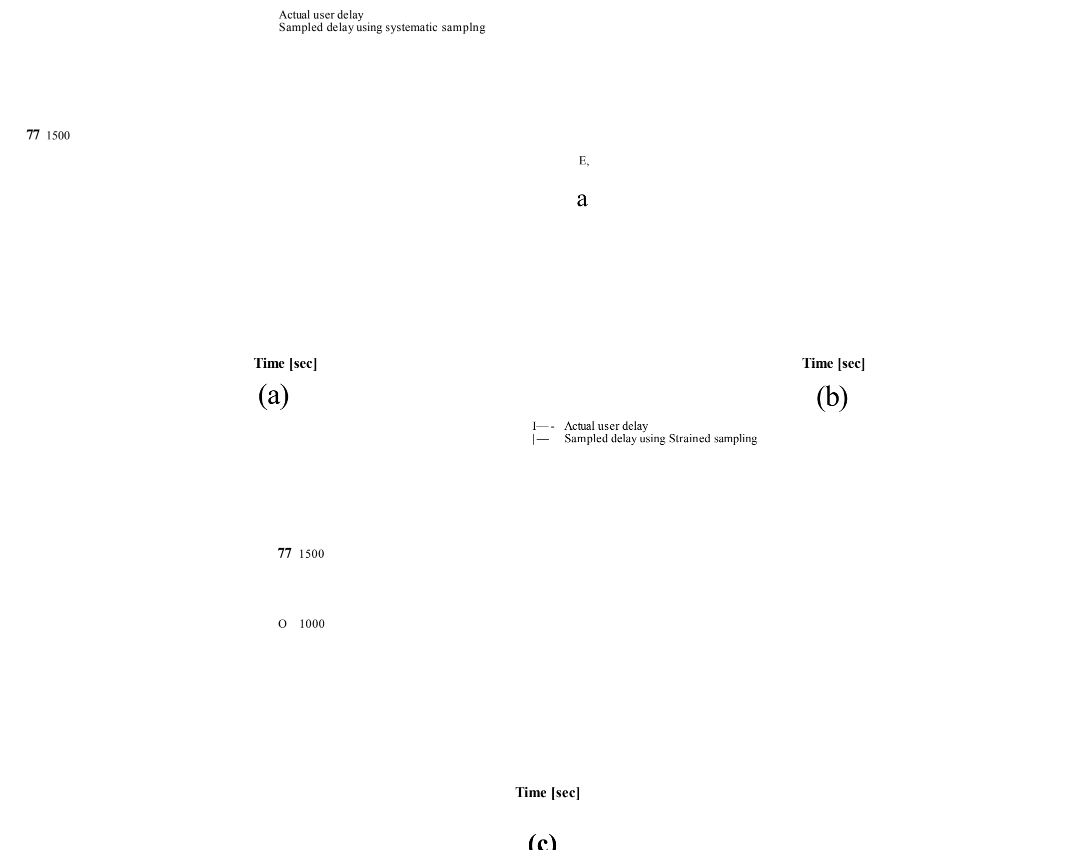


Figure 7.6: Comparison of the actual one-way delay with sampled delay versions for sample fraction of 1% using: (a) systematic, (b) random and (c) stratified sampling.

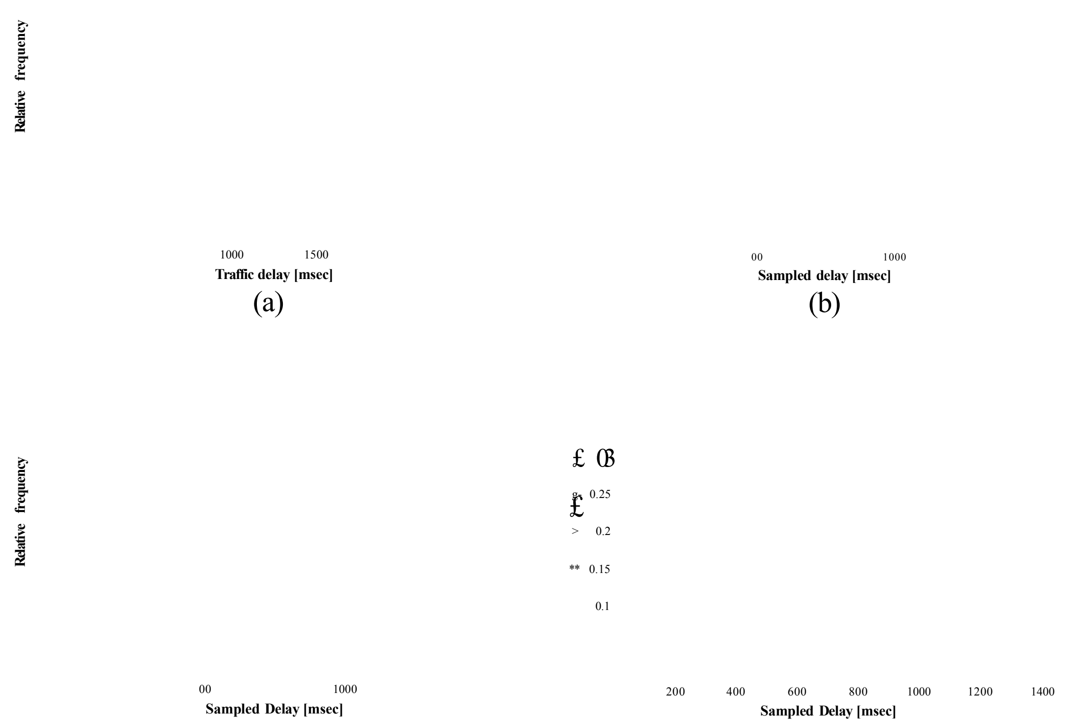


Figure 7.7: Comparison of the actual one-way delay with sampled delay versions frequency distributions using sample fraction of 1%: (a) actual, (b) systematic, (c) random and (d) stratified sampling.

As examples of the sample fractions used, the delay measurement for different sample fractions using the three sampling methods are summarised in Table 7.1(a)-(c). In this table, the mean and the St. Dev. are the average and standard deviation of each sampling method, respectively. The calculated absolute error increases as the sample size decreases and it can be seen that as the sample size increases, the variations of the sampling results from the actual mean and the actual standard deviation decrease. In addition, the estimated maximum delay approaches the actual maximum delay as the number of samples increases for the whole sampling techniques. The minimum estimated delay can be seen to be nearly equal to the actual minimum delay. The reason for those is because increasing the sample fraction will increase the number of members which will in turn increase the probability of obtaining more details of the actual population.

Table 7.1: Delay measurement results of the actual and sampled versions using: (a) systematic, (b) random and (c) stratifies sampling.

(§)

		Sample fraction [%]				
Units: [msec]	**Actual values**	**10**	**5**	**2.5**	**0.5**	**0.02**
Mean delay	316	316.2	316.4	316.7	319.5	307.5
Absolute error		0.19	0.37	0.66	3.5	8.5
St. Dev.	324.7	324.8	325	324.8	326.1	315.9
Maximum delay	2230.9	2063.7	1899.4	1662.6	1250.9	1056.6
Minimum delay	2.5	2.5	2.5	2.5	2.6	2.6

M

		Sample fraction [%]				
Units: [msec]	**Actual values**	**10**	**5**	**2.5**	**0.5**	**0.02**
Mean delay	316	316.8	315.7	312.3	317.2	321.4
Absolute error		0.79	0.28	3.7	1.2	5.4
St. Dev.	324.7	325.7	324.6	323.7	326.1	335
Maximum delay	2230.9	2013.	1896	1609.2	1227.2	1117.3
Minimum delay	2.48	2.5	2.5	2.5	2.5	2.5

M

		Sample fraction [%]				
Units: [msec]	**Actual values**	**10**	**5**	**2.5**	**0.5**	**0.02**
Mean delay	316	316	315.8	316.3	308.6	305.1
Absolute error		0.03	0.17	0.26	7.4	10.9
St. Dev.	324.7	325	324.7	324.3	318.7	315.8
Maximum delay	2230.9	2071.3	1902.3	1552.6	1108.1	1016.1
Minimum delay	2.5	2.5	2.5	2.5	2.6	2.6

As mentioned earlier, the mean and the standard deviation of the sampled versions may not give sufficient information about the estimation accuracy and which sampling method results can precisely represent the population as a function of sample size. Bias and precision (i.e. equations 7.4 and 7.5) were used to evaluate the three sampling methods in terms of the mean and percentage of violator packets of the SLA thresholds. Figure 7.8 depicts the bias of estimates of the sampled versions mean from the actual mean with up to 15 runs of the simulations. The Figure shows the results for all sampling schemes and for different sample fractions. The results illustrate that the bias is very small for all schemes for sample fractions larger than 0.5% (i.e. a sample size more than 100 packets). In addition, as expected it rapidly decreases for large sample sizes and becomes close to zero. From this Figure, it is clear that systematic sampling has the lowest bias compared to other schemes. Both systematic and stratified have more stable and smooth bias than the random type. Random sampling has some comparatively high negative bias values for sample fractions of 4% and 8%. Nevertheless, the bias is still quite small which is less than 4msec.

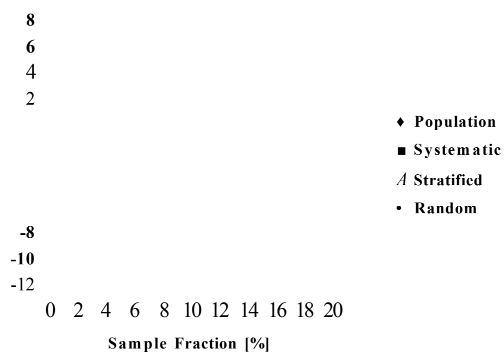


Figure 7.8: Bias of the sampled delay mean estimates.

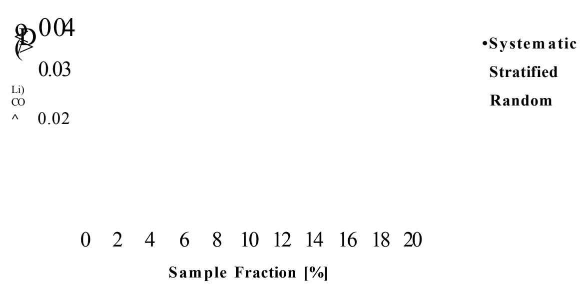


Figure 7.9: RSE of the sampled delay mean estimates.

Figure 7.9 illustrates the precision of the estimates of the sampled mean expressed by calculating the RSE as function of the empirical standard deviation of the estimates from all runs. It is apparent that systematic sampling provides the most accurate estimates compared to the stratified and random approaches. In addition, stratified sampling affords more precise estimates than the random sampling scheme. This means that random sampling measured delay value have larger deviations from the actual delay mean.

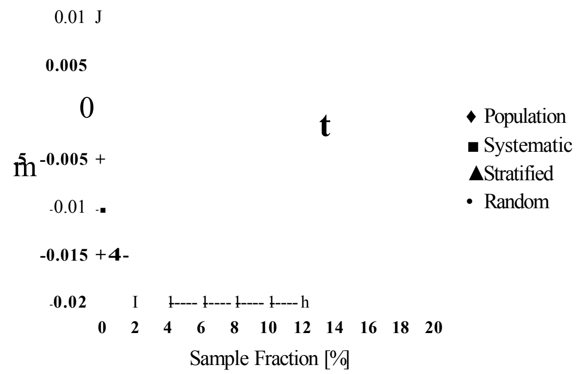


Figure 7.10: Bias of the sampled delay violator proportion from the actual delay violator proportion.

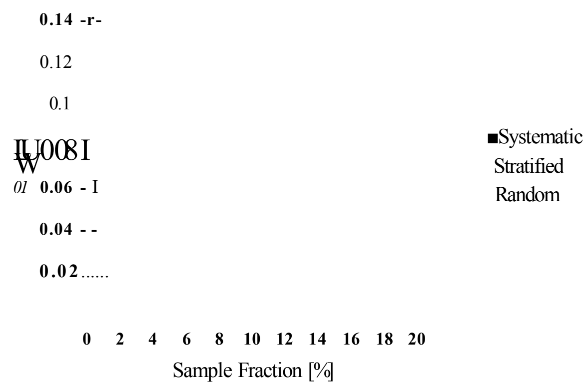


Figure 7.11: RSE of the sampled delay violator proportion from the actual delay violator proportion.

An investigation of whether a bias is introduced when estimating the proportion of violators of a delay threshold from sampled values and the precision that can be achieved with the different sampling techniques is shown in Figures 7.10 and 7.11, respectively. As an example, a delay threshold of 450msec was used in the experiments. This value was selected to represent the threshold that is required to receive a medium quality of the videoconferencing application. The resulting proportion of violators of the actual population delay was 0.4. It is apparent from Figure 7.10 that the bias is very small for all sampling schemes and it further decreases with the large sample fractions. As it can be seen, systematic sampling has the least bias and minimum variations followed by the stratified sampling. As in Figure 7.8, random sampling has high bias values for *4%* and 8% sample fractions. From Figure 7.10, it is clear that systematic sampling has the minimum RSE (highest precision) for estimating the percentage of violators of the delay threshold (450msec).

The above results indicate that systematic sampling has the best performance in terms of bias and precision. This may be due to the fact that there was nearly no influence of periodicity between the subsequent sampled delay values. Because if there was any periodicity between them, there would be very high bias values and the precision would be very low.

7.4.2 One-way Delay Variation

Another application of the sampling methodologies is to estimate the end-to-end jitter for a specific application. Figure 7.12(a)-(d) present the frequency distributions of the actual traffic in addition to the sampled jitter versions using systematic, random and stratified sampling schemes. These diagrams were obtained with a sample fraction of 1%. The distributions exhibit that the majority of the actual jitter population values were less than 20msec which is the same information that the sampling methods provided (i.e. most of the sampled packets had jitter values less than 20msec). This means that sampling technique can afford a good estimation of the actual traffic jitter distribution.

J”
=0 5

Traffic Jitter [msec]

(a)

40 60
Sampled Jitter pnsec]

(b)

Sampled jrtter [msec]

(c)

Sampled Jitter [msec]

(d)

Figure 7.12: Comparison of the actual jitter with sampled jitter versions distributions using sample fraction of 1%: (a) actual, (b) systematic, (c) random and (d) stratified sampling.

As an example of the sample fractions used, the jitter measurement for different sample fractions are summarised in Tables 7.2(a)-(c). From the Tables, the calculated absolute error increases as the sample fraction decreases. Generally, as the sample fraction increases, the variations of the sampling results from the actual mean and actual standard deviation decrease. In addition, the estimated maximum jitter approaches the actual maximum jitter as the number of samples increases for the whole sampling techniques. The minimum estimated jitter is equal to the actual minimum jitter but starts to rise as the number of samples decreases. The reason behind these observations is that increasing the number of samples will increase the number of packets included in the estimation which will increase the probability of tracking the actual population variations.

Table 7.2: Jitter measurement results of the actual and sampled versions using: (a) systematic, (b) random and (c) stratifies sampling.

(a)

		Samp e fraction [%]				
Units: [msec]	**Actual values**	**10**	**5**	**2.5**	**0.5**	**0.02**
Mean jitter	6.2	6.2	6	6.2	6.5	7
Absolute error		0.02	0.13	0.01	0.37	0.8
St. Dev.	14.6	15.3	12	12.1	13.1	13
Maximum jitter	733.2	440.6	207.7	157.9	96.9	77.5
Minimum jitter	0	0	0	0	0.06	0.07

(b)

		Samp e fraction [%]				
Units: [msecl	**Actual values**	**10**	**5**	**2.5**	**0.5**	**0.02**
Mean jitter	6.2	6.3	6	6	6.3	5.8
Absolute error		0.09	0.2	0.2	0.2	0.4
St. Dev.	14.6	15.6	12.7	11.1	10.8	8.7
Maximum jitter	733.2	426.9	242.7	121.8	76.9	46.2
Minimum jitter	0	0	0	0	0.03	0.05

		Sample fraction [%]				
Units: [msec]	**Actual values**	**10**	**5**	**2.5**	**0.5**	**0.02**
Mean jitter	6.2	6.1	6.2	5.8	5.6	5.5
Absolute error		0.03	0.08	0.3	0.5	0.6
St. Dev.	14.6	14.4	13.3	10.5	8.8	7.3
Maximum jitter	733.2	338.4	242.4	128.4	62.3	38
Minimum jitter	0	0	0	0	0.03	0.09

As for the delay, bias and precision were used to evaluate the accuracy of the three sampling methods in terms of the estimated mean jitter and the estimated percentage of the violator packets of the SLA threshold. Figure 7.13 depicts the bias of the jitter mean estimates from the actual mean based on 15 simulations runs. The Figure shows the output results for all sampling techniques and for several sample fractions. The results reveal that the bias is very small for all schemes and for all examined sample fractions. In addition, as expected it decreases further for large sample fractions and approaching zero. From this Figure, it is clear that all the sampling schemes have some comparatively high bias values at some sample fractions. Nevertheless, the bias is still quite small i.e. it is less than 0.8 msec.

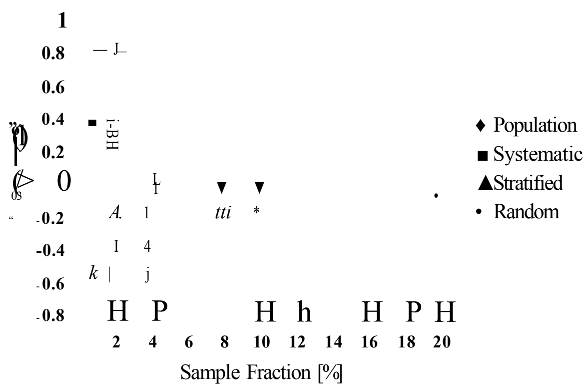


Figure 7.13: Bias of the sampled jitter mean estimates.

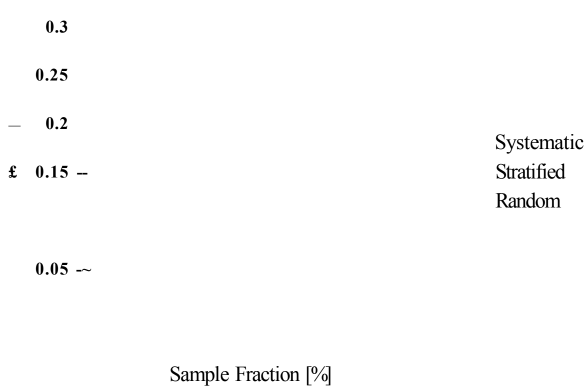


Figure 7.14: RSE of the sampled jitter mean estimates.

Figure 7.14 illustrates the precision of the estimates of the sampled jitter mean expressed by calculating the RSE as a function of the empirical jitter standard deviation of the estimates from all runs. Noticeable is that stratified sampling produces the most accurate estimates compared to the systematic and random approaches and over the whole sample fractions. Also, systematic sampling affords more precise estimates than the random sampling scheme. This means that the jitter values based on random sampling had larger variations over the 15 simulations runs.

An exploration of whether a bias is introduced when estimating the proportion of violators of a jitter threshold (SLA) from the jitter sampled values and the precision that can be obtained with the different sampling techniques are shown in Figures 7.15 and 7.16, respectively. The jitter threshold of 20msec was used in the experiments. This value was chosen to characterise the threshold which is required to perceive medium jitter quality of the videoconferencing application. The resulting proportion of violators of the actual population jitter was 0.07.

In general, Figure 7.15 reveals that the estimated bias is very small for all sampling schemes and it is sharply decreased for large sample fractions. As can be seen, systematic sampling has the lowest bias and minimum variations followed by the stratified sampling. The bias, in most cases, is in the range of [-0.005 to 0.005] which means the estimated violators are very close to the real one. Random sampling has a comparative high negative bias value at 4% sample fraction which might be due to some correlations between the subsequent jitter samples which were smaller than actual jitter value. Because if there was any correlation between them, this will produce very high bias values and the precision will be very low.

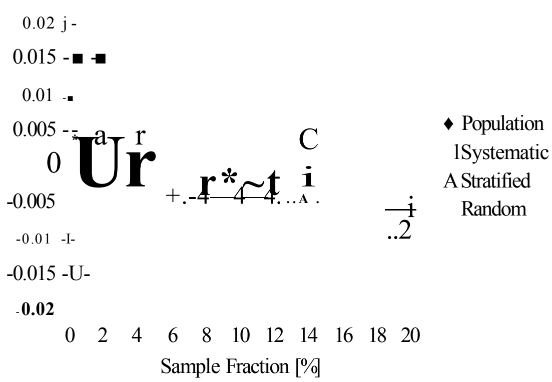


Figure 7.15: Bias of the sampled jitter violator proportion from the actual jitter violator proportion.

Figure 7.16 illustrates that all sampling schemes provide a good precision for estimating the percentage of violators of the jitter threshold (20msec). The RSE seem to be relatively high compared to the RSE of the delay (Figure 7.11). This is due to the division process in equation 7.5 by the real proportion which is very small for the jitter real proportion (0.073).

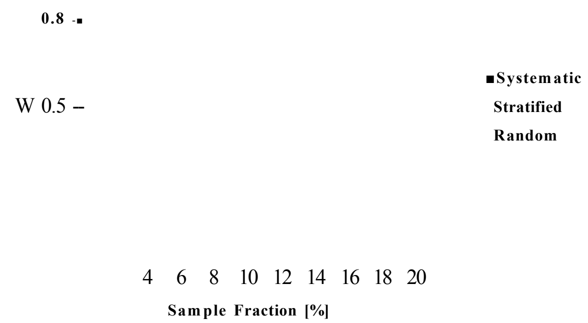


Figure 7.16: RSE of the sampled jitter violator proportion from the actual jitter violator proportion.

The above results show that the three sampling schemes were accurate in terms of the bias and precision estimation of the jitter measurements.

7.4.3 Packet Loss

Contrary to the procedure followed to estimate the delay and the jitter, packet loss ratio is not estimated directly from the sampled versions. Loss ratio is computed by counting the lost and the total sent packets over all measurement intervals. A measurement interval is the interval between every two successive samples. The advantage of using the measurement interval is that it can provide a more precise instantaneous expression method rather than the long-term packet loss average with different resolutions depending on the number of samples. Therefore, the method can be used to classify periods that contain lost packets (loss periods) and others without losses (loss-free periods) and their lengths in seconds.

The actual traffic packet losses were calculated using the windowing technique. Window size selection plays an important role in presenting the level of the measurement details. Choosing a very small window size provides a very detailed measurement, whilst a large size hides a lot of these details. In our experiments, a medium window size of 20 packets was selected. Figures 7.17(a)-(d) illustrate the actual packet losses in addition to the estimated ratios using the sampling algorithms with a sampling fraction of 1%. These diagrams demonstrate how effectively the proposed estimation approach tracks the actual traffic losses.

Time [sec]
(a)

Time [sec]
(b)

o
E
o

CL

CL 20

Time [sec]
(c)

Time [sec]
(d)

Figure 7.17: Comparison of the: (a) actual packet loss ratio with versions obtained using: (b) systematic, (c) random and (d) stratified sampling based on sample fraction of 1%.

It can be observed, from the results obtained, that during the first 170 seconds of the simulation, there were no packet losses and all the estimated results using sampling reported this. There were no packet losses because during that period the network load was light. After 170 seconds and up to 330 seconds, the load was higher; thus some losses occurred sometimes during the simulation. To some extent the sampling versions could estimate those losses. Beyond 330 seconds, the network was fully saturated and so, high losses were observed. When the sample fraction was increased to more than 1%, more details of the loss ratio were exhibited. A higher sample fraction results in higher loss resolution and detail. The frequency distributions of these loss ratios are shown in Figure 7.18(a)-(d). These distributions (actual and estimated) reveal that more than 65% of the monitored loss ratios were less than 2%. These graphs indicate how accurate the estimated losses were compared to the actual losses.

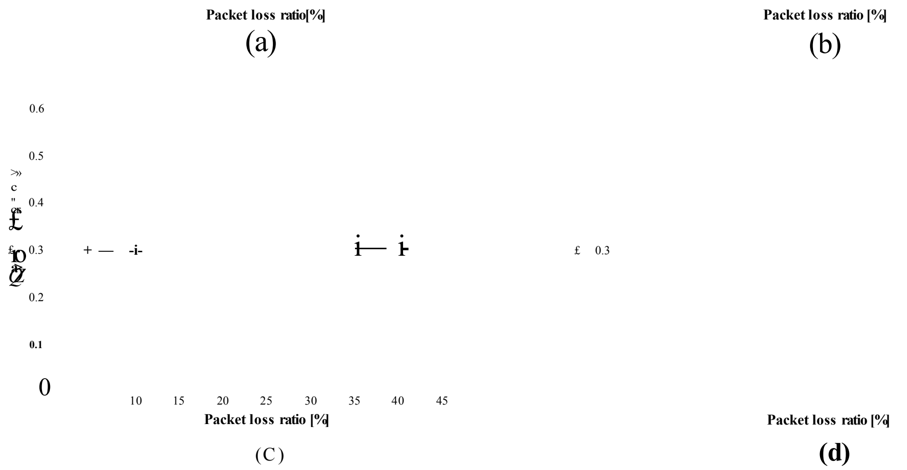


Figure 7.18: Comparison of the: (a) actual packet loss ratio frequency distribution with the distribution versions obtained using: (b) systematic, (c) random and (d) stratified sampling based on sample fraction of 1%.

Table 7.3: Mean packet loss ratio measurement results using different sampling methods.

Sample fraction (%)	**Systematic sampling [%]**		**Random sampling [%]**		**Stratified sampling [%]**	
	Mean	Abs. Error	Mean	Abs. Error	Mean	Abs. Error
10	7.1	0.02	7.1	0.08	7	0.16
5	7.6	0.42	7.5	0.33	7.5	0.31
2.5	8	0.78	7.9	0.72	8	0.78
0.5	8.1	0.90	8.1	0.89	7.8	0.6
0.02	7.9	0.76	8.2	1	7.3	0.34

The mean packet loss ratio of actual traffic was 7.16%. The results of the packet loss ratio measurements for different sample fractions using the three sampling methods are summarised in Table 7.3. This is obtained by calculating the packet loss ratio over each measurement interval (i.e. between every two successive samples) and then taking the average of the loss ratios for all intervals. The absolute error between the actual and the estimated ratio is calculated for every sample fraction as shown in the table. As expected, the absolute error reduces with the increase of sample fractions. In some situations, the random and stratified sampling may not be suitable to loss calculation due the randomness feature in the sample selection process. Due to randomness, the selected samples may be very near to each other which means that calculated loss ratio for the measurement interval between these samples may be high or low which will overestimate or underestimate the mean of the losses over the whole measurement intervals. This case occurred in the stratified sampling results as shown in Table 7.3. Both systematic and random schemes provided a good estimation of the actual loss. Decreasing the sample fractions in systematic sampling will result in biasing the estimation result, as decreasing the sample fraction will increase the measurement interval which will not provide enough details about the traffic losses occurring. This appears in the systematic sampling of the 0.02% sample fraction.

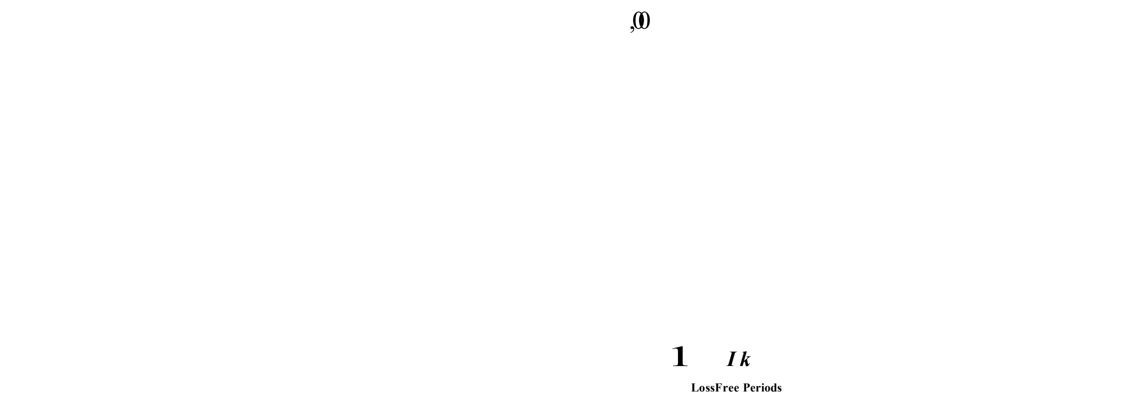
LiuAti

Figure 7.19: Frequency distributions of the length of loss and loss-free periods versus time during the measurement period using systematic sampling with: (a) 1% and (b) 10% sample fractions.

(a) (b)

Figure 7.20: Frequency distributions of the length of loss and loss-free periods versus time during the measurement period using random sampling with: (a) 1% and (b) 10% sample fractions.

Another powerful representation of the packet loss using the sampling methods is depicted in Figures 7.19-7.21. These figures illustrate the distribution of average burst lengths of the loss and loss-free periods based on 1% and 10% sample fractions. This representation gives information as how many loss and loss-free periods occurred and the length of each period as a function of sample fraction. These diagrams reveal more details can be that obtained from larger sample fractions. The ratios between the loss and loss-free periods and the total measurement time with sampling fraction of 1% for the systematic, random and stratified sampling methods are 0.5 and 0.4, 0.49 and 0.4, and 0.48 and 0.41, respectively. While for sample fraction of 10%, these values are 0.64 and 0.28, 0.64 and 0.27, and 0.64 and 0.26, respectively. Therefore, all sampling schemes provided almost similar observations.



(a) (b)

Figure 7.21: Frequency distributions of the length of loss and loss-free periods versus time during the measurement period using stratified sampling with: (a) 1% and (b) 10% sample fractions.

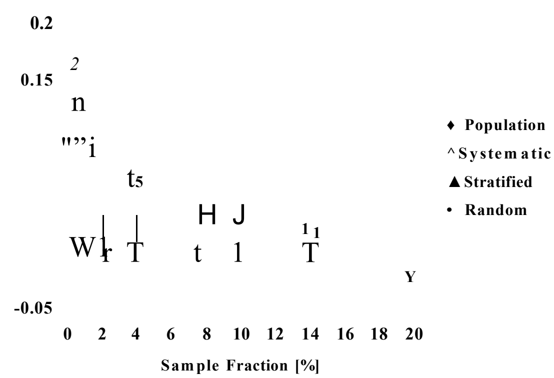


Figure 7.22: Bias of the estimated packet loss ratio violator proportions from the actual packet loss ratio violator proportion using different sampling schemes.

Similar to the delay and jitter, sampling methods were also used to validate the SLA packet loss ratio by estimating the percentage of loss ratios that violate the SLA threshold contract. Figures 7.22 and 7.23 show the bias and the RSE of the estimated proportions for different sample fractions. The SLA packet loss ratio threshold used for the estimation was 2%. The proportion of the actual traffic packet loss ratio was 0.26. Obviously, the estimated bias and RSE are exponentially decreasing as increasing the sample fraction. The bias values are small for all schemes. All sampling schemes showed relatively equal, likely bias estimations. The precision of packet loss ratio estimations that can be achieved with sampling is shown in Figure 7.23. This Figure illustrates that, in most cases, the smallest errors were obtained by the systematic sampling. This can be explained by the stability of the length of space (i.e. number of packets) between the two successive samples in contrast to the other sampling schemes where the space length is variable depending on the positions of the sampled packets.

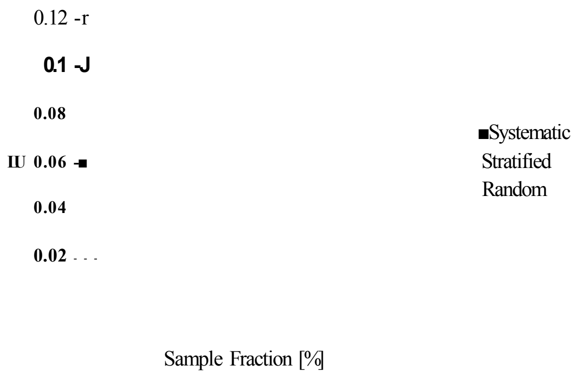


Figure 7.23: RSE of the packet loss ratio violator proportion using different sampling schemes.

7.4.4 Throughput

In addition to the long-term average and using the sampling concept, the maximum and the minimum values as well as the variation of the utilised capacity (throughput) of a specific traffic application can be obtained. Similar to the process of packet loss ratio calculation, every throughput value is calculated based on counting the number of received packets between two successive sampled packets, multiplying the result by the packet size and dividing by the time difference between the two samples timestamps.

Tables 7.4(a)-(c) show the mean, standard deviation, maximum and minimum of the throughput for different sample fractions. The higher the sample fraction the more detail about traffic behaviour can be obtained. Moreover, the higher the sample fraction the smaller the measurement interval and so, the higher the mean, the variance and the maximum and the lower the minimum of the estimated throughput values. These results depend on the network load situation; in lightly loaded situations, the variations of the estimated values are very small therefore the calculated results will be in the same range as can be seen from Figures 7.24(a)-(f). However, in medium and saturated conditions, there will be variations in the estimated throughput and these variations depend on the distance between the successive samples. Because, for example, increasing the sample fraction reduces the time difference between the successive samples, which in turn raise the calculated throughput. In general, Tables (a)-(c) and Figures (a)-(f) reveal that sampling methods perform similarly in throughput estimation.

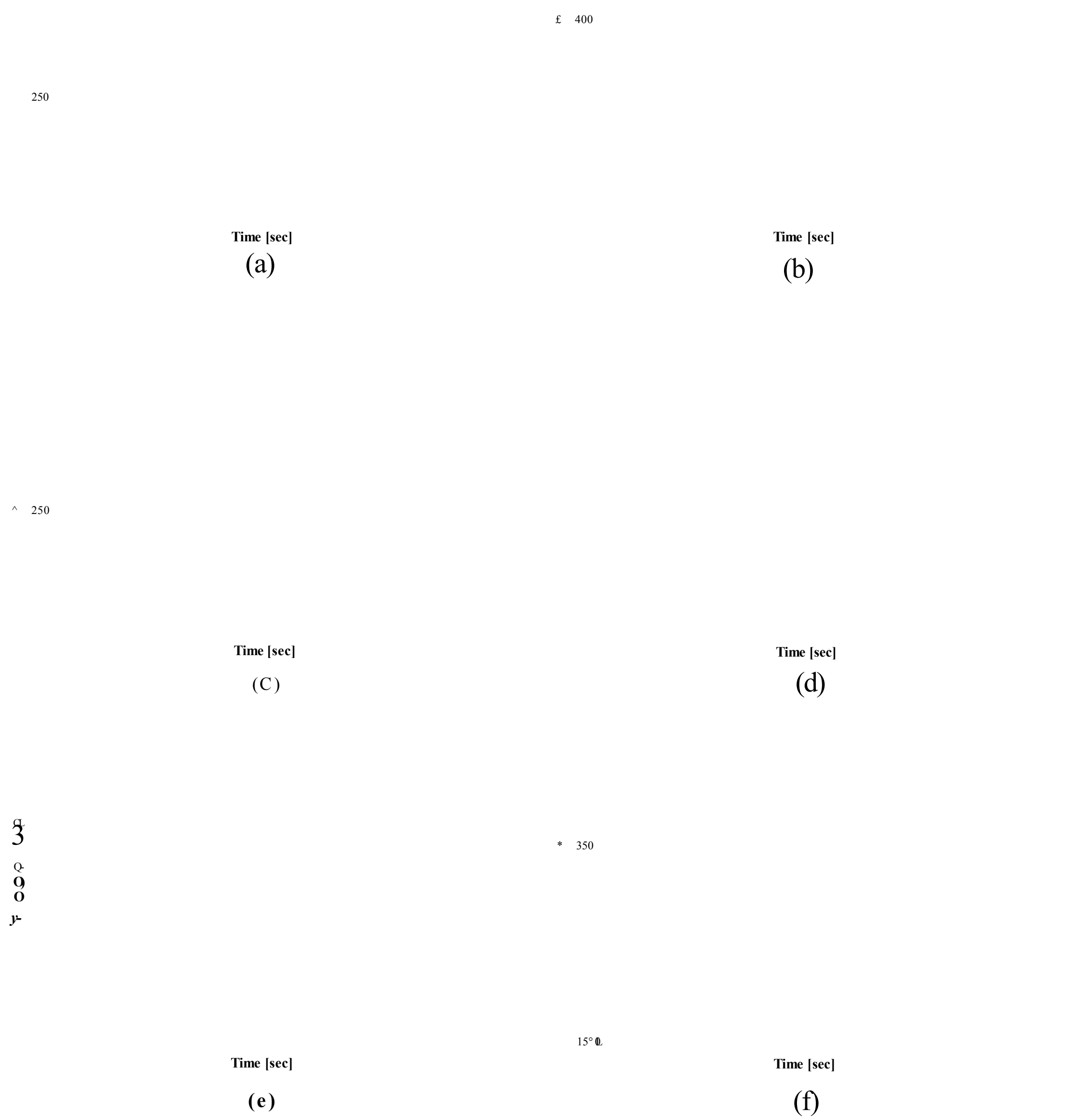


Figure 7.24: Comparison of the throughput estimations using sampling techniques with different sample fractions: systematic: (a) 1% and (b) 10%, random: (c) 1% and (d) 10% and stratified: (e) 1% and (f) 10%.

Table 7.4: Throughput measurement results using different sampling methods: (a) systematic, (b) random and (c) stratifies sampling.

(a)

Units: [Kbps]	Sample fraction [%] 10	5	2.5	0.5	0.02
Mean throughput	353	348.8	345.6	344.3	344.7
St. Dev.	72	62.8	55.2	49.9	48.3
Max. throughput	671.6	577.7	461.9	395.1	383
Min. throughput	69.4	102	141.5	206.4	214.5

(b)

Units: [Kbps]	Sample fraction [%] 10	5	2.5	0.5	0.02
Mean throughput	353.9	349.2	345.9	344.3	343.8
St. Dev.	74.2	63.7	55.9	49.7	50
Max. throughput	750.9	584.2	482.5	394.7	388.9
Min. throughput	64	103	142.1	204.7	211.2

M

Units: [Kbps]	Sample fraction [%] 10	5	2.5	0.5	0.02
Mean throughput	356.1	350.3	346.5	345.5	347.7
St. Dev.	81.2	67.6	59.1	49.1	46
Max. throughput	1174.1	813.1	599.8	405.1	391.2
Min. throughput	48.1	84	139.5	206	216.7

Figures 7.25(a)-(f) show the frequency distributions of the throughput for two different sample fractions: 1% and 10%. From the Figures, the distributions provide a good estimate of the actual traffic transmission rate (i.e. 384 Kbps). Furthermore, an improved resolution is produced using the sample fraction of 10%. The desired throughput resolution depends on the required accuracy. All sampling methods, to some extent, provide similar measurement frequency distribution results.

In addition to the above obtained threshold results, sampling methods are useful in the validation of the SLA contract. Figures 7.26 and 7.27 display the bias and the RSE of the estimated proportions for different sample fractions. The SLA throughput threshold used for the estimation process was 370Kbps. This threshold was selected to represent a packet delivery ratio of 95% (i.e. 95% of the generated packet is received). The proportion of the actual traffic throughput was 0.44. It can be observed that the bias values are small for all schemes. Besides, all sampling schemes have equally well the bias estimations for sample fractions larger than 1%. Figure 7.27 illustrates that, as observed in loss estimations, the smallest errors were obtained by the systematic sampling. This is because space (i.e. number of packets) between the two successive samples is a constant as compared with the sampling schemes where the space is variable, which depends on the positions of the sampled packets.

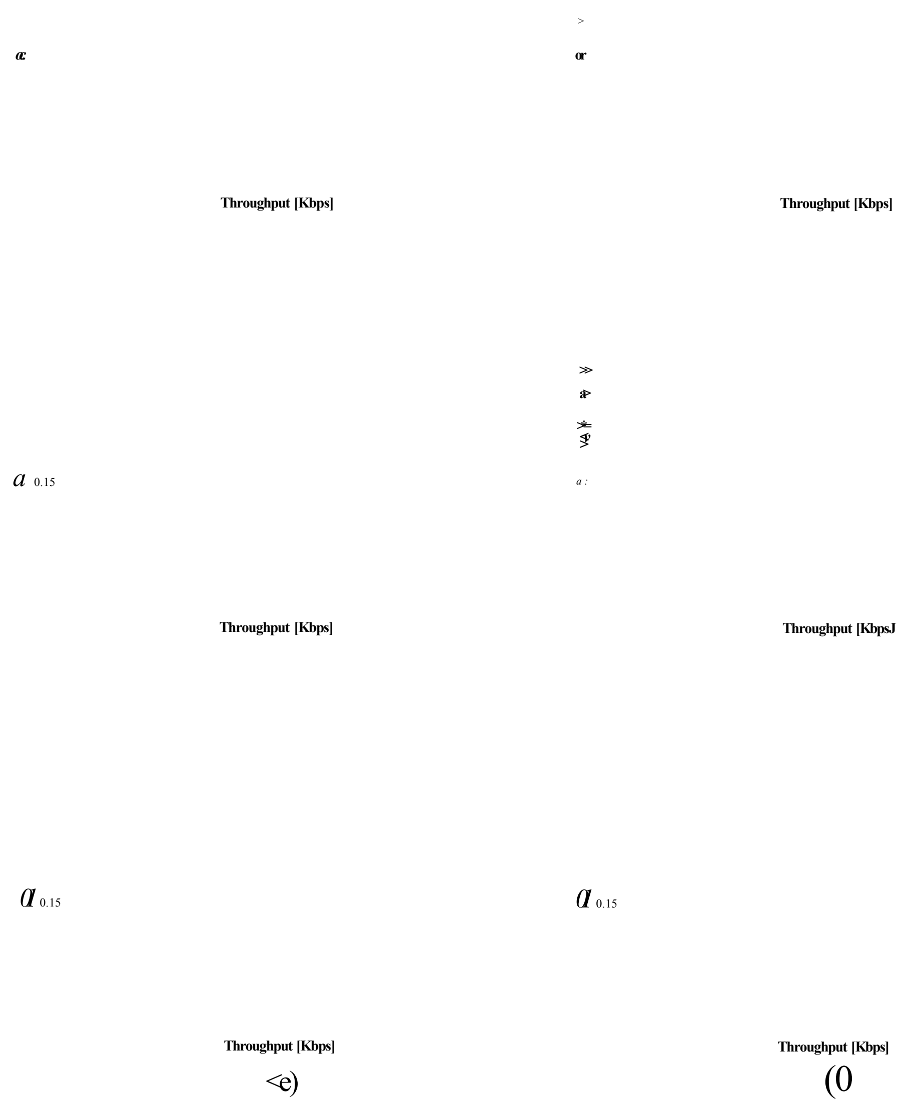


Figure 7.25:Comparison of the throughput frequency distributions using sampling techniques with different sample fractions: systematic: (a) 1% and (b) 10%, random: (c) 1% and (d) 10% and stratified: (e) 1% and (f) 10%.

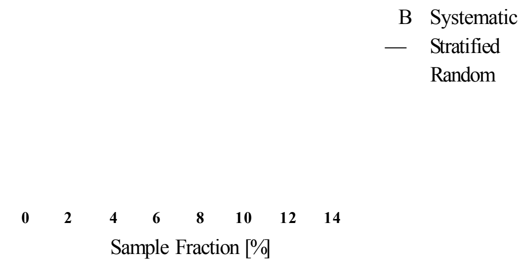


Figure 7.26 Bias of the estimated threshold violator proportions from the actual threshold violator proportion using different sampling schemes.

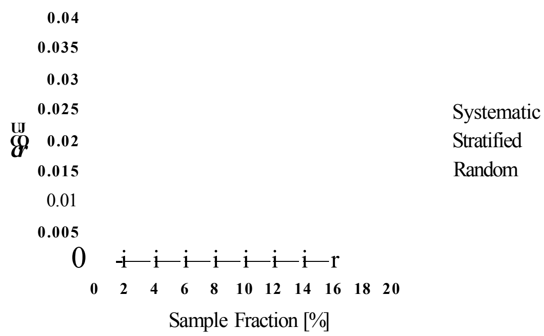


Figure 7.27: RSE of the threshold violator proportion using different sampling schemes.

7.4.5 Overall QoS

The main objective of this part is to assess the QoS of multimedia traffic using fuzzy logic and distance assessment systems based on sampling techniques and to examine the ability of sampling schemes to provide an accurate representation of the actual traffic QoS. In addition, and by relying on the assessed QoS using sampling, a validation of the user QoS guarantees (i.e. SLA contract) is also examined. The estimated QoS is obtained by calculating the QoS parameters of the sampled packets and feeding the results to the evaluation systems. The most significant parameters which influence the QoS for videoconferencing are the delay, jitter and packet losses. Figures 7.28(a)-(h) show the instantaneous actual QoS and the QoS estimated using sampling methods. These figures were produced for sample fraction of 1%. Tables 7.5(a)-(f) illustrates the results of the QoS measurements for different sample fractions using the three sampling methods. The figures and tables indicate that the estimated QoS can infer the actual user QoS during the whole period of measurement and for the different network traffic load situations.

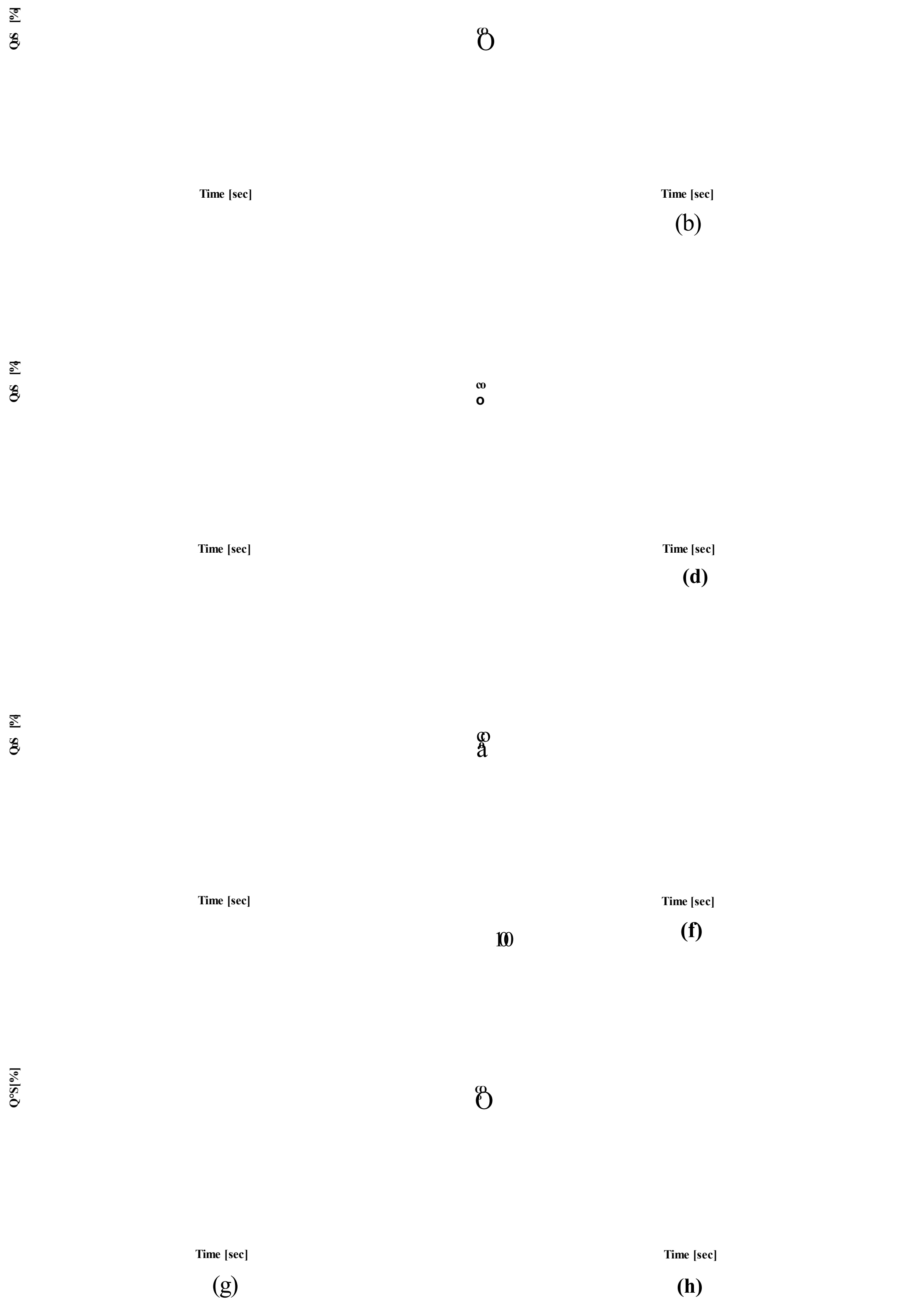


Figure 7.28: Comparison of the actual QoS with estimated versions based on sample fraction of 1% using: 1- Fuzzy: (a) actual, (c) systematic, (e) random and (g) stratified sampling and 2- Distance: (b) actual, (d) systematic, (f) random and (h) stratified sampling.

Moreover, both evaluation systems provided a good QoS estimation as compared to each other using the 1% sample fraction. As can be observed from the tables and the figures, the Distance approach produced outputs which are related but not identical to the Fuzzy system values range of 5-10%. For example, the maximum and the minimum values of the Distance approach are approximately 100 and zero respectively, while the values from the Fuzzy system are 90.5 and 9.2 respectively (the reasons why these differences were discussed in Chapter 5). Nearly, all the sampling schemes provided similar results when they were compared with each other using both evaluation systems.

The network was loaded with three different traffic loads during the simulation. During the first 170 seconds, the network was lightly loaded, moderate load during the 170-350 seconds and was heavily loaded during of 350-500 seconds. So, the network should provide the videoconferencing application with a good QoS for the first period, average QoS over the second period and poor QoS throughout the third. These results were validated as can be observed from Figures 7.28(a)-(h). Furthermore, Tables 7.5-7.7 reveal that the long-tem averages, standard deviations, maximum and minimum estimated QoS closely resemble the actual traffic QoS. The three sampling methods indicate that as the sample fraction increases, the absolute error decreases.

Table 7.5: QoS measurement results using systematic sampling method using: (a) Fuzzy system and (b) Distance system.

(a)

		Samp e fraction [%]				
Units: I%]	**Actual**	**10**	**5**	**2.5**	**0.5**	**0.02**
Mean QoS	55.3	54.9	54.8	54.7	51.96	51.2
Absolute error		0.37	0.45	0.61	2.7	4.1
St. Dev.	36.6	36.5	36.6	36.7	38.25	38.2
Maximum QoS	90.5	90.5	90.5	90.5	90.52	90.5
Minimum QoS	9.3	9.3	9.3	9.3	9.3	9.3

M

		Samp e fraction [%]				
Units: [%]	**Actual**	**10**	**5**	**2.5**	**0.5**	**0.02**
Mean QoS	61.8	61.5	61.3	60.7	57.9	56.8
Absolute error		0.28	0.56	1.1	3.9	5
St. Dev.	41.1	41.4	41.6	42.1	42.9	42.4
Maximum QoS	99.4	99.9	99.9	99.9	99.9	99.9
Minimum QoS	0	0	0	0	0.32	1.9

Table 7.6: QoS measurement results using random sampling method using: (a) Fuzzy system and (b) Distance system.

(a)

Units: [%]	Actual	Samp e fraction [%] 10	5	2.5	0.5	0.02
Mean QoS	55.3	54.8	54.9	55.1	51.8	50.9
Absolute error		0.50	0.36	0.18	3.5	4.3
St. Dev.	36.6	36.5	36.5	36.7	38	38.8
Maximum QoS	90.5	90.5	90.5	90.5	90.5	90.5
Minimum QoS	9.3	9.3	9.3	9.3	9.3	9.3

(b)

Units: [%]	Actual	Samp e fraction [%] 10	5	2.5	0.5	0.02
Mean QoS	61.8	61.4	61.7	61.4	59.7	55.3
Absolute error		0.42	0.16	0.45	2.3	6.6
St. Dev.	41.1	41.3	41.5	41.5	42.4	42.9
Maximum QoS	99.4	99.9	99.9	99.9	99.9	99.9
Minimum QoS	0	0	0	0	0	4.4

Table 7.7: QoS measurement results using stratified sampling method using: (a) Fuzzy system and (b) Distance system.

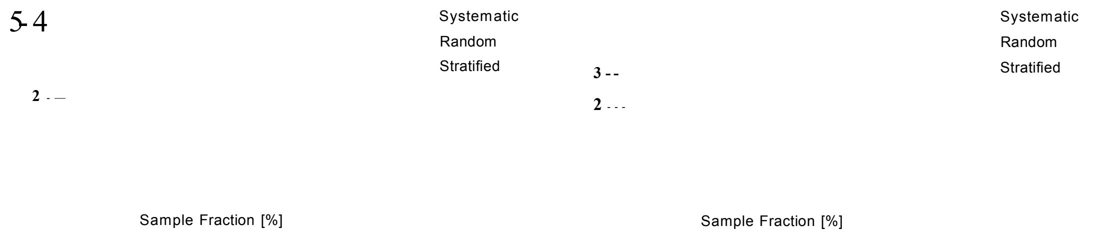
(a)

Units: [%]	Actual	Samp e fraction [%] 10	5	2.5	0.5	0.02
Mean QoS	55.3	54.9	54.8	54.7	52.8	51.4
Absolute error		0.37	0.89	0.59	2.5	3.9
St. Dev.	36.6	36.5	36.5	36.8	37.9	38.4
Maximum QoS	90.5	90.5	90.5	90.5	90.5	90.5
Minimum QoS	9.3	9.3	9.3	9.3	9.3	9.3

M

Units: [%]	Actual	Sample fraction [%] 10	5	2.5	0.5	0.02
Mean QoS	61.8	61.6	61.3	60.9	58.7	56.7
Absolute error		0.18	0.47	0.9	3.2	5.2
St. Dev.	41.1	41.3	41.6	42	42.5	42.5
Maximum QoS	99.4	99.9	99.9	99.9	99.9	99.8
Minimum QoS	0	0	0	0	1.1	2.7

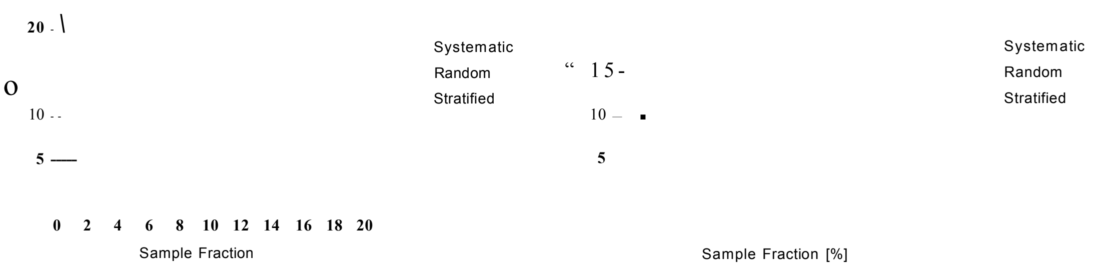
To examine the effect of sample fractions on the accuracy of sampling techniques, the statistical significance of difference, SE of difference and CIL were calculated. All the sampling techniques produced p values greater than 0.05 for all of the sample fractions. This means that the differences between the actual QoS and the estimated versions were not statistically significant. Figures 7.29(a) and (b) show the estimated QoS variation of the SE with a sample fraction for the three sampling methods. From the figure, it is obvious that there is a large drop in the error as the sample fraction is increased. The error has a high value when the sample fraction is small, and it decreases as the sample fraction increases. The larger the sample fraction, the more stable the SE. Figure 7.29(a) and (b) also show no explicit difference between the performances of the three sampling techniques.



(a) (b)

Figure 7.29: Estimated QoS variation of SE with sample fraction for the three sampling methods using the two measurement approaches: (a) Fuzzy and (b) Distance.

The 95% Cl was also computed, this shows that there will be 95% certainty that the Cl includes the difference between the two means. The interval provides an indication of how much uncertainty there is in the estimate of the true mean. All the calculated CIs contain the difference. In addition, the CIL variation with sample fraction is shown in Figure 7.30. The narrower the Cl, the shorter is the Cl Length (CIL), the closer is the estimated mean to the true mean and the less variability is in the sampled version. From the Figure, it is clear that CIL has an extreme drop for low values of sample fractions. There is less variability in CIL as the sample fraction is increased. The minimum CIL was at the largest sample fraction. From the figure, all the three methods have close CIL values for the same sample fraction.



(a) (b)

Figure 7.30: Estimated QoS variation of CIL with sample fraction for the three sampling methods using the two measurement approaches: (a) Fuzzy and (b) Distance.

QoS using sampling Actual QoS

PO(r

5

£ 30

Average

(a)

QoS using sampling Actual QoS

QoS level

(b)

(c)

(d)

£ 30

(e)

(f)

Figure 7.31: QoS bar charts using different sampling methods with 1% sample fraction: 1- Fuzzy: (a) systematic, (c) random and (e) stratified sampling and 2- Distance: (b) systematic, (d) random and (f) stratified sampling.

Another, more specific investigation about the degree by which the estimated QoS with sampling could characterise the actual QoS was carried out using the bar chart distributions. These distributions were used to quantify the level that the QoS was poor, average and good. Figure 7.31(a)-(f) demonstrate the bar charts of both the actual and the estimated QoS using the two assessment systems with sample fraction of 1%. All sampling schemes, for both systems, provide a good estimation and representation of the user traffic performance even for small sample fraction. In the worst case the difference between the estimated and the actual was 5%. As expected, the higher the sample fraction, the more accurate the representation will be.

In addition to the above investigations, in order to quantify how much the QoS was poor, average and good, after classification, the mean and standard deviation of each region were computed. Tables 7.8-7.10 show these statistics and characterise each region of the actual and the estimated QoS values using two different sample fractions. The results, from the three sampling approaches and with the two assessment systems, indicate that 10% sample fraction had better QoS estimations of the actual QoS; however 1% sample fraction results are close to the accepted range. There is no significant difference among the different sampling schemes.

Table 7.8: QoS measurement results based on systematic sampling method using: (a) Fuzzy and (b) Distance evaluation systems.

(a)

	Actual values			Samp e fraction = 0.1%			Samp e fraction = 10%		
QoS	Poor	Average	Good	Poor	Average	Good	Poor	Average	Good
Mean	11.4	51.8	88.2	10.3	49.7	89.3	11.4	50.7	88.1
Std. Dev.	4.9	9.8	5	2.2	8.7	3	5	9.9	5.1

(b)

	Actual values			Samp e fraction = 0.1%			Sample fraction = 10%		
QoS	Poor	Average	Good	Poor	Average	Good	Poor	Average	Good
Mean	11.1	42.3	98.3	11.5	44.5	98.7	11	42.3	98.5
Std. Dev.	7.4	6.1	1.8	6.7	5.8	1.6	8	6.1	1.9

Table 7.9: QoS measurement results based on random sampling method using: (a) Fuzzy and (b) Distance evaluation systems.

(a)

	Actual values			Sampl e fraction = 0.1%			Sample fraction = 10%		
QoS	Poor	Average	Good	Poor	Average	Good	Poor	Average	Good
Mean	11.4	51.8	88.2	10.4	47.3	88.7	11.6	50.9	88.1
Std. Dev.	4.9	9.8	5	2.6	6	4.4	5.2	9.9	5.1

(b)

	Actual			Sample fraction = 0.1%			Sample fraction = 10%		
QoS	Poor	Average	Good	Poor	Average	Good	Poor	Average	Good
Mean	11.1	42.3	98.3	11.8	43.4	98.7	11	42.3	98.5
Std. Dev.	7.4	6.1	1.8	6.7	5.1	1.8	8.1	6.3	1.9

Table 7.10: QoS measurement results based on stratified sampling method using: (a) Fuzzy and (b) Distance evaluation systems.

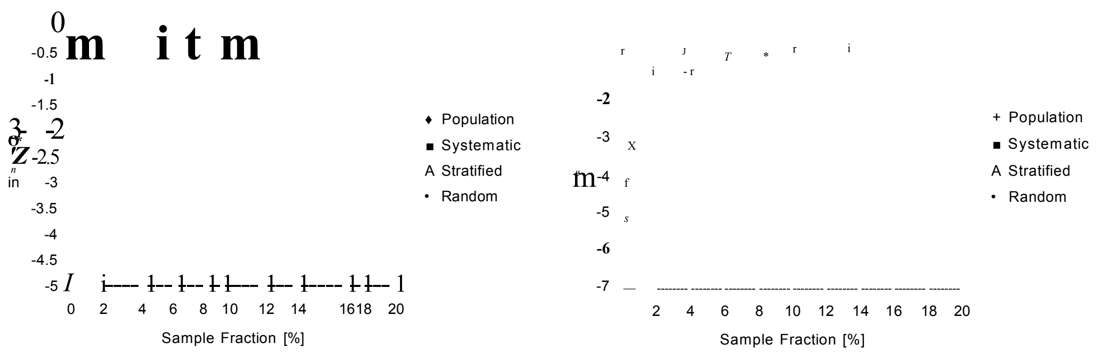
(a)

	Actual values			**Samp e fraction = 0.1%**			**Sample fraction = 10%**		
QoS	Poor	Average	Good	Poor	Average	Good	Poor	Average	Good
Mean	11.4	51.8	88.2	10.7	48.7	88.8	11.4	50.8	88.2
Std. Dev.	4.9	9.8	5	3.1	8.5	3.9	5	9.7	5

(b)

	Actual			**Samp e fraction = 0.1%**			**Sample fraction = 10%**		
QoS	Poor	Average	Good	Poor	Average	Good	Poor	Average	Good
Mean	11.1	42.3	98.3	12.6	43	98.6	11.3	42.4	98.5
Std. Dev.	7.4	6.1	1.8	7.4	5.1	1.9	8.1	6.1	1.9

It can be seen from the above results that sampling is capable of infering the actual QoS. Some users may require a guaranteed QoS, for example, a contract (i.e. SLA) between the user and the service provider to afford a QoS not less than 70%. This value should be guaranteed during the time of providing the service. Sampling schemes can be used to validate this SLA by estimating the percentage of QoS values that violate this value. In addition to this, an exploration of the best sampling scheme that can estimate the actual QoS violation of the contract in terms of bias and precision. Figures 7.32(a) and (b) depict the bias of estimates of the sample versions QoS mean from the actual QoS mean using the two evaluation systems.

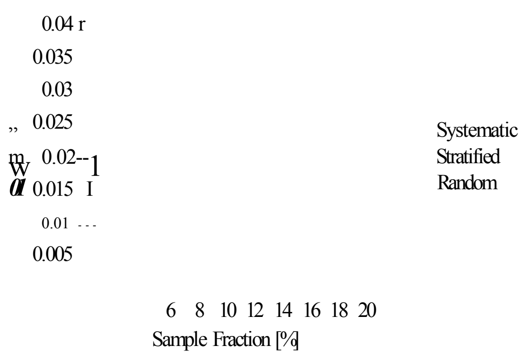


(a) (b)

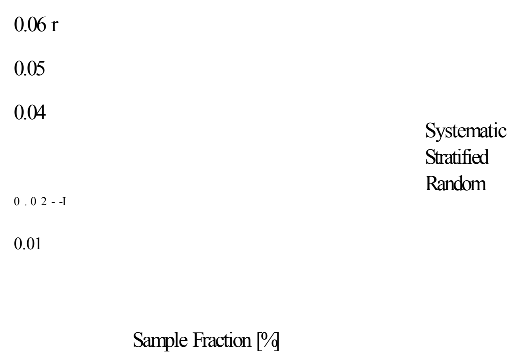
Figure 7.32: Bias of the estimated mean QoS using sampling techniques using: (a) Fuzzy and (b) Distance evaluation systems.

The above figures show the results for all sampling schemes and for different sample fractions. From these figures, the actual QoS mean was 55.3% and 61.8% using the Fuzzy and Distance assessment systems, respectively. These results illustrate that the bias is comparatively small for all schemes, especially large sample fractions. In addition, as expected it rapidly decreases for large sample fractions and becomes close to zero. From these figures, it is clear that all sampling schemes have nearly the same bias compared to each others. Moreover, these bias values are stable and smoothly decreasing. After a sample fraction of 2% the bias became constant and equal to -0.5%.

Figures 7.33(a) and (b) illustrate the precision of the estimates of the sampled mean expressed by calculating the RSE based on the empirical standard deviation of the estimates. In contrast to the bias, the precision results showed discrimination among the sampling schemes. It is apparent that systematic sampling provides the most accurate estimates compared to the stratified and random approaches. In addition, stratified sampling affords more precise estimates than the random sampling scheme. Nevertheless, the errors are of the random sampling and are still quite small.



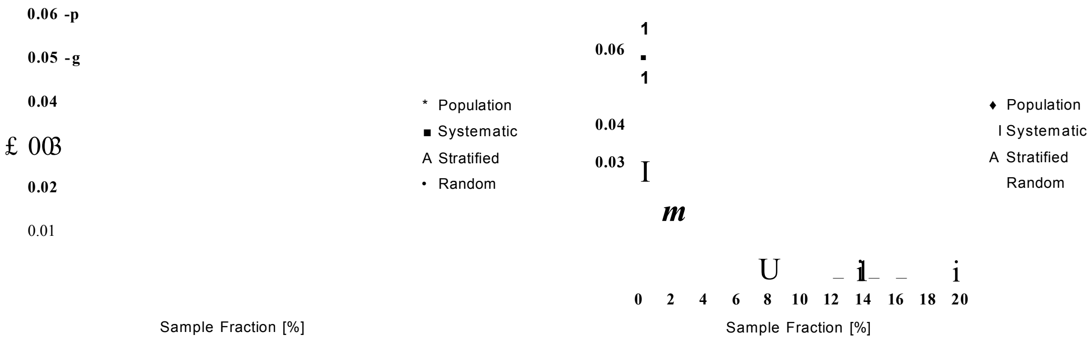
(a)



(b)

Figure 7.33: RSE of the estimated mean QoS using sampling techniques using: (a) Fuzzy and (b) Distance evaluation systems.

An examination of whether a bias is introduced when estimating the ratio of violators of a specific QoS threshold (SLA contract) from estimated values and the correctness that can be accomplished with the different sampling techniques are shown in Figures 7.34(a) and (b) and Figures 7.35(a) and (b), respectively. As an example, a QoS threshold of 70% was used in the testing. This value was selected to represent the threshold which is required to perceive a “minimum good quality” of the videoconferencing application. The actual QoS proportion of violators was 0.49. It is noticeable from figures 7.34(a) and (b) that all the sampling methods provide bias values which are comparatively large with small sample sizes (0.2% and 0.5% samples) and this starts to decline to be less than 0.015 thereafter. Approximately, both assessment systems gave similar degree of biasness. From Figure 7.35(a) and (b), systematic sampling has the minimum RSE (highest precision) for estimating the percentage of violators of the QoS threshold. Although, random sampling showed the worst performance because it presented the highest RSE error compared to other schemes. Even though, the RSE is still relatively small for random sampling.



(a) (b)

Figure 7.34: Bias of the estimated QoS violator proportion from the actual QoS violator proportion using sampling techniques using: (a) Fuzzy and (b) Distance evaluation systems.

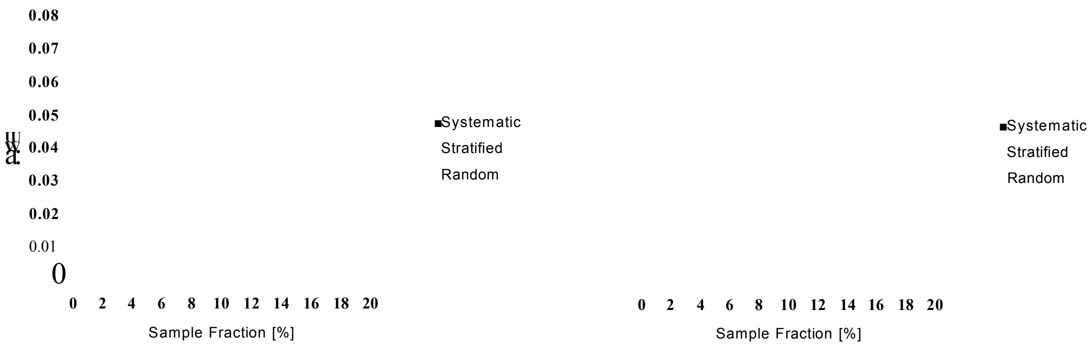


Figure 7.35: RSE of the estimated QoS violator proportion from the actual QoS violator proportion using sampling techniques using: (a) Fuzzy and (b) Distance evaluation systems.

7.5 Summary

This chapter describes a framework of sampling deployment for non-intrusive estimation of QoS parameters and the assessment of the overall QoS of a multimedia traffic over a wireless ad hoc network. This network was subjected to three different traffic load situations; light, moderate and heavy loads to examine the effectiveness of these methods to estimate the network QoS/performance based on the fuzzy and distance evaluation approaches. Experiments were performed with systematic, random, and stratified sampling and for different sampling fractions. Moreover, it has been shown how sampling schemes can be used for the confirmation and validation of the user QoS requests and guarantees (i.e. SLA). Generally, from the obtained results, all sampling methods used confer a satisfactory measure of QoS parameters and the overall

QoS. In the next chapter, a new approach is developed to reduce the biasness and rectify some of the measurements which the ordinary sampling methods could not capture very well.

Estimation of the QoS Using Sampled Passive Measurement

8.1 Introduction

Service providers are required to provide specific levels of service quality in terms of the traffic performance over their networks. Consequently, the traffic performance and QoS must be measured and assessed in a way that reflects the actual traffic performance and QoS. In this chapter, a new performance estimation method is proposed to estimate the actual network QoS and performance experienced by users based on a simple passive sampling measurement method. Furthermore, the sampled performance data are transformed and corrected in a way to accurately represent the actual traffic user performance. This method is based on two previous monitoring approaches. These approaches are discussed in the Section 8.2.

This chapter is structured as follows: Section 8.2 presents the related works. Section 8.3 discusses the concept and the derivation of the proposed approach and Section 8.4 describes the experimental set up. Section 8.5 provides the experimental results obtained from the application of the proposed approach. Lastly, Section 8.6 summarises the chapter.

8.2 Related Work

Recently, many monitoring methods have been developed to achieve the required accuracy level based on active and passive methods. For ATM networks, Lindh (2001) has proposed Operation, Administration, and Maintenance (OAM) cells which are used for fault and performance management. This presents a QoS monitoring approach which combines passive and active measurement methods. The probe packets are sent at regular intervals (per some fixed number of user packets) and measure the network performance. Studies of this mechanism applied to IP networks have been reported by Lindh, (2001) and Lindh, (2002). Passive monitoring is used to count the number of user packets between the probe packets. With this mechanism, the performance statistics obtained by the probe packets to some extent agree with those obtained by the

users because the number of probe packets is proportional to the number of user packets. However, the numbers of probe packets sent by this mechanism grow with the volume of user traffic, so more additional traffic will be injected during congestion periods. As a result, the active probe packets will perturb the network and affect the QoS of the u
ser traffic which will highly deviate the performance from the actual performance without the presence of the probe traffic.

In order to overcome the drawbacks of active and passive measurements, a different approach was proposed by Ishibanishi, et al., (2004), Ishibashi, et al., (2002) and Aida, et al., (2003). This approach suggested a performance measurement method, Change-of-Measure Based Passive/Active Monitoring (CoMPACT monitor), for estimating the actual delay experienced by the users. This method is based on a combination of both active and passive monitoring techniques. This improved precision of estimation of the actual user performance as it was based on simple active measurements transformed by passive monitoring. Later, this way is used for counting the user packets.

The basic procedures of that approach are as follows (Ishibanishi, et al., 2004):

- Measure network performance using active-probe packets; and
- Convert the network performance to actual performance experienced by user packets by weighing the performance with the number of user packets arriving near the probe packets, which is measured passively.

As described so far and from the mentioned characteristics and procedure of monitoring and estimation of the actual user performance, the authors claim that the CoMPACT method has the following advantages (Ishibanishi, et al., 2004), (Ishibashi, et al., 2002) and (Aida, et al., 2003):

- It has a slight effect on the user traffic as the extra traffic of the probe packet is independent of the volume of user traffic and negligible compared to that for the OAM method.
- It enables a reliable estimate of QoS and performance measures because it can estimate the actual performance as perceived by users.
- As information required to infer the user-experienced performance can be obtained from data measured within the period of measurement, the performance data can be obtained in a timely fashion.

- Simplification of the passive measurement which is only required for measuring the amount of traffic (counting the number of packets).

Our approach is different from the procedures followed in the CoMPACT and the OAM methods. In this chapter, a modification in the estimation approaches of these methods, and mainly CoMPACT, will be proposed. This modification will be based mainly on using passive measurement only in the process of estimating the actual user performance. The adopted mechanism of using passive measurement is performed using sampling methods rather than the active sampling mechanism. Using a passive sampling approach introduces the a dvantage of not adding extra traffic to the network and not affecting or perturbing the QoS of actual user's traffic. The basic procedure for the proposed approach is as follows:

(i) Take a number of samples of the ongoing current traffic and measure the network performance based on measuring the QoS parameters (delay, jitter, packet loss and throughput) and then the overall QoS of the sampled packets.

(ii) Convert the sampled version to accurately represent the actual performance experienced by user packets by weighting the performance with the number of packets arriving between the sampled packets, which are counted passively.

Thus, our approach overcomes the disadvantages of both active and passive monitoring schemes. This is represented by selecting samples from the actual traffic and then the estimations based on these samples are corrected by a weight of count of the arriving packets between successive samples. So, the new approach neither disturbs or biases the actual network performance (as in active methods) compared to the OAM and CoMPACT methods nor depends on the whole traffic measurements (as in passive methods).

8.3 Description of the Proposed Method

The CoMPACT method was used by Ishibanishi, et al., (2004), Ishibashi, et al., (2002) and Aida, et al., (2003) to estimate the one-way delay of an application based on active measurement. In our work, this method will be used to estimate QoS parameters based on a combination of passive measurement and sampling techniques. The proposed method is a scalable estimation technique which enables the details of characteristics and performance measures about the user actual traffic behaviour to be obtained during measurement periods.

This section will describe the mathematical framework of the CoMPACT method and how this framework can be extended and modified so it can be used with passive sampling methods rather than the active probing technique.

8.3.1 Estimation of the User QoS

Suppose the network under consideration is shared by K users and let $X_k(n)$ denotes the measurement objective (delay, jitter,... etc) of the *nth* packet of user k. X has the distribution function F. This distribution is given by (Aida, et al., 2003):

$$\begin{aligned} \Pr(X > a) &= \int 1_{\{X>a\}} dF(x) \\ &= E_F\left[1_{\{X>a\}}\right] \end{aligned} \tag{8.1}$$

where a is an arbitrary real number, $E_F[.]$ is the expectation with respect to F and $1_{\{.\}}$ denotes the indicator function which can be written as,

$$1_{\{x\rangle a\}} = \begin{cases} 1 & \text{if } x > a \\ 0 & \text{otherwise} \end{cases} \tag{8.2}$$

If n packets arrive in a measurement period, then $X(i)$ denotes the i^{th} value of X. Then the estimator $Z_X(n,a)$ of the distribution of X, which is like the mean estimator, is given by (Ishibashi, et al., 2002) as:

$$Z_X(n,a) = \frac{1}{n}\sum_{i=1}^{n} 1_{\{X(i)>a\}} \tag{8.3}$$

Suppose a situation occurred in which it is difficult to measure the user traffic directly and an estimator of its distribution cannot be obtained using equation 8.3; which requires capturing all packets to calculate the QoS, like situations in high speed networks. To solve the problem of QoS measurement in these conditions, an approach to estimate the performance based on active probing was proposed by Ishibanishi, et al., (2004), Ishibashi, et al., (2002) and Aida, et al., (2003). In the following, this approach is adapted and applied for performance estimation using sampling methods.

Let $V(t)$ be the network performance at time t and X to be the value of $V(t)$ measured at a certain time; then $V(t_i) = X(t_i)$. Also, let Y be the value of $V(t)$ sampled at a specific time, and let the distribution function of Y be Q. Thus, Y is considered to be the network performance as measured by sampled packets and the distribution of Y is to estimate the distribution of X. Note that there may be some discrepancies between the F and Q

distributions depending on the number of samples used, the type of sampling and sampling times. The distribution of X can be rewritten based on the distribution of Y as follows (Ishibanishi, et al., 2004) (Ishibashi, et al., 2002):

$$\Pr(Y > a) = \int 1_{\{Y>a\}} dQ(y) \cong \Pr(X > a)\,; \text{ then;}$$
$$\Pr(X > a) \cong \int 1_{\{y>a\}} \frac{dF(y)}{dQ(y)} dQ(y) = E_Q\left[1_{\{Y>a\}} \frac{dF(Y)}{dQ(Y)}\right] \tag{8.4}$$

where $E_Q[.]$ denotes the expectation with respect to the Q.

Suppose n user packets are received at the monitoring point and m V samples are taken at different times (s_j). Let $Y(j)$ be the j-th measurement sample at time s_j such that $Y(j) = V(s_j)$, $j=1,2,3...m$. Then an estimator $Z_Y(m,a)$ of $Pr(X>a)$ (as in equation 8.1) to approximate $Z_X(n,a)$ can be expressed using $Y(j)$ as follows (Ishibanishi, et al., 2004) (Ishibashi, et al., 2002):

$$\Pr(X > a) \cong E_Q\left[1_{\{Y>a\}} \frac{dF(Y)}{dQ(Y)}\right] = \frac{1}{m}\sum_{j=1}^{m} 1_{\{Y>a\}} \frac{dF(Y)}{dQ(Y)} = Z_Y(m,a), \text{ So;}$$
$$Z_Y(m,a) = \frac{1}{m}\sum_{i-1}^{m} 1_{\{Y(j)>a\}} L(j)$$
$$\text{where} \quad L(j) = \frac{dF(Y(j))}{dQ(Y(j))} \tag{8.5}$$

$L(j)$ is the ratio between the probabilities of X and Y. It is called the likelihood ratio. If Y is easy to measure (in our approach simply by sampling) and $\frac{dF(Y(j))}{dQ(Y(j))}$ can be derived, then the estimator $Z_X(n,a)$ can be easily found from the Y measurements as shown in the following subsection.

8.3.2 Likelihood Ratio Calculation

Likelihood ratio $L(j)$ can be attained by deploying the passive measurement, in which simply a counter is implemented in the monitoring node to count the number of user packets arriving between every two successive sampled packets. Let $\rho_X(t,\delta)$ be traffic volume (i.e. the number of user packets) arriving in an interval $[t,\ t+\ \delta(t)]$ and let $\rho_Y(t,\delta)$ be the number of measurements (i.e. the number of sampled packets) in the interval $[t,\ t+\ \delta(t)]$.

The δ time values are assumed to be short enough in order not to miss important details and variations of the actual user traffic performance $V(t)$. This assumption provides that a single measurement (sample) of Y in that interval (i.e. $[t,\ t+\ \delta(t)]$) can be interpreted to be equivalent to $\rho_X(t,\delta)/\rho_Y(t,\delta)$ measurements of X. So, $L(j)$ can be rewritten as $L(j, \delta)$ and defined as the ratio between the distributions of the user packets received at a given period to the distribution of the sampled packets in that period. The Likelihood ratio can then be calculated as in (Ishibanishi, et al., 2004) (Ishibashi, et al., 2002):

$$L(j,\delta)=\frac{\rho_X(s_j,\delta)\Big/\sum_{j=1}^{m}\rho_X(s_j,\delta)}{\rho_Y(s_j,\delta)\Big/\sum_{j=1}^{m}\rho_Y(s_j,\delta)} \tag{8.6}$$

ρ_X and ρ_Y represent the number of the user traffic and the sampled packets, respectively, at the given period. Thus the likelihood ratio can be obtained by passive measurement. The distribution of X is estimated as (Ishibanishi, et al., 2004) (Ishibashi, et al., 2002):

As $\sum_{j=1}^{m}\rho_X(s_j) = n$ *and* $\sum_{j=1}^{m}\rho_Y(s_j) = m$; then from equation (8.6);

$$L(j,\delta)=\frac{\rho_X(s_j,\delta)\big/n}{\rho_Y(s_j,\delta)\big/m} \tag{8.7}$$

Substituting equation (8.7) in equation (8.4), Z_y will be;

$$Z_Y(m,a)=\frac{1}{n}\sum_{j=1}^{m}1_{\{Y(j)>a\}}\frac{\rho_X(s_j,\delta)}{\rho_Y(s_j,\delta)} \tag{8.8}$$

Using the above derivation steps of the distribution of X, an estimator of the mean of the user traffic X, $M_Y(m)$ can be obtained. This mean estimator is (Ishibashi, et al., 2002) (Ishibanishi, et al., 2004):

$$M_Y(m)=\frac{1}{n}\sum_{j=1}^{m}Y(j)\frac{\rho_X(s_j,\delta)}{\rho_Y(s_j,\delta)} \tag{8.9}$$

To simplify equations 8.7, 8.8 and 8.9; and as denoted above; X_k is the actual user QoS parameter to be estimated and Y_j is the measured parameter using the sampled packets at s_j, the number of packets for user k arriving in the sampling (measurement) interval $[s_j, s_{j+1}]$ is $\rho_k(j)$, and so the number of total arrived packets for user k is:

$$n_k=\sum_{j=1}^{m}\rho_k(j) \tag{8.10}$$

Moreover, since there is a single Y measurement value during the $[s_j, s_{j+1}]$ period, then $\rho_Y(s_j,\delta)$ is equal to one. Substituting $\rho_Y(s_j,\delta)$ = 1 and equation 8.10 in equation 8.6, the likelihood ratio will be:

$$L_k(j,\delta) = \rho_k(j)\frac{m}{n_k} \tag{8.11}$$

After substituting this into equation 8.7 and 8.8, the estimate of the user parameter distribution and mean based upon the sampled packet are given by (Ishibanishi, et al., 2004) (Ishibashi, et al., 2002):

$$Z_y(k,m,a) = \frac{1}{n_k}\sum_{j=1}^{m} 1_{\{Y(i)>a\}}\,\rho_k(j) \tag{8.12}$$

$$M_Y(m) = \frac{1}{n_k}\sum_{j=1}^{m} Y(J)\rho_k(j) \tag{8.13}$$

As described so far and from equations 8.8 and 8.9, the proposed method passively samples the network/user QoS performance and passively counts the number of the arrived user packets. It is expected that the proposed method will have the following advantages:

- No extra traffic is created for inferring the user performance like active probe traffic; therefore there is no effect on the user traffic and the network performance.
- It provides a reliable estimate of QoS and performance measures because it depends mainly on the actual traffic itself not on active samples, so it can efficiently estimate the actual performance as perceived by users.
- As information, required to infer the user-experienced performance, can be obtained from data measured within the period of measurement, the performance data can be obtained in a timely fashion (Ishibanishi, et al., 2004).

8.3.3 Application of the Method

The CoMPACT method was used to estimate the delay of user traffic only (Ishibanishi, et al., 2004), (Ishibashi, et al., 2002) and (Aida, et al., 2003). The purpose of this work is to devise a measurement and monitoring system that can accurately estimate the main QoS parameters and the overall QoS/performance of the actual user. In the experiments, the estimation approach will be based on deployment of the systematic and random sampling techniques. Detailed information about sampling is provided in Chapter 7.

The ordinary active measurements have been examined to evaluate the overall user QoS. For detailed results of estimated QoS parameters and the overall QoS using active methods, refer to Chapter (6). In addition, in Chapter (7), the traditional sampling schemes have been used also to infer the QoS parameters and the overall QoS of the user traffic.

The contributions of this chapter are three folds: First, the CoMPACT method will be applied to check its efficiency for jitter estimation of the user traffic. In addition, the OAM and the CoMPACT methods will be utilised to estimate the user throughput and packet loss ratio. In order to evaluate the overall QoS, Fuzzy and Distance evaluation systems are applied to assess the overall QoS mean and distribution using the estimated QoS parameters from the CoMPACT and OAM methods. Second, the CoMPACT and OAM schemes will be modified for the QoS monitoring system to be a purely passive monitoring approach based on sampling deployment as a core of the evaluation system. Third, comparison between the extended CoMPACT system and the new proposed system will be carried out.

8.4 Experimental Set up

In order to implement and demonstrate the application of the proposed approach and fulfil the three tasks mentioned in the previous sub-section, NS-2 was used. The network topology and the traffic load characteristics used in the experiments of validation of the proposed approaches are the same topology and traffic characteristics which were used in Chapters 6 and 7. The sampling process has the same procedure which was followed in Chapter 7. The validation of these approaches will be examined for the videoconferencing multimedia application.

8.5 Results and Discussion

In this section, an application of the proposed method is to estimate the end-to-end delay, jitter, packet loss ratio, throughput and finally the overall QoS using the fuzzy logic and the distance evaluation systems. In addition to the proposed approach and to allow for comparisons, these parameters and the overall QoS are also evaluated using the CoMPACT method. The results will include the estimation of the Cumulative Distribution Functions (CDFs), mean and the proportion of violation of specific thresholds for the QoS parameters and the overall QoS.

8.5.1 One-way Delay

One application of the proposed method is to estimate the end-to-end delay between two nodes carrying a videoconferencing application. This delay is estimated using two approaches: probing technique and sampling methods (systematic and random). Figures 7.6(a)-(c) show the actual traffic delay and the delays obtained based on the two sampling approaches. From these Figures, the sampled delay captures the time variance of the actual delay, i.e. it has nearly the same performance and behaviour. Nevertheless, there are some discrepancies between the sampled and the actual delays. In addition, there are some fluctuations the sampled delay could not capture very well, that was due to the fact that the number of sampled packets were small compared with the number of the user traffic packets.

Delay threshold (a) [msec]

(a) (b)

Figure 8.1: Delay distributions of the actual, sampled and estimated delay using systematic sampling with: (a) 1.2% and (b) 5% sample fractions.

Figures 8.1(a) and (b) and 8.2(a) and (b) show the delay distributions of the actual user, the sampled packets using the systematic and random sampling, and an estimation of the user packet delay based on the sampled packet using equation (8.12). The Figures showed, as expected from the instantaneous delay, a discrepancy between the distribution of the actual delay and the sampled versions. Using the proposed sampling method, however, a more accurate estimate of the actual delay distribution is obtained compared with the results obtained using sampling only. Both sampling methods provided a good distribution representation of the actual delay and for the both of the sample fractions sizes.

Delay threshold (a) [msec]

Delay threshold (a) [msec]

(a) (b)

Figure 8.2: Delay distributions of the actual, sampled and estimated delay using random sampling with: (a) 1.2% and (b) 5% sample fractions.

l& ertn fic delay vrihioti jxobe
U6ei tfeflic delay wilh probe
R cbe delay
Eslimzted user defey

1

600 000 1000
Delay threshold (a) [msec]

Delay threshold (a) [msec]

(a) (b)

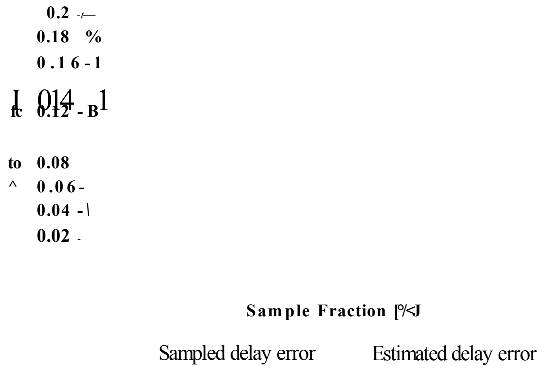
Figure 8.3: Delay distributions of the actual, probe and estimated delay using CoMPACT method with: (a) 4 packets/sec and (b) 1 packet/sec probe rates.

In order to compare the results obtained from our approach with the CoMPACT approach, Figure 8.3 depicts the distributions of the actual delay with and without the presence of probe traffic and the distributions obtained based on the results of active measurement (probe traffic). It can also be seen that this method presented a good distribution estimate of the actual delay from the probe traffic delay measurements. This accuracy is due to the fact that the probe packets are able to capture the time variation of the actual delay. This accuracy increases as the probe rate increases. Flowever, the disadvantage of this method is in its intrusiveness nature which causes a non-negligible amount of extra traffic which can be observed from the resulting the figures. This characteristic causes biasness and discrepancies when comparing the measurement results obtained from the probe traffic and the results obtained from the actual traffic without the presence of the probe traffic. In general, the presence of the probes will deviate and exacerbate the QoS experienced by the users. Increasing the probe rate will increase the amount of this discrepancy as illustrated from Figure 8.3.

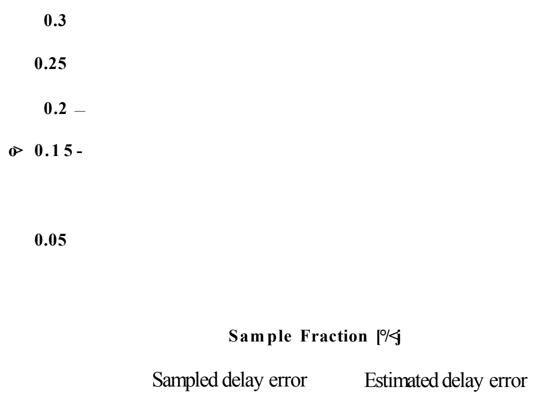
In addition to the distribution estimation of the actual end-to-end delay based on sampling, the proposed approach was used to estimate the actual mean delay based on the sampled versions. Due to the discrepancies between the actual and the sampled delay, there will be some differences between their means. This indicates that the sampled packet delays will bias the estimation of the actual delay which may be overestimated or underestimated. In order to correct the sampled versions to be closer to the actual delay, our method was applied using equation (8.13). The level of correction was checked by calculating the relative error. This includes a comparison between the relative errors obtained from the difference between the means of the actual delay and the sampled versions and the relative error between the actual and the estimated delays. The relative error was calculated as follows:

$$\text{Re}\,lative\ error = \frac{|Sampled\,(or\ Estimated)\ mean\ -\ Actual\ \ mean|}{Actual\ \ mean} \qquad (8.14)$$

The sampling fraction size plays an essential role in the process of estimation. Therefore, the relationship between the accuracy of the estimation of the actual delay and the sampling fraction size (which is equivalent to the length sampling period (i.e. measurement interval)) must be examined. Theoretically, the accuracy of estimation is expected to increase with a larger fraction size. Figures 8.4(a) and (b) show the results of relative error calculations of both systematic and random sampling using several sampling fractions. From these results, our method efficiently reduced the biasness between the sampled versions and the actual delays and produced relative errors which are much lower than those achieved by the simple sampling methods. Furthermore, this Figure confirms that the estimated delay error decays and converges to zero as the number of samples increases. The estimated errors obtained using systematic sampling were less than those obtained from the random method. This means that the systematic method outperformed the random method since it offered relative errors of less than 0.05 while the random sampling errors were less than 0.1. However, these results confirmed that the proposed approach is efficient and more accurate than relying only on the simple sampling methods.



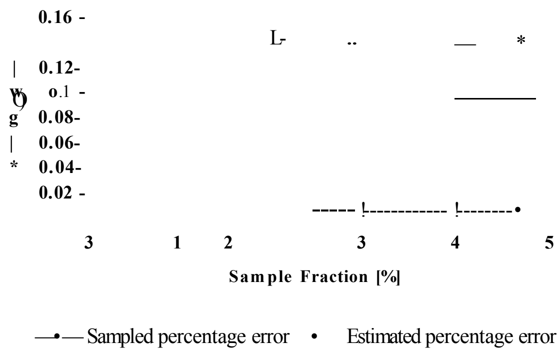
(a)



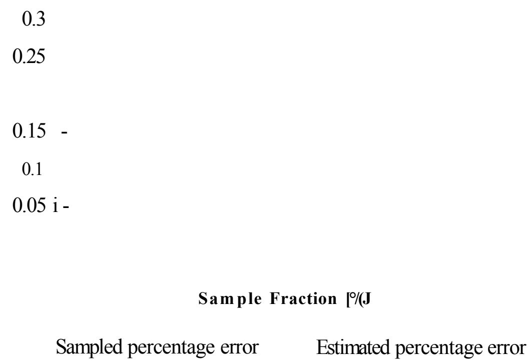
(b)

Figure 8.4: Relative errors between the actual traffic delay and both the sampled and estimated delays using: (a) Systematic and (b) random sampling methods.

Another important application of the proposed method is in the SLA monitoring. The purpose is to check if the packets in a specific flow conform to the guarantees given in an SLA. Generally, the estimation of the long-term mean of a given parameter provides some insights about the service quality provided for an application but it is not sufficient to examine the SLA conformance. This is based on an estimate of the percentage (proportion) of packet's QoS value that violates the SLA contract used (i.e. above a pre-defined threshold (a)). As an example, a packet with delay value less than the threshold is considered conformant, while packets with delay value greater or equal to the threshold are considered violator. After the packets are classified into violators and conformant according to the threshold (a), the percentage of the violators is calculated. This has been done in Chapter 7 using sampling techniques because evaluation of the whole populations is, sometimes, difficult and includes more information than needed. Due to some discrepancies and inaccuracies between the actual and the sampled results of the percentage of the violator packets, these results should be corrected and the discrepancies should be reduced in order to be much closer to the actual results. This correction is achieved using the proposed estimation process.



(a)



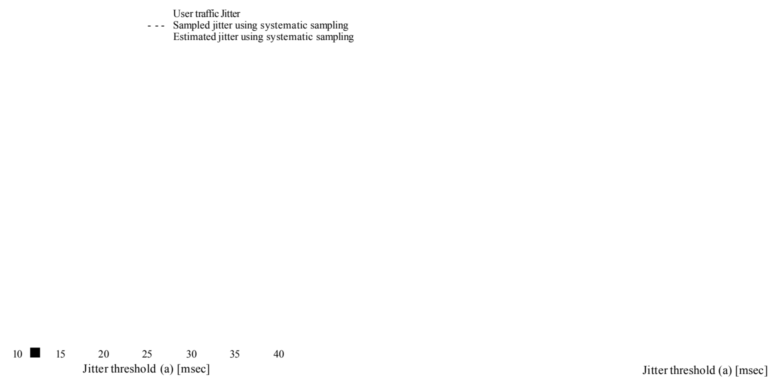
(b)

Figure 8.5: Relative errors between the actual traffic delay SLA violation percentage and both the sampled and estimated delay violation percentages using: (a) systematic and (b) random sampling methods.

Figures 8.5(a) and (b) show a comparison between the results obtained when calculating the relative errors obtained from the difference between the percentage of the actual delay violators and the sampled packet violators and the relative error between the actual and the estimated delay violators using equation (8.14) for the systematic and random sampling approaches. The delay threshold used in these calculations was 400msec. As in the estimation of the mean delay, the estimated percentage relative error reduces and converges to zero as the sample size is increased. Thus, one can use the appropriate sample size to get the estimation of the required accuracy. From these figures, our estimation approach outperformed the simple sampling method because it provided relative errors which are much less than those of the regular sampling approach. Systematic sampling, similar to the mean delay estimation, offered more accurate outcomes than the random scheme, especially for sample fractions, larger than 0.25% in our experiments.

8.5.2 One-way Delay Variation

Another application of the proposed method is to estimate the one-way delay variation (jitter) when videoconferencing between two end-points nodes. This jitter is estimated using the two approaches: CoMPACT and sampling methods (systematic and random). From the results obtained in Chapter 7, there were some differences between the sampled and the actual jitter. In addition, there were some fluctuations the sampled jitter could not capture very well due to small number of sampled packets compared with the number of the user traffic packets. Figures 8.6(a) and (b) and 8.7(a) and (b) depict the jitter distributions of the actual user, the sampled packets using the systematic and random sampling, and an estimation of the user packet jitter based on the sampled packet using equation 8.12. As expected, the discrepancy between the actual jitter and the sampled versions is reflected on the distributions also which means that there are some differences between the actual jitter distributions and the distributions obtained from the sampled packet's jitter. As can be seen from the figures, these discrepancies were reduced using our proposed method. Therefore, highly accurate estimates of the actual jitter were obtained from the results of sampling. Both sampling methods provided a close distribution representation to the actual jitter and for both the sample fraction sizes.



(a) (b)

Figure 8.6: Jitter distributions of the actual, sampled and estimated jitter using systematic sampling with: (a) 1.2% and (b) 5% sample fractions.

User traffic jitter
Sampled jitter using random sampling
Estimated jitter using random sampling

Jitter threshold (a) [msec]

(a) (b)

Figure 8.7: Jitter distributions of the actual, sampled and estimated jitter using systematic sampling with: (a) 1.2% and (b) 5% sample fractions.

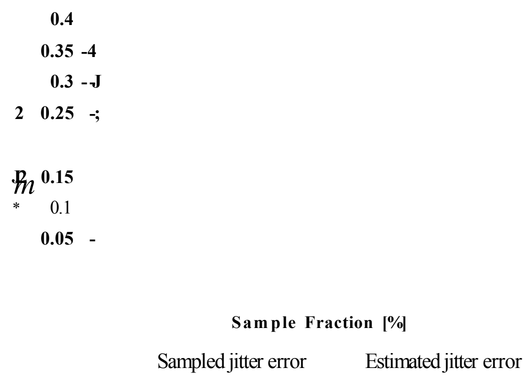
To evaluate and compare the results obtained from our approach, the CoMPACT method which was used for the delay estimation was extended and applied to estimate the actual jitter distribution based on the jitter measurements obtained from the injected probe traffic. Figures 8.8(a) and (b) represent the distributions of the actual jitter with and without the existence of probe traffic and the jitter distribution obtained using the results of active measurements (probe traffic). It can be seen that the CoMPACT method provided an inaccurate and misleading distribution estimate of the actual jitter based on the probe traffic jitter measurements. This result is due to the fact that the probe packets are able to capture the time variation of the actual delay but are not able to sample the actual traffic jitter. The probe jitter is very dependent on the traffic load in the network. In our network, the traffic load had three situations light, medium and heavy. These different traffic loads result in high values of the probe jitter. That is because the more loaded the network, the higher the contention between the nodes. This contention will enforce the nodes to defer their transmissions for some times like SIFS and DCF IFS (SIFS and DIFS). So, these packets were queuing during the busy times of the network channel because it was occupied by some other nodes. The deferral of transmitting some packets will cause some variations in the delays of the consecutive probe packets. A probe packet that goes through a less busy condition may be followed by a high contention period which is met by the next probe which will experience more delay. The extreme difference in delay experienced by these probes will result in a higher jitter. The user traffic does not have this problem as the probe traffic does because the next packet is more then likely to be in the same burst. Therefore the difference in delay between the subsequent user packets is minimal, resulting in a lower jitter for the user traffic. Therefore, the estimated jitter measurement results of the probe traffics are higher than the actual user values. However, increasing the probe rate reduced the difference between the two traffics measurements. This is because increasing the probe rate increases the samples that are in the same network condition which will provide more reasonable results for the probe traffic as shown in Figure 8.8(a) and (b). On the other hand, this will exacerbate the QoS experienced by the users and will increase the amount of the discrepancies when comparing the measurements results obtained from the actual traffic with the presence of the probe traffic and the results obtained from the actual traffic without the presence of the probe traffic. From this discussion, it can be concluded that the CoMPACT approach is unsuitable for jitter estimations. Nevertheless, our approach showed an accurate estimation of actual jitter distribution.

Jitterttreshoid (a) [msec]

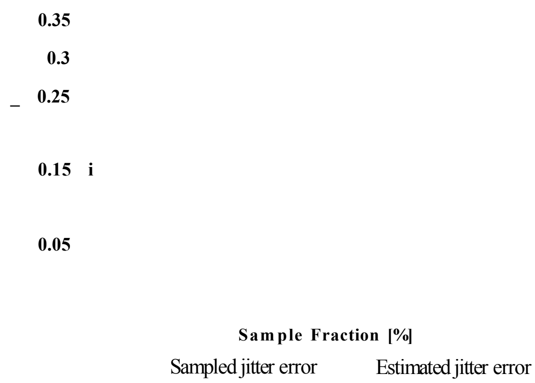
(a) (b)

Figure 8.8: Jitter distributions of the actual, probe and estimated jitter using CoMPACT method with: (a) 4 packets/sec and (b) 1 packet/sec probe rates.

The proposed method was also used to estimate the actual mean jitter based on the sampled versions. Due to selecting only a fraction of the actual traffic, discrepancies between the actual and the sampled jitters will create some differences between their means. Therefore, the sampled packets will deviate the actual jitter mean. In order to reduce the difference between the sampled versions and the actual jitter means, our method was used using equation 8.13. The results of this equation were used to calculate the jitter relative error from equation 8.14. Figure 8.9(a) and (b) depict the calculated relative error between the means of the actual jitter and the sampled versions and the relative error between the actual and the estimated jitters. From these results, our method was capable of reducing the biasness between the sampled versions and the actual jitters. This reveals that increasing the sample size will decrease the difference between the estimated and the actual jitter means. Similar to delay estimation, jitter estimated errors obtained from systematic sampling were less than those obtained from the random method. However, these results confirm that our approach is more accurate than relying only on the simple sampling methods.



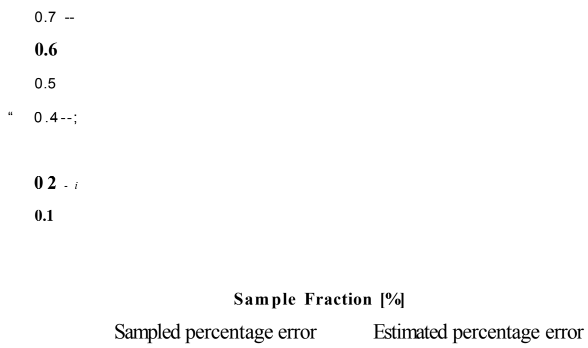
(a)



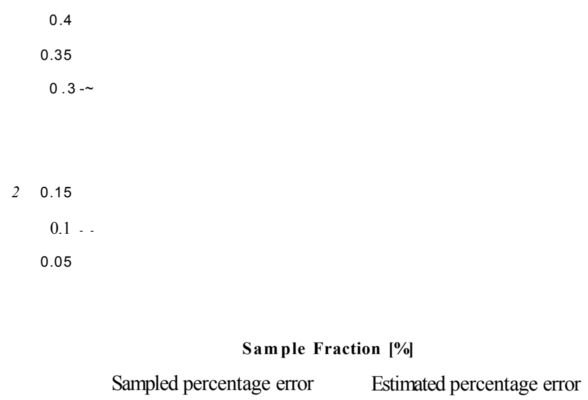
(b)

Figure 8.9: Relative errors between the actual traffic jitter and both the sampled and estimated jitters using: (a) Systematic and (b) random sampling methods.

The method was also applied to monitor the jitter and to investigate whether it conforms to the guarantees given in an SLA. This is based on estimating the proportion of the violator packets from the proportion obtained by the regular systematic and random sampling methods. As an example, the threshold of jitter violation used in these experiments was 20msec. Figure 8.10(a) and (b) show the comparison between the results of calculating the relative errors obtained from the difference between the percentage of the actual jitter violators and the sampled packet violators and the relative error between the actual and the estimated jitter violators using equation 8.14 for systematic and random sampling approaches. As in the estimation of the mean jitter, the estimated percentage relative error lessens and approaches to zero as the sample size increases. Thus, one can get the required accuracy using the suitable sample size. For example, selection of 4% sample fraction will provide jitter estimation measurements of the violation percentage very close to the actual violation percentage. From these figures, our estimation approach provided relative errors which were less than the errors obtained based on the sampling method only. Systematic sampling offered smaller relative errors than those of the random sampling.



(a)

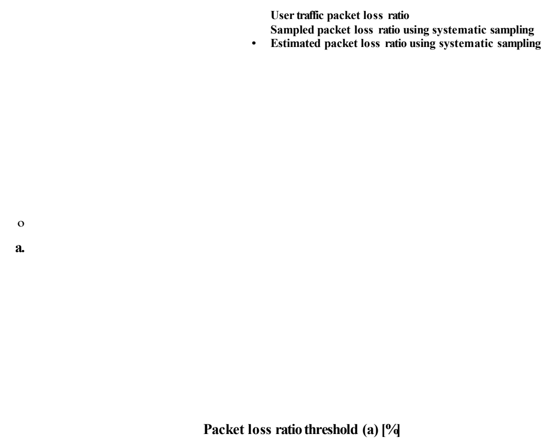


(b)

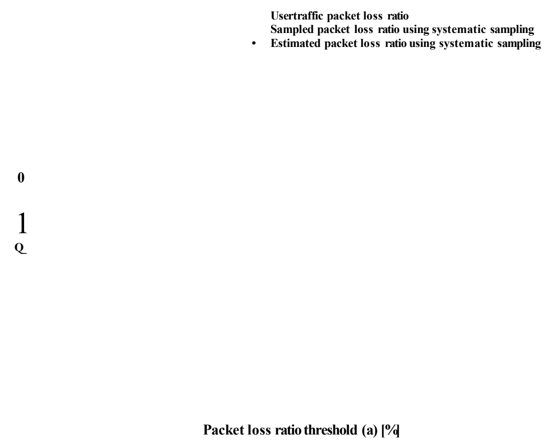
Figure 8.10: Relative errors between the actual traffic jitter SLA violation percentage and both the sampled and estimated jitter violation percentages using: (a) Systematic and (b) random sampling methods.

8.5.3 Packet Loss

The new estimation method was also used to estimate the actual traffic packet loss ratio. The actual traffic packet loss ratio was computed by using the windowing (blocking) technique discussed earlier in Chapters 4 and 5. In these experiments, a window size of 20 packets was used. By counting how many packets were lost and how many packets were sent during the 20 window dependent on the packet ID, the packet loss ratio was calculated. Using the sampling techniques, the packet loss ratio was computed by counting how many packets were lost and how many packets were sent between every two successive samples. After calculating the packet loss ratio, the distributions of these ratios were obtained.



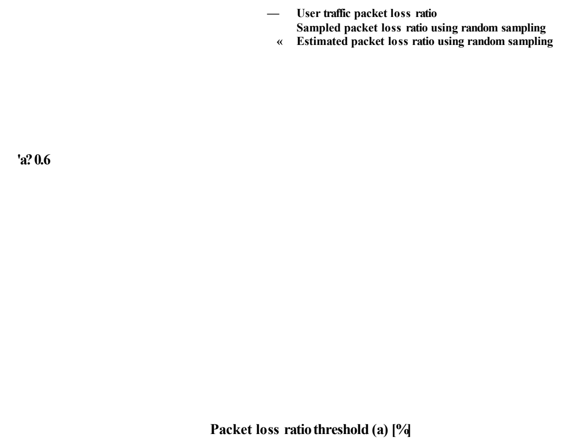
(a)



(b)

Figure 8.11: Packet loss ratio distributions of the actual, sampled and estimated packet loss ratio using systematic sampling with: (a) 5% and (b) 1.2% sample fractions.

Figures 8.11(a) and (b) and 8.12(a) and (b) show the distributions of the actual packet loss ratio, the distribution of packet loss ratio based on sampling and the estimated distribution using the proposed approach with 5% and 1.2% sample fractions for systematic and random methods. From these figures, it can be seen that our method was able to estimate the distribution of the packet loss ratio with a good accuracy. Moreover, it could relieve the difference between the actual and the sampled loss ratio. The estimated distribution is closer to the actual distribution as the sample fraction increases.



(a)

(b)

Figure 8.12: Packet loss ratio distributions of the actual, sampled and estimated packet loss ratio using random sampling with: (a) 5% and (b) 1.2% sample fractions.

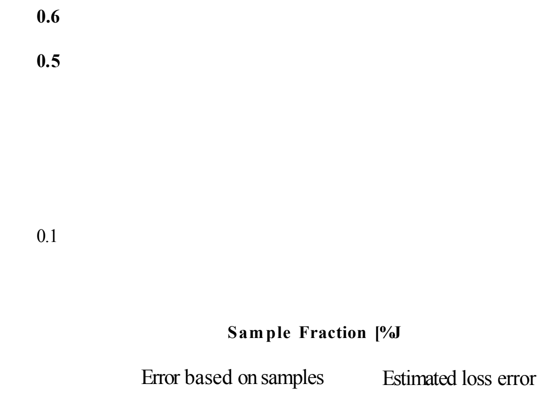
In addition, the CoMPACT method was applied to obtain the packet loss distribution based on the probe packets using the principle of the OAM method. Figure 8.13(a) and (b) illustrate the actual packet loss distribution with and without the presence of the probe traffic and the estimated packet loss ratio based on the probe packets for two different probe rates. These rates were 4 packets/sec and 1 packet/sec. The figures show, firstly, that this method gave a distribution estimate of the actual loss from the probe traffic loss measurements, nearly, analogous to the distribution obtained based on the probe traffic itself. So, CoMPACT could not improve the probe loss distribution. Secondly, due to the intrusiveness nature of the CoMPACT, this caused a non-negligible amount of extra traffic that creates biasness when comparing the measurement results obtained from the actual traffic without the presence of the probe traffic with the results of the actual traffic in the presence of the probe traffic and with the results of the probe traffic itself as shown in Figure 8.13(a) and (b). From this, we can see that our method performed better than the CoMPACT and offered a more accurate packet loss ratio estimation and did not perturb the network by adding extra traffic.

(a)

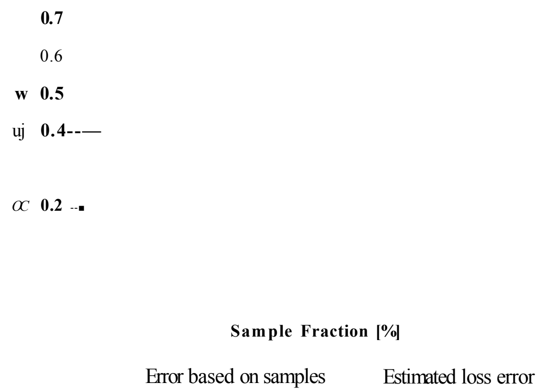
(b)

Figure 8.13: Packet loss ratio distributions of the actual, using the probe and estimated packet loss ratio using CoMPACT method with: (a) 4 packets/sec and (b) 1 packet/sec probe rates.

After calculating the mean packet loss ratio based on sampling schemes and feeding these results to equation 8.13 to get the estimated loss ratio, the relative errors between the actual and the sampled and then between the actual and the estimated were calculated to evaluate the effectiveness of our approach in packet loss ratio measurement. Figures 8.14(a) and (b) demonstrate the result of calculation for these relative errors using systematic and random sampling for different fraction sizes. These figures exhibit that our method performed accurately and provided estimated loss ratio close to the actual loss ratio mean based on the relative error results. This estimated mean approaches the actual mean and the error decays to reach zero as increasing the sample fraction for both sampling techniques. The results obtained using the systematic sampling were closer to the actual loss results than those attained using the random method which is apparent from the resulted error results. Therefore, the sample size is selected depending on the required accuracy (relative error). The higher the sample size is, the more accurate is the result.



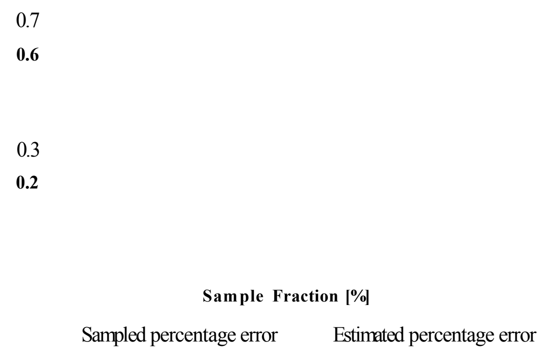
(a)



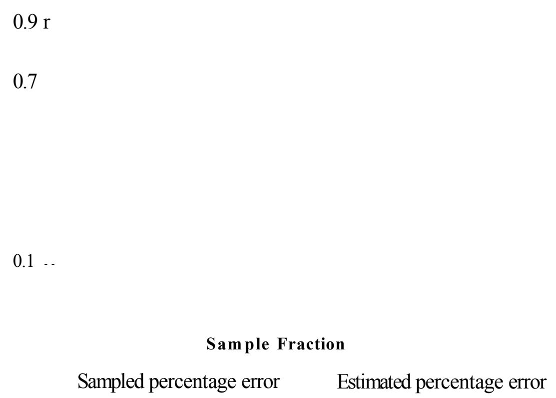
(b)

Figure 8.14: Relative errors between the actual traffic packet loss ratio and both the sampled and estimated packet loss ratios using: (a) Systematic and (b) random sampling methods.

As for delay and jitter, the method was applied to estimate the conformity of the SLA for the packet loss ratio parameter. The packet loss ratio threshold used in these experiments was 2%. Figures 8.15(a) and (b) exemplify a comparison between the results of computing the relative errors obtained from the difference between the percentage of the actual loss ratio violators and the sampled loss ratio violators and the relative error between the actual and the estimated loss violators using equation 8.14 for systematic and random sampling approaches. As in the estimation of the mean delay and jitter, by increasing the sample size, the estimated proportion relative errors reduce and converge to zero. Moreover, from these figures, our estimation approach outperformed the simple sampling method because it provided relative errors that were much less than those of the regular sampling approach even for small sample sizes. Accordingly, depending on the requested estimation accuracy, the appropriate sample size can be selected.



(a)



(b)

Figure 8.15: Relative errors between the actual traffic packet loss ratio SLA violation percentage and both the sampled and estimated packet loss ratio violation percentages using: (a) systematic and (b) random sampling methods.

It can be seen from Figures 8.14(a) and (b) and 8.15(a) and (b), the estimated packet loss ratio relative errors of the mean and the SLA violation exponentially decrease as the sample size increased. In addition, and in most cases, the smallest errors were obtained by the systematic sampling. This may be explained by the stability of the distance (number of packets or time difference) between the any two successive samples in contrast to the random sampling where the distance is variable and depends on the time or the position of the sampled packets.

8.5.4 Throughput

Another application of this method is to estimate the throughput experienced for a specific user. Since the packet size, the number of packets received between any two successive samples and the timestamps for every sample can be recorded at the receiving node, it is possible to calculate the instantaneous throughput. This throughput is calculated using equation (7.2). After calculating the throughput values, a distribution (using the regular simple sampling) and estimated distribution (using our estimation method) for the actual throughput can be obtained using equations 8.3 and 8.12. The actual throughput was calculated using the same windowing technique discussed earlier in packet loss calculation using a window size of 20. The distributions of the actual, sampled and estimated throughput are shown in Figures 8.16(a) and (b) and 8.17(a) and (b). The throughput distribution obtained using the CoMPACT method are depicted in Figure 18 using probe rates of 4 packets/sec and 1 packet/sec.

(a)

(b)

Figure 8.16: Throughput distributions of the actual, sampled and estimated throughput using systematic sampling with: (a) 5% and (b) 1.2% sample fractions.

Form these figures; it is apparent that our estimation method outperformed the CoMPACT approach even for large number of probes. Moreover, the effect of adding the probe traffic on the actual user throughput is clear from the Figures. On the other hand, our method offered a good representation of the actual throughput rather than the regular sampling techniques. Systematic sampling produced a more accurate estimate of the throughput distributions compared with those obtained using the random sampling. The throughput distribution resolution depends on the required accuracy. The larger the sample fraction, the less the discrepancies between the estimated and the actual throughput distributions are, especially for high throughput threshold values (i.e. greater than 350 Kbps in our case).

User traffic throughput
Sampled throughput using systematic sampling
• Estimated throughput using systematic sampling

Throughput threshold (a) [Kbps]

(a)

User traffic throughput
Sampled throughput using systematic sampling
* Estimated throughput using systematic sampling

Throughput threshold (a) [Kbps]

(b)

Figure 8.17: Throughput distributions of the actual, sampled and estimated throughput using random sampling with: (a) 5% and (b) 1.2% sample fractions.

100 150 200 250 300 3GD 400 45D 500 560 000

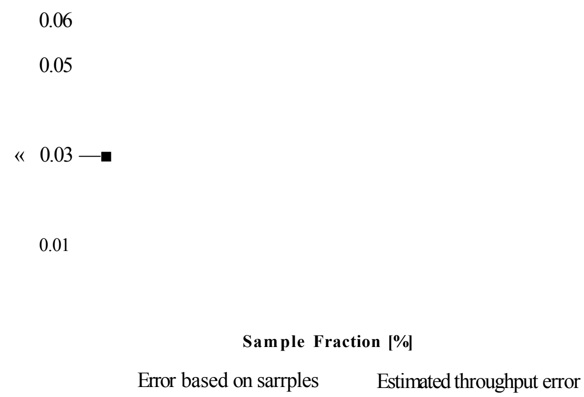
TlroL^ftxt tireshdd (a) /°/$

(a)

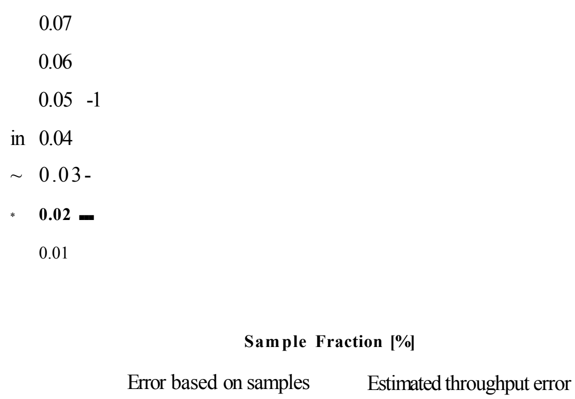
(b)

Figure 8.18: Throughput distributions of the actual, using the probe and estimated throughput using CoMPACT method with: (a) 4 packets/sec and (b) 1 packet/sec probe rates.

In addition to the distribution estimations, the devised approach was also applied to infer the actual throughput mean. After calculating the throughput values using equation 8.15 for the actual and the sampled traffics, the mean of these values was computed by averaging. The relative error between the means of the actual throughput and the sampled one and between the actual and the estimated throughput was then calculated using equation 8.14. The resulted relative error calculations are exposed in Figures 8.19(a) and (b). The results showed that the estimation method provided relative errors lower than the errors provided by the simple sampling. Besides, it can be noticed that the estimated errors are very small for both sampling techniques. This indicates that the estimated throughput mean results were closer to the actual mean especially for the systematic sampling. This is due to the same reason discussed in loss analysis. Systematic sampling has a constant number of packets between the successive sampling as compared with the random method which is variable because of its arbitrariness nature of selecting the samples. The larger the sample size the more accurate the estimation is.



(a)



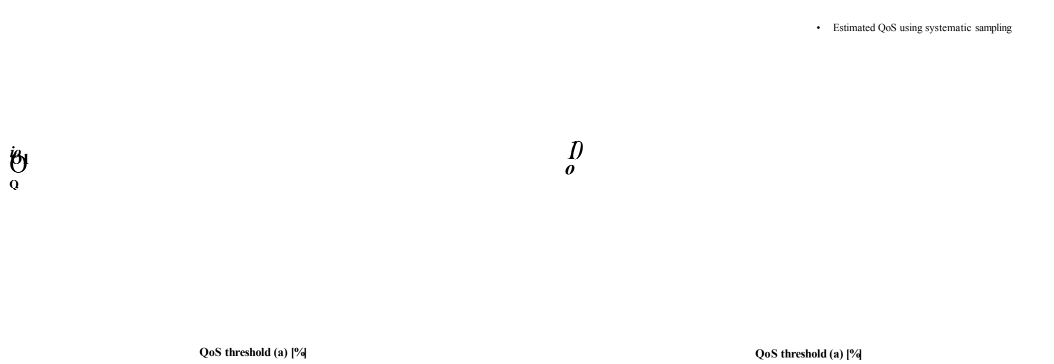
(b)

Figure 8.19: Relative errors between the actual traffic throughput and both the sampled and estimated throughput using: (a) systematic and (b) random sampling methods.

8.5.5 Overall QoS

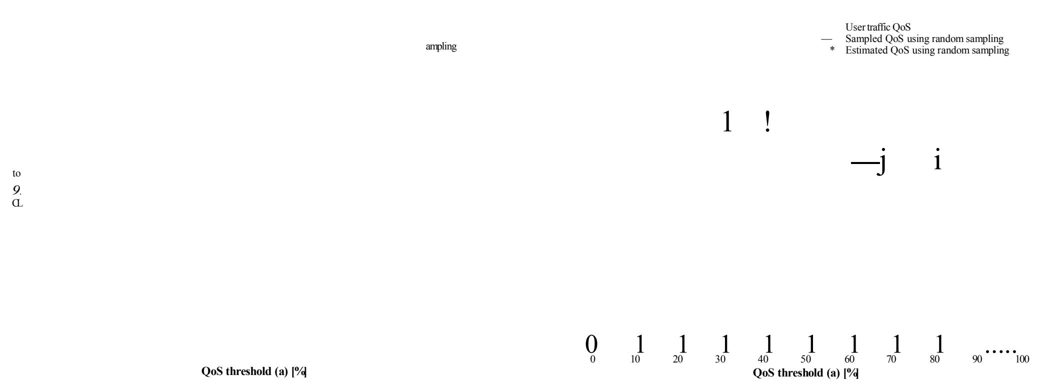
The core of this chapter is to evaluate and assess the overall QoS. This assessment is accomplished using the two evaluation systems, Fuzzy and Distance systems. Based on the obtained results of QoS parameters, it was found that there were some discrepancies between the results of the actual and the sampled versions of the QoS parameters. These discrepancies will also be reflected on the assessed overall QoS using the simple sampling methods. In order to reduce these discrepancies and to correct the sampled overall QoS toward the actual overall QoS, our proposed estimation system was applied using equations 8.12 and 8.13.

Equation 8.12 was used to estimate the distribution of the overall QoS based on the results of the simple sampling methods. Figures 8.20-8.23 demonstrate the application of this equation. This Figures show the actual, sampled and the estimated distributions of the overall QoS assessed by the fuzzy and the distance evaluation systems using systematic and random sampling with two different sampling fractions. As anticipated from the discrepancies between the actual and the sampled QoS parameters, the attained distributions of the overall assessed QoS based on these parameters also have some differences. As exhibited on the graphs, these differences were eliminated using the proposed estimation method. This means that the new method provided an estimation of the overall QoS which is an accurate representation of the actual user QoS using both assessment systems. Nevertheless, systematic sampling resembled the actual distribution better than the random method using the two assessment systems. The larger the sample size, the smaller the difference between the actual and the estimated distributions.



(a) (b)

Figure 8.20: QoS distributions of the actual, sampled and estimated QoS based on systematic sampling using the fuzzy system with: (a) 1.2% and (b) 5% sample fractions.



(a) (b)

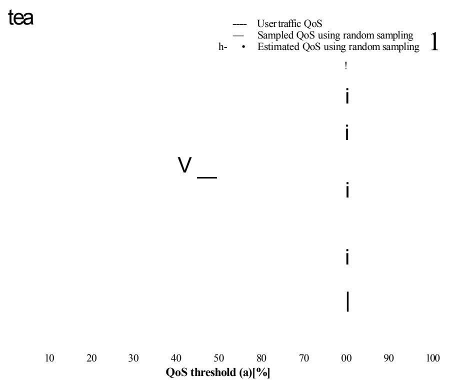
Figure 8.21: QoS distributions of the actual, sampled and estimated QoS based on random sampling using the fuzzy system with: (a) 1.2% and (b) 5% sample fractions.

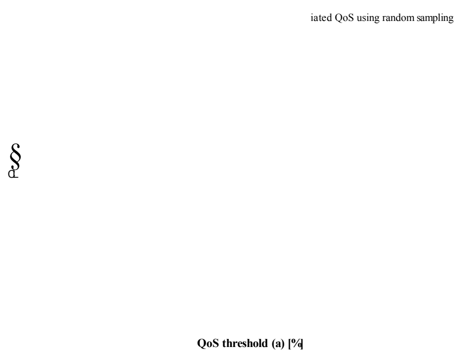
QoS threshold (a)[%]

QoS threshold (a) [%]

(a) (b)

Figure 8.22: QoS distributions of the actual, sampled and estimated QoS based on systematic sampling using the distance system with: (a) 1.2% and (b) 5% sample fractions.

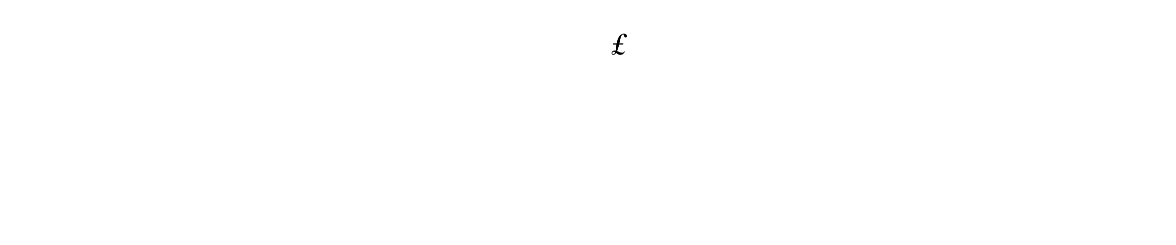




(a) (b)

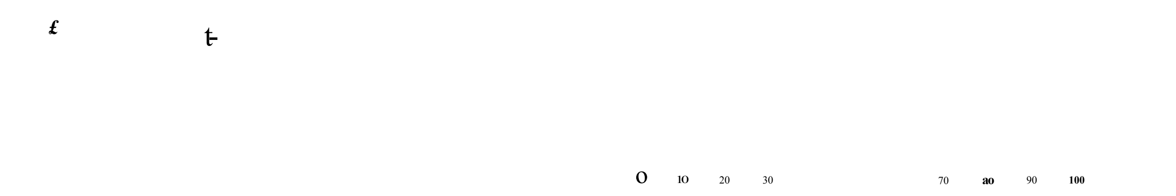
Figure 8.23: QoS distributions of the actual, sampled and estimated QoS based on random sampling using the distance system with: (a) 1.2% and (b) 5% sample fractions.

To validate and compare the results achieved by our proposed method, the CoMPACT approach was also extended and applied to estimate the overall QoS based on the delay, jitter and packet loss ratio measurement results obtained from the injected probe traffic. Figures 8.24(a) and (b) and 8.25(a) and (b) show the overall QoS distribution results of the actual traffic (with and without the presence of the probe traffic) and the probe traffic in addition to the estimated distribution based on the probe measurements using the two assessment methods (i.e. fuzzy and distance). These figures reveal that the CoMPACT method performed poorly when estimating the distribution of the actual overall QoS. In addition, this figure provides important information about the influence of injecting the probe traffic into the network on the overall QoS of the actual traffic as revealed by Figures 8.24 and 8.25.



(a) (b)

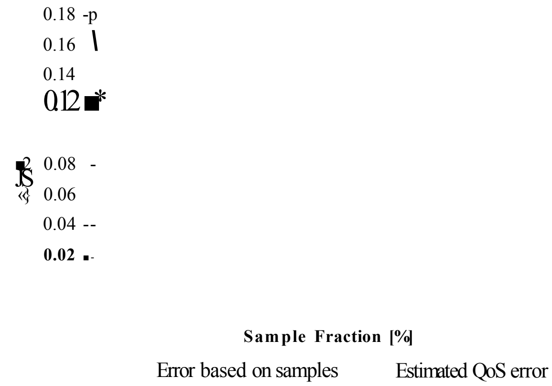
Figure 8.24: QoS distributions of the actual, using the probe and estimated QoS based on CoMPACT method using the Fuzzy system with: (a) 4 packets/sec and (b) 1 packet/sec probe rates.



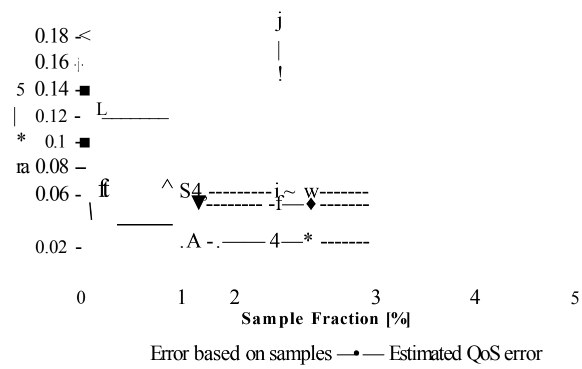
(a) (b)

Figure 8.25: QoS distributions of the actual, using the probe and estimated QoS based on CoMPACT method using the Distance system with: (a) 4 packets/sec and (b) 1 packet/sec probe rates.

It is apparent from the figures that increasing the probe rate will increase the difference between the overall QoS of the actual traffic with and without the existence of the probe traffic. Due to these reasons, the devised estimation system performed better than the CoMPACT one as it did not use the probe traffic in the estimation process which will perturb the actual network performance and user QoS. Furthermore, our approach provided a more powerful estimation compared with the CoMPACT estimation results as stated by Figures 8.21-8.25 and using the two QoS assessment systems.



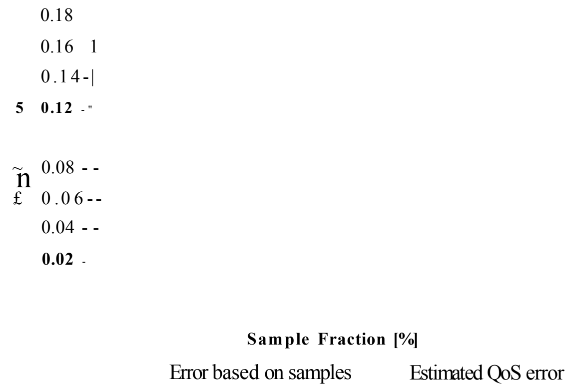
(a)



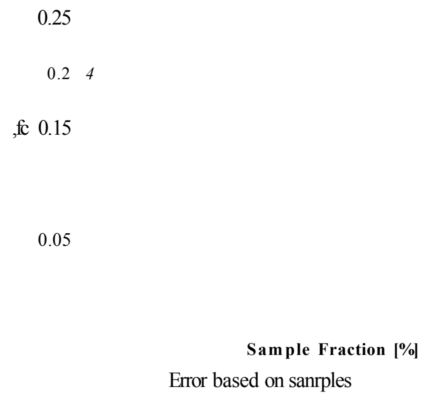
(b)

Figure 8.26: Relative errors between the actual traffic QoS and both the sampled and estimated QoS using Fuzzy system based on: (a) systematic and (b) random sampling methods.

Moreover, the developed method was also applied to approximate the sampled versions overall QoS mean to match the actual traffic overall QoS mean. This approximation was realized using equation 8.13. To study the accuracy of this approximation the results of equation 8.13 were applied to equation 8.14 to obtain the relative errors for the sampled and estimated overall QoS from the actual traffic overall QoS. Figures 8.26(a) and (b) and 8.27(a) and (b) illustrate the calculated relative errors obtained using the two assessment systems for systematic and random sampling with different sampling sizes. From these figures, our method, efficiently, reduced the biasness between the sampled versions and the actual QoS. This indicates that increasing the sample fraction will result in decreasing the difference between the estimated and the actual QoS means. Sample fractions greater than 2% gave overall estimated QoS which is identical to the actual user overall QoS because the relative errors based on these fractions became nearly constant and approximately close to zero as shown in Figures 8.26(a) and (b) and 8.27(a) and (b). Both sampling methods and both QoS evaluation systems afford accurate error results for QoS assessment.



(a)



(b)

Figure 8.27: Relative errors between the actual traffic QoS and both the sampled and estimated QoS using Distance system based on: (a) systematic and (b) random sampling methods.

An additional and important application of the estimation method is the monitoring of the QoS SLA. The purpose of this is to ensure that the assessed QoS complies with the guarantees given in an SLA. The estimation of the overall QoS mean provides some information about the overall service quality provided for an application but it is inadequate to inspect the SLA conformance. The proposed method was used to improve the accuracy and remove the discrepancies between the actual and the sampled results of the proportion of the QoS violation. Figures 8.28(a) and (b) and 8.29(a) and (b) illustrate a comparison between the relative errors obtained from the difference between the percentage of the actual QoS violators and the sampled packet violators and the relative error between the actual and the estimated QoS violators using equation 8.14 for systematic and random sampling approaches. The QoS threshold used was 70%. The estimated percentage relative error converges to zero when the sample size is increased. Furthermore as can be seen from these figures, our estimation approach outperformed the simple sampling methods as it resulted in fewer relative errors compared to the regular sampling approaches. Consequently, the obtained assessment accuracy depends on the sample size used.

0.25

0.2

t 0.15--

0.1 -

0.05

Sample Fraction [%]

Sampled percentage error Estimated percentage error

0.25

0.2 -.

t 0.15-a

0.05 -

Sample Fraction [%]

Sampled percentage error Estimated percentage error

(b)

Figure 8.28: Relative errors between the actual traffic QoS SLA violation percentage and both the sampled and estimated QoS violation percentages using Fuzzy system based on: (a) systematic and (b) random sampling methods.

0.25

0.2 -

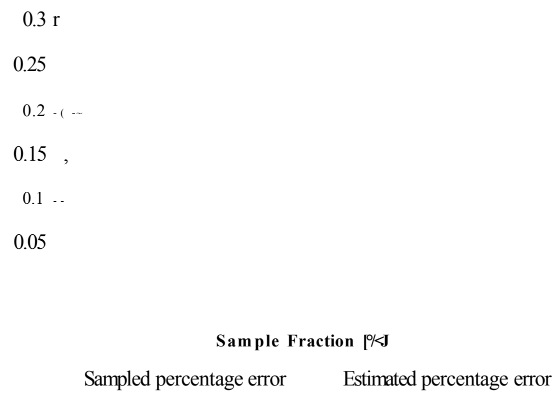
£ 015

0.05

Sample Fraction [%J

Sampled percentage error Estimated percentage error

(a)



(b)

Figure 8.29: Relative errors between the actual traffic QoS SLA violation percentage and both the sampled and estimated QoS violation percentages using Fuzzy system based on: (a) systematic and (b) random sampling methods.

8.6 Summary

This chapter presented a new estimation approach of the actual QoS parameters and the overall QoS. This approach was based purely on the passive monitoring method. The adopted mechanism of using passive measurement is performed based on sampling methods rather than the active sampling mechanism as in the CoMPACT and the OAM methods. So, the new approach neither disturbs nor biases the actual network performance (as in active methods) compared to the OAM and CoMPACT methods. It did not depend on the whole traffic measurements (as in passive methods). Therefore, our approach overcomes the disadvantages of both active and passive monitoring schemes.

The estimation process included the estimation of distribution, mean and SLA violation percentages of the delay, jitter, packet loss ratio, throughput and the overall QoS. Furthermore, the accuracy of the devised approach was tested by calculating the relative error between the actual, the sampled and the estimated using the proposed method. The new method reported smaller errors compared with the normal sampling techniques in representation of the actual user traffic. In addition, it outperformed the CoMPACT method in estimation of the QoS parameters and the overall QoS in terms of accuracy and disturbance or biasness of the actual network performance.

Further discussion of the results obtained from the developed assessment and estimation systems will be presented in the next chapter which will conclude the thesis and provide recommendations for further work.

Discussion, Conclusions and Future Work

9.1 Introduction

The need to obtain and evaluate the QoS of multimedia applications is an essential requirement for technical and commercial reasons. The Mean Opinion Score (MOS) is the most widely used subjective quality measure and the globally acknowledged metric that is recommended by the ITU (ITU, 1996a). The fundamental problems for subjective MOS measurement are that it is costly, time-consuming, and cannot be used for long term and large scale monitoring in any network infrastructure. These have introduced the objective schemes to meet the demands of quality measurement and monitoring in computer and communications networks.

Objective assessment of quality can be intrusive or non-intrusive. Intrusive methods are powerful and accurate, but sometimes they are not very suitable because of the requirement for a reference data. On the other hand, non-intrusive methods are suitable for monitoring the quality directly from IP network and/or non-network parameters. However, current non-intrusive techniques (e.g. statistical E-model or neural network models) depend on subjective tests to obtain the model parameters or to generate the training sets. Unfortunately, due to the subjective tests drawbacks, these models have restrictions and can not face all the possible scenarios in dynamic networks, like wireless networks.

The major objectives of this thesis are two fold: (i) to undertake a fundamental investigation to quantify the effect of the QoS parameters on the perceived overall QoS in wireless networks, (ii) to apply the results to develop efficient mechanisms for intrusive and non-intrusive QoS measurement and estimation for audio and video applications.

This chapter is divided into two sections. The first section presents the conclusions of this thesis. The second section highlights future research directions.

9.2 Discussions and Conclusions

The reality that wireless ad hoc networks are significantly different in size, QoS needs, power availability, and processing capabilities leads to the conclusion that these networks need to be further studied. This study investigated some fundamental aspects and devised efficient QoS assessment and measurement techniques for multimedia traffic over ad hoc networks. These techniques are categorised into direct measurement and indirect measurement (estimation) of the application QoS. In the direct measurement (Chapter 5), the QoS was continuously evaluated based on all packets of the application traffic (i.e., passive). On the other hand, the indirect measurement (Chapters 6, 7 and 8) was based on estimating or inferring the application actual QoS depending on QoS parameters obtained from other traffics, i.e. artificial workload (probe traffic), or using samples from the original traffic itself.

Chapter 5 of this thesis focused on the deployment of intelligent and non-intelligent methods to assess the QoS of multimedia traffic over wireless ad hoc networks. A fuzzy logic, in addition to distance measure systems were developed to evaluate the QoS of audio and videoconferencing multimedia applications. These techniques showed how the QoS parameters could be combined r to produce the output QoS without the necessity for analytical models. In addition, these methods have the advantage that they were a source-to-destination evaluation process without the need of intermediate node cooperation in terms of processing demand. The two proposed assessment systems provided results, which were to some extent close to each other with small differences. These differences were due to the procedures followed by each method. The distance system provided a larger range and generally produced higher output QoS values than the fuzzy system. That was due to the fact that fuzzy system is intelligent and is governed by membership functions, which may provide smooth transitions between the system states. On the other hand, the distance evaluation system is a non-intelligent approach, which mainly depends on the difference between the measured parameters values and the required thresholds and then combining (adding) the differences that produce direct crisp values without any fuzzification.

After grouping the measured QoS of each application into three regions (poor, average and good) and based on the proposed assessment systems, it was easy to quantify how much each application QoS was poor, average and good, which provided a picture about the level of the sustained overall QoS. Furthermore, the obtained QoS distributions provided a good estimation of the QoS at point of comparison in the range of 0 to 100 percentages. This can estimate not only the actual performance and QoS of the individual applications but also the mixed applications overall QoS. The use of the normalisation technique in the calculation of the overall QoS to represent the overall performance of the network gave better results compared with the averaging methods because it reduced the variations and took into account the real values of the QoS of each application. It was also observed that the developed systems were very useful in the measurement of the capacity of the wireless network. From the simulation results, it can be concluded that the standard 802.11 DCF (i.e., channel bandwidth 2 Mbps) can support only 7 simultaneous audio (4 Good QoS and 3 Average QoS) and three videoconferencing sources. This is due to the stringent jitter requirement of the audio application and stringent loss requirement of the videoconferencing.

The above developed assessment systems were used in the process of estimating the overall QoS. Experiments were performed using probing technique with different probe rates and sampling methodologies with different sampling fractions. The simulated network was subjected to three different load situations; light, moderate and heavy loads to examine the effectiveness of these methods to estimate the network QoS/performance. As an example, the simulations were carried out using videoconferencing applications.

In Chapter 6, a new approach for the monitoring of the actual traffic QoS parameters and the overall QoS has been developed. This approach was based on a combination of active and passive schemes. Using the proposed system, delay and delay variation, overall average losses as well as lossy and loss-free periods, throughput and finally the overall QoS have been estimated. The size of the monitoring block played a crucial role in the process of estimation in terms of precision and the level of resolution of the estimated results. The obtained results showed that employing too few monitoring packets resulted in inaccuracies in measurements and caused poor assessments and decisions. On the other hand, too many monitoring packets impacted the network and biased the measurements. The results demonstrated that the resolution of the delay and the delay variation depends mainly on the number of monitoring packets (samples). On the contrary, the loss ratio and the throughput measurements depend on how many user data packets were received between the monitoring packets. Furthermore, increasing the monitoring packets transmission rate resulted in increasing the precision of the estimated QoS parameters and the assessed overall QoS however it did introduced more disturbances to the network performance. From the results, it can be concluded that this method offered a good estimation for the delay, throughput, packet losses and QoS when using different probe rates. Nevertheless, this technique demonstrated some limitations in the delay variation estimation which will be directly reflected on the evaluated QoS.

In order to overcome some of the drawbacks of the probing approach: precision and intrusiveness, another estimation mechanism was devised in Chapter 7. This approach was based on the sampling techniques deployment for non-intrusive estimation of QoS parameters and overall QoS. This method has the advantage of not adding an extra load to the network as in the active methods. In addition, it is not like the passive measurement, which requires the transfer, comparison and calculations for the whole captured data. Generally, from the obtained results, the analysed sampling methods conferred a satisfactory measure of QoS parameters and the overall QoS in terms of average, standard deviation, maximum, minimum values, calculating the degree of significance between the actual population and the sampled versions, Standard Error (SE) and Confidence Interval Length (CIL). Moreover, it has been shown how sampling schemes can be used for the confirmation and validation of the user QoS requests and guarantees (i.e. SLA). All sampling techniques produced no statistical significant difference for different sample sizes based on results obtained using the t-test. In addition to that, it was obvious that all sampling schemes produced an adequate estimation of the histogram distributions of the QoS parameters. Furthermore, the three sampling approaches presented reasonable Cumulative Distribution Function (CDF) estimations of the actual QoS CDF. The degree accuracy of the actual traffic representation is limited by the sample size. Larger sample sizes provide improved accuracy.

Sampling was also explored for the validation of QoS parameters and the overall QoS of SLA contracts in terms of biasness and precision. Based on the produced results, systematic sampling provided the best performance in terms of biasness and precision among the sampling methods. This may be due to the fact that there was nearly no influence of periodicity between the subsequent sampled values and so the correlation between them was low. Therefore, if there was any periodicity between the samples, there will be very high bias values and the precision will be very low for the same sample size. Furthermore, stratified sampling performed better than random. This is due to the nature of its sample selection process which is based on stratification of the parent population; in contrast random sampling is mainly based on a random selection process. A detailed discussion of these follows. In our experiments, the network was subjected to three different load conditions and the measurements were executed over the whole simulation time. The sampling starting point plays an important role in the samples selection process. Systematic sampling selects the starting points according to an already known deterministic function which is controllable, whereas, random sampling selects the starting point in accordance to a non-controllable random process. This means that systematic samples, because of their periodicity characteristic, can cover the whole measurement interval while the random samples, due to randomness, may cover portions of the measurement period. Because of the different network situations, the generated random numbers may not sample some of these situations or the number of samples taken from one situation is very small or very large compared to the samples from other network situations which will bias the obtained results. However, systematic sampling will obtain samples from different positions depending upon starting point, periodicity and measurement interval. On the other hand, in these situations, stratified sampling was better than random and sometimes better than systematic sampling because the entire population was considered in the stratification process and the sample was randomly selected from every stratum. Stratified sampling was superior to systematic sampling because in systematic, sometimes, the packets being sampled exhibited some periodicity.

Form the findings of Chapter 7, simple sampling methods performed well in estimating the actual user QoS. Furthermore, the sampling techniques showed different levels of accuracy for the same sample fraction. Sometimes, large sample sizes are required to achieve a certain level of accuracy. In order to reduce the sample sizes and lessen the degree of biasness between the actual and the sampled QoS, another performance measurement approach for assessing the actual user performance was proposed in Chapter 8. This approach was purely based on the passive monitoring method. It required simple passive counting of the number of packets and simple measurement of traffic performance based on sampling techniques. This included the estimation of distribution, mean and SLA violation percentages. The result of applying this method indicated that it has the advantage of not adding an extra load to the network like the

Change-of-Measure Based Passive/Active Monitoring (CoMAPCT) one. Moreover, it was obvious that the devised technique produced efficient estimations of QoS parameters and the overall QoS which were closer to the actual estimations compared to the estimations obtained by the CoMAPCT technique and the standard sampling methods.

The accuracy of this method was also tested by calculating the relative error between the actual, the sampled and the estimated means using the proposed method. It was found that the accuracy of the estimated mean and the percentage of the SLA violators depended on the number of samples selected. In addition, and even for small sampling fractions, the developed approach provided acceptable estimations of the actual distributions, means and SLA violation proportions of the assessed QoS parameters and the overall QoS. Systematic sampling, from the obtained results and at least for the scenarios used in these experiments, provided a more accurate estimation than the random method.

From this study, and regardless of the network topology and traffic conditions used, the main benefit drawn is that the designed systems provide a valuable assessment and estimation of the application QoS and the network performance in terms of overall application QoS. It can also be concluded that the measured QoS was a good indication of the network conditions and resource availability; since, for example, poor QoS is a reflection of inadequate resources vacant to support the application QoS. Besides, it can be deduced that the output QoS value can be used to monitor the wireless channel to be kept from reaching the congestion point; as loss, delay and then jitter increase rapidly once this point is reached which will deteriorate the performance of the network represented by the measured QoS.

Based on the above conclusions and discussion of the proposed QoS assessment and estimation approaches, overall conclusions may be outlined. Firstly, according to both evaluation systems (i.e., fuzzy and distance), they exhibited comparable outcomes, but in terms of scalability, simplicity, processing, and output range; the distance measure technique outperformed the fuzzy system. Secondly, regarding the devised QoS estimation mechanisms:

- Active probing technique showed some inaccuracies in the obtained measurements and some disturbance to the QoS/performance of the network due to a shortage of the resources availability like bandwidth. Due to these limitations; solutions must be proposed to overcome these disadvantages to provide suitable methods for QoS measurement in wireless networks.

- Passive simple sampling technique provided good QoS estimations but it often suffered from biasness and requirements of large sample fractions.
- Estimation technique which was presented in Chapter 8 granted powerful estimation results based on the conversion process applied to the ordinary sampling methods.

Based on the above comparisons, the final conclusion that can be deduced is that the modified (corrected) sampling estimation technique (Chapter 8) and using the distance measure approach provides the best solution for the problem of QoS a ssessment and estimation.

9.3 Future Work

QoS measurement is still a growing research area. Many open problems are waiting to be investigated and addressed. Based on the research done in this thesis, some future research directions are suggested.

1. **Real network validation**

   The thesis has presented new methods for assessing and estimating the QoS of two main network applications. Although execution and validation of the work have been carried out via simulations, real network scale validations are still needed.

2. **Correct jitter estimation**

   Throughout investigating and developing the QoS estimation approach based on probing technique, the jitter was the main parameter which caused the inaccuracy in results compared to the actual user QoS. Devising new methods to correct the measured jitter to be comparable to the actual user jitter is very important to estimate the actual user QoS correctly.

3. **Generalizing the developed approaches**

   The results presented in this work are for small to moderate size networks. Generalizing the developed approaches to serve large networks would be beneficial. This may include developing policies to handle large MANETs QoS monitoring.

4. QoS measurement for other multimedia services

The approaches presented in this thesis for measuring the QoS of some multimedia applications are generic. They can be easily applied to other applications like media streaming (i.e., audio and video) taking into account every application's specific QoS requirements.

5. QoS assessment over other packet networks

Although the thesis has focused on wireless ad hoc networks (mainly best-effort IP networks), the approach of the QoS measurement can be applied to Internet (best-effort) and to managed IP networks (e.g. DiffServ). The proposed measurement approaches are suitable for any network, as both fuzzy and distance approaches, probing, and sampling techniques are based on a comparison of the reference QoS requirements and the measured QoS parameters of the application transmitted through the network. An important requirement for applying these methods is to understand and obtain the relevant parameters which affect the corresponding application's QoS. These parameters or the range of the values of these parameters are application, user and network dependent. For managed networks, the network performance and the measured QoS will differ from that of the best-effort mode. For example, the range of packet loss ratio or delay may be much smaller than that from the best effort networks for certain QoS classes.

6. QoS performance optimisation and control

The measured QoS can be used to optimise the received quality of the multimedia services along with the changing network conditions and to control the QoS and manage the utilisation of the network available resources, especially ad hoc networks. The overall QoS measure is better than the traditional use of only individual parameters (e.g. delay, packet loss and jitter) as it may provide a direct link to the end user's point of view. The measured QoS control or optimisation can have variety of possible applications. The following are some examples.

(i) Building an intelligent CAC algorithm. The value of the QoS will be forwarded to be used in a CAC algorithm. The CAC algorithm will determine if any new traffic will be admitted in the network or not depending on many factors. These factors include the measured QoS value, QoS requirements of the new traffic, and the state of the network. Therefore, the CAC algorithm problem addresses the issues of finding the suitable network conditions so that the QoS requirements are satisfied.

(ii) Developing routing algorithms based on the measured overall QoS to select the optimum route which can satisfy the required end-to-end QoS for every specific application.

(iii) For media streaming, the assessed QoS can be used for server selection. For example, it can be used to search for an audio/video server which can provide an optimum end-to-end audio/video QoS, instead of traditionally obtaining optimum individual network parameters (e.g. minimum end-to-end delay, jitter or packet loss).

Finally, the assessed QoS can play an essential role in optimising the quality of the multimedia services along with the changing network conditions and control the use of the network resources. Overall, the outcomes and findings of this thesis contribute to the techniques for drawing a realistic picture of the wireless multimedia networks QoS and provide a firm basis and useful insights on how to effectively design future QoS solutions.

References

Aad, I. (2002). *Quality of Service in Wireless Local Area Networks*. PhD Thesis, Joseph Fourier De Grenoble University, France.

Aboelela, E. (1998). *Fuzzy Logic Applications for Routing and Management in Communication Networks*. PhD Thesis. Electrical and Computer Engineering Dept., University of Miami, Fl, USA.

Afifi, H., Cervantes-Pérez, F., Mohamed, S. and Rubino, G. (2001). Real-time Video Quality Assessment in Packet Networks: A Neural Network Model. *International Conference on Parallel and Distributed Processing Techniques and Applications (PDPTA)*. Las Vegas, USA.

Agarwal, A. (2000). Quality of Service (QoS) in the New Public Network Architecture. *IEEE Canadian Review* 36, 22-25.

Aida, M., Ishibashi, K., and Kanazawa, T. (2002). CoMPACT-Monitor: Change-of-Measure Based on Passive/Active Monitoring-Weighted Active Sampling Scheme to Infer QoS. *In Proceedings of the 2002 Symposium on Application and Internet (SAINT'02w), IEEE Computer Society.* Nara City, Japan.

Aida, M., Miyoshi, N., and Ishibashi, K. (2003). A Scalable and Lightweight QoS Monitoring Technique Combining Passive and Active Approaches: On the Mathematical Formulation of CoMPACT Monitor. *INFOCOM 2003*, San Franciso, CA, USA.

Alkahtani, A., Woodward, M., and Al-Begain, K. (2003). An Overview of Quality of Service (QoS) and QoS Routing in Communication Networks. *In Proceedings of 4th Annual Postgraduate Symposium on the Convergence of Telecommunications, Networking and Broadcasting (PGNet)*. Liverpool, UK.

Almes, G., Kalidindi, S., and Zekauskas, M. (1999). A One-Way Delay Metric for IPPM. *RFC 2679.*

Almes, G., Kalidindi, S., and Zekauskas, M. (1999). A One-Way Packet Loss Metric for IPPM. *RFC 2680.*

Anastasi, G., and Lenzini, L. (2000). QoS Provided by the IEEE 802.11 Wireless LAN to Advanced Data Applications: a Simulation Analysis. *Wireless Networks* **6**(2), 99-108.

Anjum, F., Elaoud, M., Famolari, D., Ghosh, A., Vaidyanathan, R., Dutta, A., Agrawal, P., Kodama, T., and Katsube, Y. (2003). Voice Performance in WLAN Networks - An Experimental Study. *In Proceedings of IEEE GLOBECOM*, **6**, 3504-3508. San Francisco, USA.

Asgari, A., Trimintzios, P., Irons, M., Egan, R. and Pavlou, G. (2003). Building Quality-of-Service Monitoring Systems for Traffic Engineering and Service Management. *International Journal of Network Management*, **11**(4), 399-426.

Assessing VoIP Call Quality Using the E-model. (2005). [online]. White paper. Last accessed on 15 July 2006 at URL: http://www.ixiacom.com/pdfs/library/white_papers/voip_quality.pdf.

Ayyash, M. (2005). *A New Robust Quality of Service Routing Protocol for Wireless Mobile Ad Hoc Networks*. PhD Thesis. Illinois Institute of Technology, Illinois, USA.

Ayyash, M., Alzoubi, K., Alsbou, Y. (2006). Preemptive Quality of Service Infrastructure for Wireless Mobile Ad Hoc Networks. *In Proceedings of the 2006 International Conference on Communications and Mobile Computing (IWCMC '06)*, 707-712. Vancouver, British Columbia, Canada.

Batchelor, B. (1978). *Pattern Recognition - Ideas in Practice*. Plenum Press, New York, London.

Bianchi, G. (2000). Performance Analysis of the IEEE 802.11 Distributed Coordination Function. *IEEE Journal on selected areas in communications* **18** (3), 535–547.

Bianchi, G., Capone, A., and Petrioli, C. (2000). Throughput Analysis of End-to-End Measurement-Based Admission Control in IP. *In Proceedings of IEEE INFOCOM 2000*. Tel Aviv, Israel.

Bohdanowicz, A. and Weber, J. (2002). Conceptual Framework of Integrated Variance Reduction Techniques for Evaluation of Communication Systems. *In Proceedings of IASTED International Conference on Communications and Computer Networks*, CCN 2002, 335-341. Cambridge, USA.

Bolot, C. (1993). Characterizing End-to-End Packet Delay and Loss in the Internet. *Journal of High Speed Networks,* **2**(3), 305-323.

Braun, T. (2004). End-to-End Mechanisms for QoS Support in Wireless Networks. *COST 279 Mid-Term Workshop*. Rome, Italy.

Brekne, T., Clementsn, M., Heegaard, P., Ingvaldsen, T. and Viken, B. (2002). State of the Art in Performance Monitoring and Measurements. Fornebu: Telenor Forskning og Utyikling. *R&D Report R 15/2002.*

Breslau, L. and Shenker, S. (1998). Best-Effort versus Reservations: A Simple Comparative Analysis. *ACM SIGCOMM Computer Communication Review, In Proceedings of the ACM SIGCOMM'98 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication,* **28**(4), 3-16. Vancouver, British Columbia, Canada.

Brian, C., Indra W., Jeong G, and Sakai, T. (1997). IEEE 802.11 Wireless Local Area Networks. *IEEE Communications Magazine*, **35**(9), 116-126.

Brownlee, N. (2000). Packet Matching for NeTraMet Distributions. [online]. *Presentation at RTFM Get-Together*, IETF Adelaide. Last accessed on 22 August 2006 at URL: http://www.auckland.ac.nz/net/Internet/rtfm/meetings/.

Brownlee, N., Mills, C., and Ruth, G. (1999). Traffic Flow Measurement: Architecture. *Internet RFC 2722.*

CAIDA Measurement Tool Taxonomy. (2006). [online]. Last accessed on 18 February 2006 at URL: http://www.caida.org.

Calyam P., Sridharan M., Mandrawa W., and Schopis P. (2004). Performance Measurement and Analysis of H.323 Traffic. *In Proceedings of the 5th International Workshop on Passive and Active Network Measurement, Springer Lecture Notes in Computer Science.*

Caprihan, G., Kumar, S., and Saran, H. (1997). A QoS Translation Methodology Based on User Perception. *In Proceedings of IEEE Singapore International Conference on Networking*. Singapore.

Case, J. (1990). Simple Network Management Protocol (SNMP). *RFC 1157.*

Celebi, E. (2001). *Performance Evaluation of Mobile Ad Hoc Networks Routing Protocols*. MSc Thesis in Systems and Control Engineering, Bogazici University.

Celebi, E. (2002). *Performance Evaluation of Wireless Multi-hop Ad-hoc Network Routing Protocols*. MSc thesis, Bogazici University.

Chakrabarti, S., and Mishra, A. (2001). QoS Issues in Ad Hoc Wireless Networks. *IEEE Communication magazine* **39**(2), 142-148.

Cheong, F. and Lai, R. (1999). QoS Specification and Mapping for Distributed Multimedia Systems: A Survey of Issues. *The Journal of Systems and Software*, **45**(2), 127-139.

Chiang, M. and Carlsson, G. (2001). Admission Control, Power Control and QoS Analyses for Ad Hoc Networks. *In Proceedings of IEEE International Conference on Communications*, 245 – 249. Helsinki, Finland.

Choi, Y.-H., and Hwang, I. (2005). In-service QoS Monitoring of Real-time Applications Using SM MIB. *International Journal of Network Management*, **15**(1), 31-42.

Choi, B., Moon, S., Cruz R., Zhang Z., and Diot C. (2003). Practical Delay Measurements for ISPs. *SPRINT ATL Research Report RR03-ATL- 051910.*

Cisco Internet Performance Monitor. (2004). [online] Last accessed on 6 September 2004 at URL: http:// http://www.cisco.com/en/US/products/sw/cscowork/ps1008/products_user_guide_chapter09186a0080087847.html.

Cisco Systems. (2006). [online]. Last accessed on 27 August 2006 at URL: http://www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc.

Claffy, K., Polyzos, G., and Braun, H. (1993). Application of Sampling Methodologies to Network Traffic Characterization. *In Proceedings of ACM SIGCOMM'93*. San Francisco, CA, USA.

Cleary, J. (2005). Waitko Applied Networks Dynamics (WAND). [online]. Last accessed on 6 September 2005 at URL:http://wand.cs.waikato.ac.nz/pubDetail.php?id=175.

Clink. (2002). Last accessed on 06 September 2004 at URL: http://rocky.wellesley.edu/downey/clink.

Cole, R. and Rosenbluth, J. (2001). Voice over IP Performance Monitoring. *ACM SIGCOMM, Computer Communication Review*, **31**(2), 9-24.

Converdale, P. (2003). End to End QoS - it's the user that matters. Workshop on End-to-End Quality of Service. What is it? How do we get it? [online] Last accessed on 05 September 2004 at URL: http://www.telchemy.com/conferences/2003/itu_sla.pdf. Geneva.

CoralReef (2005), CAIDA. [online]. Last accessed on 11 July 2005 at URL http://www.caida.org/tools/measurement/coralreef/.

Cottrell, L. (2001). Passive vs. Active Monitoring. [online]. Last accessed on 15 June 2006 at URL: http://www.slac.stanford.edu/comp/net/wan-mon/passive-vs-active.html.

Cranley, N. and Davis, M. (2005). Performance Evaluation of Video Streaming with Background Traffic over IEEE 802.11 WLAN Networks. *In Proceedings of WMuNeP 2005, 1st ACM International Workshop on Wireless Multimedia Networking and Performance Modelling*. Montreal, Canada.

Crawley, E., Nair, R., Rajagopalan, B. and Samdrick, H. (1998). A Framework for QoS Based Routing in the Internet. *RFC2386, Technical report.*

Curado, M. and Monteiro, E. (2001). An Overview of Quality of Service Routing Issues. *In Proceedings of the 5th World Multiconference on Systemics, Cybernetics and Informatics (SCI 2001)*. Orland, USA.

Cozzani, I. and Giordano, S. (1998). Traffic Sampling Methods for End-to-end QoS Evaluation in Large Heterogeneous Networks. *Computer Networks and ISDN Systems*, **30**(16-18), 1697-1706.

Dai, R. (2000). A Technical White Paper on Sage's PSQM Test. [online]. Last accessed on 15 June 2005 at URL: http://www.sageinst.com/downloads/925/psqmwp8_00.pdf.

D'Antonio, S., Esposito, M., Gargiulo, M., Romano, S. P. and Ventre, G. (2003). A Component-based Approach to SLA Monitoring in Premium IP Networks. *First International Workshop on Inter-domain Performance and Simulation, IPS 2003.* Salzburg.

Daoudi, K. (2006). State of the Art in Speech and Audio Processing. [online]. Last accessed on 6 June 2006 at URL: http://www-rocq.inria.fr/imedia/Muscle/WP5/Docs/WP5_SoA_Task3_sept04.pdf.

Demichelis, C. and Chimento, P. (2002). IP Packet Delay Variation Metric for IPPM. *RFC 3393.*

Di Caro, G. (2003). Analysis of Simulation Environments for Mobile Ad Hoc Networks. *Technical Report, IDSIA-24-03*. Dalle Molle Institute for Artificial Intelligence, Lugano, Switzerland.

Dong, Y., Makrakis, D., and Sullivan, T. (2003). Effective Admission Control in Multihop Mobile Ad Hoc Networks. *In Proceedings of International Conference on Communications Technology (ICCT 2003)*, 1291-1294. Ottawa University, Canada.

Dovrolis, C., Ramanathan, P., and Moore, D. (2001). What do Packet Dispersion Techniques Measure? *In Proceedings of INFOCOM 2001*. Anchorage, Alaska.

Dressler, F. (2003). A Metric for Numerical Evaluation of the QoS of an Internet Connection. *In Proceedings of the 18th International Teletraffic Congress (ITC18)*, **5b**, 1221-1230. Berlin, Germany.

Drobisz, J. and Christensen, K. (1998). Adaptive Sampling Methods to Determine Network Traffic Statistics Including the Hurst Parameter. *23rd IEEE Annual Conference on Local Computer Networks*, 238-247.

Dupcinov, M. and Krco, S. (2002). Routing in Ad Hoc Networks. *Technical report*. Applied Research Lab, EEI, Ericsson, Ireland.

Duffield, N. and Grossglauser, M. (2000). Trajectory Sampling for Direct Traffic Observation. *In Proceedings of ACM SIGCOMM'2000, Computer Communications Review*, **30**(4), 271-282. Stockholm, Sweden.

Duffield, N., Lund, C., and Thorup, M. (2001). Charging from Sampled Network Usage. *In Proceedings of ACM Internet Measurement Workshop IMW 2001*, 245-256. San Francisco, USA.

Duffield, N.G. (2004). Sampling for Passive Internet Measurement: A Review. *Statistical Science*, **19**(3), 472-498.

Eidenberger, H. (2003). Distance Measures for MPEG-7-based Retrieval. *In Proceedings of ACM Multimedia Information Retrieval Workshop, ACM Multimedia Conference Proceedings*, 130 - 137. Berkeley, California.

Eikenes, G., and O. Grostol. (2003). *Management of Quality of Service and other Fnctions in Mobile Ad Hoc Networks*. MSc. Thesis in Information and Communication Technology. Agder University College.

Engel, T., Granzer, H., Koch, B., Winter, M., Sampatakos, P., Venieris, I., Hussmann, H., Ricciato, F., and Salsano, S. (2003). AQUILA: Adaptive Resource Control for QoS Using an IP Based Layered Architecture. *IEEE Communications Magazine,* **41**(1), 46-53.

Fall, K. and Varadhan, K. (2005). The ns Manual (formally ns Notes and Documentation).

Feldmann, A. (2000). NetScope: Traffic Engineering for IP Networks. *IEEE Networks Magazine*, **14**(2), 11-19.

Ferguson, P. and Huston, G. (1998). *Quality of Service: Delivering QoS in the Internet and the Corporate Network.* Wiley Computer Books, New York.

Fernandez, M., Pedroza, A., and Rezende, J. (2003). Converting QoS Policy Specification into Fuzzy Logic Parameters. *The 18th International Teletraffic Congress.* Berlin, Germany.

Fluckiger, F. (1995). *Understanding Networked Multimedia.* Prentice Hall.

Flup, E., (1999). *Resource Allocation and Pricing for QoS Management in Computer Networks.* PhD Thesis. North Carolina State University.

Fraleigh, C. (2001). Design and Deployment of Passive Monitoring Infrastructure. *In Proceedings of Passive and Active Measurement Workshop.* Amsterdam.

Fraleigh, C., Moon, S., Lyles, B., Cotton, C., Khan, M., Moll, D., Rockell, R., Seely, T., and Diot, C. (2003). Packet-level Traffic Measurement from the Sprint IP Backbone. *IEEE Network Magazine,* **17**(6), 6-16.

Fuzzy Logic Fundamentals. (2005). [online]. Last accessed on 15 June 2005 at URL: http://www.informit.com/content/images/0135705991/samplechapter/0135705991.pdf.

Galetzka, M. (2004). User-perceived Quality of Service in Hybrid Broadcast and Telecommunication Networks. *In Proceedings of the 5th Workshop Digital Broadcasting.* Fraunhofer.

Gao, D., Cai, J., Bao, P. and He, Z. (2005). MPEG-4 Video Streaming Quality Evaluation in IEEE 802.11e WLANs. *IEEE ICIP'05,* **1**, 197-200. Italy.

Garg, S. and Kappes, M. (2002). On the Throughput of 802.11b Networks for VoIP. *Technical Report ALR-2002-012.* Avaya Labs Research, Washington DC, USA.

Graham, I., Donnelly, S., Martin, S., Martens, J., Cleary, J. (1998). Nonintrusive and Accurate Measurement of Unidirectional Delay and Delay Variation on the Internet. *In Proceedings of INET'98.* Geneva, Switzerland.

GraphPad t-test. (2004). [Online]. Last accessed on 17 September 2004 at URL: http://www.graphpad.com/quickcalcs/index.cfm.

Hall, T. (2001). Objective Speech Quality Measures for Internet Telephony in: Voice over IP (VoIP) Technology. *In Proceedings of SPIE,* 4522, 128-136. Denver, USA.

Hannikainen, M., Niemi, M., and Hamalaian, T. (2002). Performance of the Ad Hoc IEEE 802.11b Wireless LAN. *In Proceedings* of *International Conference on Telecommunications (ICT 2002),* 938-945. Beijing, China.

Hardy, W. (2002). *QoS: Measurement and Evaluation of Telecommunications Quality of Service.* John Wiley & Sons.

Hasib, M., and Schormans, A. (2003). Limitations of Passive & Active Measurement Methods in Packet Networks. *In Proceedings of London Communications Symposium (LCS2003).* London, UK.

Heegaard, P. (2002). Active Tests for SLA Validation-Correctness, Precision, and Intrusiveness. *R&DN 30/2002*. Last accessed on 5 September 2004 at URL: http://www.telenor.no/fou/publisering/notater/N_30_2002.pdf.

Hofmann, U. and Miloucheva, I. (2001). Distributed Measurement and Monitoring in IP Networks (CMToolset for AQUILA DiffServ). *SCI 2001/ISAS 2001 Orlando 7/2001*.

Hofmann, U. Miloucheva, I., Pfeiffenberger, T. and Strohmeier, F. (2002). Evaluation of Architectures for QoS Analysis of Applications in Internet Environment. *In Proceedings of the 10th International Conference on Telecommunication Systems*. Monterey, CA, USA.

Holland, G., Vaidya, N., and Bahl, P. (2001). A Rate-adaptive MAC Protocol for Multi-hop Wireless Networks. *In Proceedings of the 7th Annual International Conference on Mobile Computing and Networking*, 236 – 250. Rome, Italy.

Hu, N. and Steenkiste, P. (2003). Evaluation and Characterization of Available Bandwidth Probing Techniques. *In the IEEE Journal on Selected Areas in Communications, Special Issue in Internet and WWW Measurement, Mapping, and Modelling*, **21**(6), 879- 894.

Hudson, D., and Cohen M. (2000). *Neural Networks and Artificial Intelligence for Biomedical Engineering*. New York: IEEE press.

IETF PSAMP. (2006). [online]. Last accessed on 22 March 2006 at URL: http://psamp.ccrle.nec.de.

Ishibanishi, K., Kanzawa, T., Aida, M., and Ishii, H., (2004). Active/Passive Combination-type Performance Measurement Method Using Change-of-Measure Framework. *ELSEVIER Computer Communication*, **27**(9), 868-879.

Ishibashi, K., Kanazawa, T., and Aida, M. (2002). Active/Passive Combination type Performance Measurement Method Using Change-of-measure Framework. *In Proceedings of IEEE GLOBECOM 2002*, 2538-2542. Taipei, Taiwan.

ITU-T Recommendation G.711. (1998). Pulse Code Modulation of Voice Frequencies. ITU Geneva.

ITU-T Recommendation E.800. (1994). ITU-T E.800: Terms and Definitions Related to Quality of Service and Network Performance Including Dependability.

ITU-T Recommendation H.323. (1999). Infrastructure of Audiovisual Services-Systems and Terminal Equipment for Audiovisual Services, Series H: Audiovisual and multimedia systems.

ITU-T Recommendation P.800. (1996). Methods for Subjective Determination of Transmission Quality.

ITU-T Recommendation P.830. (1996). Subjective Performance Assessment of Telephone-band and Wideband Digital Codecs.

ITU-T Recommendation 861. (1998). Objective Quality Measurement of Telephone-band (300-3400 Hz) Speech Codecs.

ITU-T Recommendation G.108. (1999). Application of the E-model: A planning guide.

ITU-R Recommendation BT.500-10. (2000). Methodology for the Subjective Assessment of the Quality of Television Pictures.

ITU-T Recommendation G.1000. (2001). Communications Quality of Service: A Framework and Definitions.

ITU-T Recommendation G.1010. (2001). End-user Multimedia QoS Categories.

ITU-T Recommendation P.862. (2001). Perceptual Evaluation of Speech Quality (PESQ), An Objective Method for End-to-End Speech Quality Assessment of Narrow-band Telephone Networks and Speech Codecs.

ITU-T Recommendation P.910. (2002). Subjective Video Quality Assessment Methods for Multimedia Applications.

ITU-T: Recommendation G.107. (2003). The E-Model, A Computational Model for Use in Transmission Planning (2003) ITU–T Standard.

Jedwab, J., Phaal, P., Pinna, B. (1992). Traffic Estimation for the Largest Sources on a Network: Using Packet Sampling with Limited Storage. *HP Technical Report*. HP Laboratories, Bristol.

Jiang, W. (2003). *QoS Measurement and Management for Internet Real-time Multimedia Services*. PhD thesis, Columbia University.

Jiang, Y., Tham, C. and Ko, C. (2000). Challenges and Approaches in Providing QoS Monitoring. *International Journal of Network Management, Wiley/ACM*, **10**(6), 323-334.

Johnsson, A. (2005). *Bandwidth Measurements in Wired and Wireless Networks*. Licentiate thesis, Mälardalen University Press.

Kajackas, A., and Anskaitis, A. (2005). Investigation the Ability of Objective Measures of the Perceptual Speech Quality in Mobile Networks. *Kaunas: Technologija*, **7**(63), 10-15.

Kanodia, V., Li, C., Sabharwal, A., Sadegi, B., and Knightly, E. (2002). Ordered Packet Scheduling in Wireless Ad Hoc Networks: Mechanisms and Performance Analysis. *In Proceedings of the 3rd ACM international symposium on Mobile ad hoc networking & computing,* 58-70. Lausanne, Switzerland.

Kasigwa, J., Baryamureeba, V., and Williams, D. (2004). A Framework for End-to-End Quality of Service Provisioning Models in IP based Networks. *Transactions on Engineering, Computing and Technology*, **3**, 223-228. Istanbul, Turkey.

Khurana, S., Kahol, A., and Jayasumana, A. (1998). Effect of Hidden Terminals on the Performance of IEEE 802.11 MAC Protocol. *In Proceedings of 23rd Annual Conference on Local Computer Networks*, 12-20. Boston, Massachusetts, USA.

Kleinrock, L. and Tobagi, F. A. (1975). Packet Switching in Radio Channels: Part 2: The Hidden Terminal Problem in Carrier Sense Multiple-Access and the Busy-Tone Solution. *IEEE Transactions on Communications*, COM-**23**(12), 1400–1416.

Koepsel, A. and Wolisz, A. (2001). Voice Transmission in an IEEE 802.11 WLAN Based Access Network. *In Proceedings of the 4th ACM International Workshop on Wireless Mobile Multimedia (WoWMoM 2001)*, 23–32. Rom, Italy.

Koepsel, A., Ebert, J.-P., and Wolisz, A. (2000). A Performance Comparison of Point and Distributed Coordination Function of an IEEE 802.11 WLAN in the Presence of Real-Time Requirements. *In Proceedings of the 7th International Workshop on Mobile Multimedia Communications (MoMuC 2000)*. Tokyo, Japan.

Krishnaiah, P. and Rao, C. (1988). *Handbook of Statistics: Volume 6, Sampling*, Elsevier Science Publishing.

Kuncheva, L., and Steimann, F. (1999). Fuzzy diagnosis. *Artificial Intelligence in Med.*, **16**(2), 121-128.

Kwon, T., Choi Y., Bisdikian, C., and Naghshineh, M. (2003). QoS Provisioning in Wireless/Mobile Multimedia Networks Using an Adaptive Framework. *Wireless Network*, **9**(1), 51-59.

Lai, K., and Baker, M. (2000). Measuring Link Bandwidths Using a Deterministic Model of Packet Delay. *In Proceedings of ACM SIGCOMM 2000*, 283–294. Stockholm.

Landfeld, B., Seneviratne, A., Melander, B., and Gunninberg, P. (1999). Network Performance Monitoring for Applications Using EXPAND. *Engineering Transactions*, **2**(3), 155–163.

Landfeld, B., Sookavatana, P., and Seneviratne, A. (2000). The Case for a Hybrid Passive/Active Network Monitoring Scheme in the Wireless Internet. *In Proceedings of IEEE International Conference on Networks (ICON'00)*, 139-143. Singapore.

Li, B., Chang, E. and Wu, C.-T. (2002). DPF - A Perceptual Distance Function for Image Retrieval. *In Proceedings of IEEE International Conference on Image Processing*, 597-600. New York.

Lindgren, A., Almquist, A., and Schel'en, O. (2003). Quality of Service Schemes for IEEE 802.11 Wireless LANs - An Evaluation. *In Special Issue of the Journal on Special Topics in Mobile Networking and Applications (MONET) on Performance Evaluation of QoS Architectures in Mobile Networks*, **8**(3):223–235.

Lindh, T. (2001). A Framework for Embedded Monitoring of QoS Parameters in IP-Based Virtual Private Networks. [online]. *In Proceedings of Passive and Active Measurement Workshop (PAM2001)*. Amsterdam, The Netherlands. Last accessed on 5 September 2004 at URL: http://www.nlanr.net/Data-users/cityear.html.

Lindh, T. (2002). A New Approach to Performance Monitoring in IP Networks - Combining Active and Passive Methods. *In Proceedings of Passive and Active Measurements (PAM 2002)*, 128-137. Colorado, USA.

Lo, P. and Ngai, S. (2004). *Characterising Errors in Wireless LANs*. [online]. *Master Thesis*. School of Electrical Engineering & Telecommunications, University of New South Wales. Last accessed on 25 August 2006 at URL: http://uluru.ee.unsw.edu.au/~tim/projects/measure:11/2004/thesis.pdf.

Ma, W., Yan, J., and Huang, C. (2004). Adaptive Sampling for Network Performance Measurement under Voice Traffic. *In Proceedings of IEEE International Conference on Communications (ICC'04)*, **2**, 1129- 1134. Paris.

Macker, J., and Courson, S. (2003). *Mobile Ad Hoc Networks (MANET)*. IEEE Press Wiley, New York.

Maggie, Ng., and Matchman H. (2000). QoS Schemes in Wireless Networks. [online]. The Chinese University of Hong Kong, Computer Science and Engineering Department. Last accessed on 19 February 2004 at URL: http://www.cse.cuhk.edu.hk/~lyu/student/fyp99/lyu9903/.

Manku, G. and Motwani R. (2002). Approximate Frequency Counts over Data Streams. *In Proceedings of the 28th ACM International Conference on Very Large Data Bases (VLDB)*, 346-357.

Mark, L., Pohl, G., Schmoll, C. and Zseby, T. (2003). Passive One Way Delay Measurements and Data Export. *In Proceedings of the First International Workshop on Inter-domain Performance and Simulation*. Salzburg.

Markopoulou, A., Tobagi, F., Karam, M. (2002). Assessment of VoIP Quality over Internet Backbones. *In Proceedings of IEEE INFOCOM. 2002 - The Conference on Computer Communications,* **1**, 150- 159. New York, USA.

Marshall, A. (2003). Introduction to Multimedia. [online]. Last accessed on 05 September 2004 at URL: http://www.cs.cf.ac.uk/Dave/Multimedia/PDF/BSC_MM_handouts_03.pdf.

Masala, E., Chiasserini, C.-F., Meo, M., and Martin J. (2003). Real-time Transmission of H.264 Video over 802.11-based Wireless Ad Hoc Networks. *In Proceedings of Workshop on DSP in Mobile and Vehicular Systems*. Nagoya, Japan.

The Mathworks - Fuzzy Logic Toolbox. (2005). [online]. Last accessed on 07 July 2005 at URL: http://www.mathworks.com/access/helpdesk/help/toolbox/fuzzy/index.html.

Marsh, I., and Li, F. (2003). Wide Area Measurements of Voice over IP Quality. *SICS Technical Report T2003:08.*

Matthews, W. and Cottrell, L. (2000). The PingER Project: Active Internet Performance Monitoring for the HENP Community. *IEEE Communications Magazine*, **38**(5), 130-136.

McGregor, T., Braun, H.-W., and Brown, J. (2000). The nlar network analysis infrastructure. *IEEE Communication Magazine*, **38**(5).

Michaut, F. and Lepage, F. (2005). Application-oriented Network Metrology: Metrics and Active Measurement Tools. *IEEE Communications Surveys & Tutorials*, 7(2).

Miloucheva, I., Nassri A., and Anzaloni, A. (2004). Automated Analysis of Network QoS Parameters for Voice over IP Applications. *In Proceedings of the 2nd International Workshop on Inter-Domain Performance and Simulation (IPS 2004).* Budapest, Hungary.

Mohamed, S., Cervantes-P′erez, F., and Afifi, H. (2000). Real-Time Audio Quality Assessment in Packet Networks. *Network and Information Systems Journal*, 595–609.

Mohamed, S., Cervantes-P′erez, F., and Afifi, H. (2001). Integrating Networks Measurements and Speech Quality Subjective Scores for Control Purposes. *In Proceedings of IEEE INFOCOM'01*, **2**, 641-649. Alaska, USA.

Mohamed, S. and Rubino, G. (2002). A Study of Real-Time Packet Video Quality Using Random Neural Networks. *IEEE Transactions on Circuits and Systems for Video Technology*, **12**(12), 1071–1083.

Mohamed, S., Rubino, G., and Varela, M. (2004). A Method for Quantitative Evaluation of Audio Quality over Packet Networks and its Comparison with Existing Techniques. *In Proceedings of the Measurement of Speech and Audio Quality in Networks workshop (MESAQIN'04)*, Prague, Czech Republic.

Molina-Jimenez, C., Shrivastava, S., Crowcroft, J. and Gevros, P. (2004). On the Monitoring of Contractual Service Level Agreements. *In Proceedings of the IEEE Conference on Electronic Commerce CEC'04, The First IEEE International Workshop on Electronic Contracting (WEC)*, 1-8. San Diego.

Mullin, J., Smallwood, L., Watson, A., Wilson, G. (2001). New Techniques for Assessing Audio and Video Quality in Real-time Interactive Communications. *IHM-HCI Tutorial.*

Nahrstedt, K. and Chen, S. (1998). An Overview of Quality of Service Routing for Next-Generation High-Speed Networks: Problems and Solutions. *IEEE Network*, **12**(6), 64-79.

Nedeljkovic, I. (2004). Image classification based on fuzzy logic. *In Proceedings of Geo-Imagery Bridging Continents XXth ISPRS Congress*, 83-88. Istanbul, Turkey.

NetFlow Services and Applications. (2005). [online]. White Paper Cisco Systems. Last accessed on 20 September 2005 at URL: http://www.cisco.com/warp/public/cc/pd/iosw/ioft/neflct/tech/napps_wp.htm.

Nikaein, N. and Bonnet, C. (2002). A Glance at Quality of Service Models for Mobile Ad Hoc Networks. *16eme Congress DNAC (De Nouvelles Architectures pour les Communications)*. Paris, France.

National Laboratory for Applied Network Research (NLANR). (2006). [online]. Last accessed on 27 August 2006 at URL: http://moat.nlanr.net/.

The Network Simulator - NS-2. (2005). [online]. Last accessed on 23 March 2005 at URL:http://www.isi.edu/nsnam/ns.

Oliveira, R. and Braum, T. (2004). A Fuzzy Logic Engine to Assist TCP Error Detection in Wireless Mobile Ad Hoc Networks. *Next Generation Teletraffic and Wired/Wireless Advanced Networking New2an'04.*

Papagiannaki, K., Moon, S., Fraleigh, C., Thiran, P., Tobagi, F., Diot, C. (2002). Analysis of Measured Single-Hop Delay from an Operational Backbone Network. *In Proceedings of IEEE INFOCOM*. New York, USA.

Pathchar. (2002). [online]. Last accessed on 27 August 2006 on URL: http://www.caida.org/tools/utilities/others/pathchar.

Paul, D. and Lillian, N. (1989). Management of Sampled Real-time Network Measurements. *In Proceedings of the 14th IEEE Conference on Local Computer Networks*, 62-68. Minneapolis, USA.

Paxson, V. (1999). End-to-end Internet Packet Dynamics. *IEEE/ACM Transactions on Networking*, 7(3), 277–292.

Paxson, V. (1997). *Measurements and Analysis of End-to-End Internet Dynamics*. PhD thesis, University of California, Berkeley, USA.

Passive Measurements. (2005). [online]. Last accessed on 16 June 2005 at URL: http://wand.cs.waikato.ac.nz/pubs/19/html/node9.html.

Pasztor, A., and Veitch, D. (2001). High Precision Active Probing for Internet Measurement. *In Proceedings of INET'2001*. Stockholm, Sweden.

Pchar. (2002). [online]. Last accessed on 26 August 2006 at URL: http://www.employees.org/~bmah/Software/pchar.

Pennock, S. (2002). Accuracy of the Perceptual Evaluation of Speech Quality (PESQ) Algorithm. *In Proceedings of Workshop Measurement of Speech and Audio Quality in Networks MESAQIN'*02. Prague, Czech Republic.

Pias, M. and Wilbur, S. (2001). EdgeMeter: Distributed Network Metering. *In Proceedings of the International IEEE Openarch Conference*. Anchorage, Alaska.

Pitsillides, A. and Sekercioglu, A. (1999). Fuzzy Logic Based Congestion Control. *COST 257: Impacts of New Services on the Architecture and Network Performance of Broadband Networks*. Larnaca, Cyprus.

Postel, J. (1981). Internet Control Message Protocol. *RFC 792*. [online]. Last accessed on 26 August 2006 at URL: http://www.ietf.org/rfc/rfc0792.txt.

Portoles-Comeras, M., Mangues-Bafalluy, J., Requena-Esteso, M. (2006). Multi-radio Based Active and Passive Wireless Network Measurements. *In Proceedings of the*

International Symposium on Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks (WinMee'06), 1-6. Boston, USA.

Portoles-Comeras, M., Requena-Esteso, M., Mangues-Bafalluy, J., Cardenete-Suriol, M. (2006). Monitoring Wireless Networks: Performance Assessment of Sniffer Architectures. *In Proceedings of the IEEE International Conference on Communications (ICC 2006)*. Istanbul, Turkey.

Qiang, Ni, Romdhani, L., Turletti, T. (2004). A Survey of QoS Enhancements for IEEE 802.11 Wireless LAN. *Journal of Wireless Communications and Mobile Computing*, 4(5), 547-566.

Räisänen, V. (2003). *Implementing Service Quality in IP Networks*. John Wiley & Sons, Ltd.

Raisanen, V., Grotefeld, G., and Morton, A. (2002). Network Performance Measurement for Periodic Streams. *RFC 3432*. [online]. Last accessed on 26 August 2006 at URL:

Rix, A., Beerends, J., Hollier, M., and Hekstra, A. (2000). PESQ – the New ITU Standard for End-to-End Speech Quality Assessment. *AES 109th Convention*. Los Angeles, USA.

Rix, A. and Hollier, M. (2000). The Perceptual Analysis Measurement System for Robust End-to-End Speech Quality Assessment. *In Proceedings of IEEE ICASSP'00*, **3**, 1515-1518. Istanbul, Turkey.

Romaszko, S. and Blondia, C. (2004). A Survey of MAC protocols for Ad Hoc Networks and IEEE 802.11. *In Proceedings of the 4th National Conference MiSSI 2004*, 23-33. Poland.

Ross, T. (2004). *Fuzzy Logic with Engineering Applications*. 2nd edition, John Wiley & Sons, Ltd.

Rubino, G., Varela, M. and Bonnin, J. (2006). Controlling Multimedia QoS in the Future Home Network Using the PSQA Metric. *The Computer Journal* **49**(2), 137-155.

Royer, E. and Chai-Keong, T. (1999). A Review of Current Routing Protocols for Ad Hoc Mobile Wireless Networks. *IEEE Personal Communications*, **6**, 46-55.

Sadegh, K. (2001). The Fuzzy Revolution: Goodbye to the Aristotelian. Weltanschauung. *Artif. Intell. Med.*, **21**(1-3), 1-25.

Saliba, J., Beresford, A., Ivanovich, M., and Fitzpatrick, P. (2005). User-Perceived Quality of Service in Wireless Data Networks. *Personal and Ubiquitous Computing*, **9**(6), 413 - 422.

Saraireh, M., Saatchi, R., Shur, U. and Strachan, R. (2004). Fuzzy Logic Based Evaluation of Quality of Service for Multimedia Transmission. *In Proceedings of PREP 2004*. University of Hertfordshire, Hertfordshire, UK.

Schmitt, J., Heckmann, O., Pandit, K., Hoffmann, M., Jobmann, M. (2002). LETSQoS Milestone 1, Network QoS - An Up-to-date Overview. *LETS QoS Project*. Technical University of Kaiserslautern.

Shah, I. (2001). Bringing Comprehensive Quality of Service Capabilities to Next Generation Networks. (2001). White Paper for Motorola's C-Port Family of Network Processors. [online]. Last accessed on 27 August 2006 at URL: http://www.freescale.com/files/netcomm/doc/white_paper/QOSM-WP.pdf.

Sheu, J., Liu, C., Wu, S., and Tseng, Y. (2004). A Priority MAC Protocol to Support Real-time Traffic in Ad hoc Networks. *Wireless Networks*, **10**(1), 61-69.

Skitter. (2002). [online]. Last accessed on 26 August 2006 at URL: http://www.caida.org/tools/measurement/skitter.

Smotlacha, V. (2001). QoS Oriented Measurement in IP Networks. *CESNET Technical Report Number 17/2001*.

Stine, J. and Veciana, G. (2004). A Paradigm for Quality of Service in Wireless Ad Hoc Networks Using Synchronous Signalling and Node States. *IEEE Journal on Selected Areas in Communications*, **22**(7), 1301-1321.

Sun, L. (2004). *Speech Quality Prediction for Voice over Internet Protocol Networks*. PhD thesis. School of Computing, Communications and Electronics, University of Plymouth, UK.

Surveyor Measurement Tool. (2004). [online]. Last accessed on 06 September 2004 at URL: http://www.advanced.org/surveyor/.

Sweet, C., Devarapalli, V., and Sidhu, D. (1999). IEEE 802.11 Performance in an Ad Hoc Environment. *Technical Report*, UMBC, Department of Computer Science & Electrical Engineering.

Syrjälä, M. (2003). Wireless Networks Performance, Quality of Service and Interoperability. [online]. Last accessed on 06 September 2004 at URL: http://www.tml.hut.fi/Studies/T-110.551/2003/papers/2.pdf.

Tanenbaum, A. (2003). *Computer Networks*. 4th edition, New Jersey, Prentice Hall PTR.

Tatsuya, Y. (2001). Subjective Video Quality Assessment for Adaptive Quality-of-Service Control. *In Proceedings of IEEE International Conference on Multimedia and Expo (ICME)*, 397- 398. Tokyo, Japan.

Teknomo, K. (2006). [online]. Last accessed on 01 June 2006 at URL: http://people.revoledu.com/kardi/tutorial/Similarity/EuclideanDistance.html.

Terano, T. (1992). *Fuzzy Systems and its Applications*. Academic Press Inc.

Tham, C., Jiang, Y. and Ko, C. (2000). Monitoring QoS Distribution in Multimedia Networks. *International Journal of Network Management*, **10**, 75-90.

Toh, C., Delwar, M., and Allen, D. (2002). Evaluating the Communication Performance of an Ad Hoc Wireless Networks. *IEEE Transactions on Wireless Communications*, **1**(3), 402–414.

Traceroute. (2002). [online]. Last accessed on 06 March 2004 at URL: ftp://ftp.ee.lbl.gov/tracroute.tar.gz.

Tseng, Y., Ni, S., Chen, Y., and Sheu, J. (1999). The Broadcast Storm Problem in a Mobile Ad Hoc Network. *In Proceedings of the 5th annual ACM/IEEE International Conference on Mobile Computing and networking*, 151-162. Seattle, USA.

Internet Draft of t-test. (2005). [Online]. Last accessed on 22 September 2005 at URL: http://faculty.vassar.edu/lowry/ch11pt1.html.

Valaee, S., and Li, B. (2002). Distributed Call Admission Control for Ad Hoc Networks. *IEEE Vehicular Technology Conference (VTC)*, **2**, 1244-1248. Vancouver, BC, Canada.

Visser, M. and El Zarki, M. (1995). Voice and Data Transmission over an 802.11 Wireless Network. *In Proceedings of IEEE PIMRC'95*, 648–652. Toronto, Canada.

Wang, Ph., Yemini, Y., Florissi, D., Zinky, J., and Florissi, P. (2000). Experimental QoS Performances of Multimedia Application. *In Proceedings of IEEE INFOCOM 2000 - The Conference on Computer Communications,* **1**, 970-979. Tel Aviv, Israel.

Watson, A. and Sasse, M. (1996). Evaluating Audio and Video Quality in Low-Cost Multimedia Conferencing Systems. *Interacting with Computers*, **8**(3), 255-275.

We, T., and Chen, S. (1999). A New Method for Constructing Membership Functions and Fuzzy Rules from Training Examples. *IEEE Transactions on Systems, Man and Cybernetics - Part B: Cybernetics,* **29**(1), 25-40.

Wu, H., Lambrecht, C., Yuen, M., and Qiu, B. (1996). Quantitative Quality and Impairment Metrics for Digitally Coded Images and Image Sequences. *In Proceedings of Australian Telecommunication Networks & Applications Conference*, **2**, 389-394. Melbourne.

Wu, S. and Pols, L. (1996). A Distance Measure for Objective Quality Evaluation of Speech Communication Channels Using Also Dynamic Spectral Features. *In Proceedings of the Institute of Phonetic Sciences Amsterdam*, **20**, 27-42.

Xue, Q. and Ganz, A. (2003). Ad Hoc QoS On-demand Routing (AQOR) in Mobile Ad Hoc Networks. *Journal of Parallel and Distributed Computing*, **63**,154-165.

Yang, Y., Wan, J., and Kravets, R. (2003). Achievable Bandwidth Prediction in Multihop Wireless Networks. *Technical Report, UIUCDCS-R-2003-2367.* University of Illinois at Urbana-Champaign, Urbana, IL.

Yeo, J., Banerjee, S. and Agrawala, A. (2002). Measuring Traffic on the Wireless Medium: Experience and Pitfalls. *Technical Report CS-TR 4421*, Department of Computer Science, University of Maryland, College Park.

Yeo, J., Youssef, M., Agrawala, A. (2004). A Framework for Wireless LAN Monitoring and its Applications. *In Proceedings of the 3rd ACM Workshop on Wireless Security (WiSe'04), in Conjunction with ACM MobiCom'04*, 70 - 79. Philadelphia, USA.

Zadeh, L. (1965). Fuzzy Sets. *Information and Control*, **8**, 338-353.

Zhai, H., Kwon, Y. and Fang, Y. (2004). Performance Analysis of IEEE 802.11 MAC Protocols in Wireless LANs. *Wireless Communications and Mobile Computing, Special Issue on Emerging WLAN Technologies and Applications*, **4**(8), 917–931.

Zhai, H., Chen, X., and Fang, Y. (2005). How Well Can the IEEE 802.11 Wireless LAN Support Quality of Service? *IEEE Transactions on Wireless Communications,* **4**(6), 3084-3094.

Zhao, W., Olshefski, D., and Schulzrinne, H. (2000). Internet Quality of Service: An Overview. *Technical Report CUCS-003-00*. Columbia University.

Zseby, T., Zander, S. and Carle, G. (2001). Evaluation of Building Blocks for Passive One-way Delay Measurements. *In Proceedings of Passive and Active Measurement Workshop (PAM 2001)*. Amsterdam, The Netherlands.

Zseby, T., and Scheiner, F. (2004). QoS Monitoring and Measurement Benchmarking. *Information Society Technology (Next Generation Networks)*. [online]. Last accessed on 06 September 2004 at URL: http://www.ist-mome.org/documents/M_and_M_benchmark.pdf.

Zseby, T. (2002). Deployment of Sampling Methods for SLA Validation with Non-Intrusive Measurements. *In Proceedings of Passive and Active Measurement Workshop (PAM 2002)*, 84-94. Colorado, USA.

Zseby, T. (2003). Stratification Strategies for Sampling-based Non-intrusive Measurements of One-way Delay. *In Proceedings of Passive and Active Measurement Workshop (PAM 2003)*, 171-179. La Jolla, USA.

Zseby, T. (2004). Comparison of Sampling Methods for Non-Intrusive SLA Validation. *In Proceedings of 2nd Workshop on End-to-End Monitoring Techniques and Services (E2EMON)*. San Diego, USA.

Zseby, T. (2005). Sampling Techniques for Non-Intrusive QoS Measurements: Challenges and Strategies. [online]. *Computer Communications Special Issue on Monitoring and Measurement.* Last accessed on 28 August 2006 at URL: http://www.6qm.com/files/cc-special-sampling-tech-2004-08-31.pdf.