The Effects of Network Factors on the Performance of 3G UMTS Applications

by

Fabian Mark Jacobs

A thesis submitted to the University of Cape Town in fulfilment of the requirements for the degree of Master of Science in the Department of Computer Science

Cape Town

February 2010

Supervised by

Prof. Ken Macgregor

.

The Effects of Network Factors on the Performance of 3G UMTS Applications

By

Fabian Mark Jacobs

A thesis submitted to the University of Cape Town in fulfilment of the requirements for the degree of Master of Science in the Department of Computer Science

Cape Town

February 2010

Supervised by

Prof. Ken Macgregor

.

Acknowledgements

I would thank my supervisor Prof. Ken MacGregor for giving me the opportunity to do my Masters in my field of interest and guiding me in the direction of my thesis topic. Thank you to Andrew Gill and James Lane for their assistance in reading and checking my thesis. Special thanks to my girlfriend Lona, my family and friends for their support and understanding during my very busy work and study schedule. Thank you to my place of work Soft Craft Systems for their understanding and flexibility regarding my studies. I thank God who makes all things possible, I thank God for the family and friends who supported and assisted me throughout my Masters.

.

Table of Contents

List of Figures

List of Tables

Table of Acronyms

Acronym	Meaning
3G	Third Generation
3GPP	3rd Generation Partnership Project
BS	Base Station
CN	Core Network
DCH	Dedicated Channel
DNS	Domain Name Server
EDGE	Enhanced Data Rates for GSM Evolution
EURANE	Enhanced UMTS Radio Access Network Extension
FDD	Frequency Division Duplex
FTP	File Transfer Protocol
GGSN	Gateway GPRS (3G) Server Node
GMSC	Gateway Mobile Switching Centre
GPRS	General Packet Radio Service
GSM	Global System for Mobile communication
HLR	Home Location Register
HSDPA	High Speed Download Packet Access
HSS	Home Subscriber server
HTTP	Hyper Text Transfer Protocol
IMS	Internet and Multimedia System
IP	Internet Protocol
L1	Physical Layer
MAC	Medium Access Control
MDA	Mail Delivery Agent
MMS	Multimedia Message Service
MMSC	Multimedia Messaging Service Centre
MS	Mobile Station
MSC	Mobile Switching Centre
MUA	Mail User Agent
Node B	Base Station
NS2	Network Simulator 2
OTcl	Object Tickle (a scripting language)
OVSF	Orthogonal Variable Spreading Factor
PDP	Packet Data Protocol

RLC	Radio Link Control
RNC	Radio Network Controller
RTCP	Real Time Control Protocol
RT	Real Time
RTP	Real Time Protocol
RTSP	Real Time Streaming Protocol
SGSN	Serving GPRS (3G) Server Node
SIP	Session Initiation Protocol
SMS	Short Message Service
SMSC	Short Message Service Centre
SMTP	Simple Mail Transfer Protocol
SS7	Signalling System #7
TCP	Transfer Control Protocol (Vegas is a TCP implementation)
TDD	Time Division Duplex
TDMA	Time Division Multiple Access
TTI	Transmission Time Interval
UDP	User Datagram Protocol
UE	User Entity
UMTS	Universal Mobile Telecommunication System
URL	Universal Resource Locator
UTRAN	UMTS Terrestrial Radio Access Network
VLR	Visitors Location Register
WAP Gateway	Wireless Application Protocol Gateway
WCDMA	Wide Code Division Multiple Access

Abstract

3G is the wireless network technology expected to allow wireless applications to perform on par with that of wired applications. However 3G has factors which limit its performance. These factors include both device factors such as small screens, limited battery power and life, as well as network factors such as high delay networks and low bandwidths. This thesis investigates the following questions:

- How do network factors affect the performance of 3G UMTS applications?
- Which network factors have the most significant impact on a specific application?
- Are there any minimum requirements needed for an application?

Eight popular 3G applications were investigated: FTP, email, MMS, SMS, HTTP web browsing, broadcast media, video calling and streaming media.

This was done through the use of NS2 simulator. Many networks are set up with varying network conditions. Each application was then simulated on each of these network set-ups and the results were recorded. The results of the various network conditions were then compared to each other and analysed. The effect on a specific application as a specific network factor was changed was identified.

The results indicate that not all applications are affected to the same degree by each network factor. The SMS service is immune to all network factors, while most other applications are reliant on bandwidth as the major factor. HTTP web browsing results show end-to-end delay as being its major factor. When a low bandwidth is used the FTP, email and MMS services are still operational, whereas in broadcast media, video calling and streaming media the service quality will be unacceptable to the user if the bandwidth is too low, deeming these services un-usable. End-to-end delay has no effect on streaming applications. Based on results of these simulations, suggestions are made to maximise cell performance.

Chapter 1

Introduction

1.1 Background to Research

3G is the latest generation of wireless cellular technology to be deployed world wide. This will allow anywhere anytime multimedia services. 3G is credited as the wireless technology which will be able to deliver a range of multimedia services on par with that of wired networks.

The use of wireless technology is impacted by several factors limiting its performance. The range of factors include, small devices limited in memory, processing power and battery life, as well as network features such as limited bandwidth, high delay, high noise and bit error rates. [Chan, 2006] mentions the scarcity of available radio spectrum as a factor limiting 3G performance. These factors may cause 3G to underperform, detracting from the users' experience. The performance is significantly affected when the network is shared by many users; users' throughput drops markedly from that of the theoretical quoted ideal 3G throughput value.

The research lists and investigates the factors which limit 3G wireless networks performance. The focus is on network factors and how they affect the performance of an application. In particular, are there certain minimum conditions needed for an application to perform acceptably and which of these factors have the greatest effect on a given application's performance. The results show the change in application performance as the network conditions change.

Some of the popular services offered in 3G were examined: FTP, SMS, MMS, email, web browsing, media streaming services (broadcasting media, streaming media and video calling) to investigate what affects 3G's performance on wireless networks. These applications were simulated under various network conditions, and performance measured on metrics that are user centred (measurements which will interest a user of the service). These are mainly speed and time measurements, and number of users that can enjoy a service concurrently.

Performance, which for this research is defined as the throughput achieved by an application, is also affected by the relatively high delay of wireless networks compared to that of wired networks. These and other factors of wireless networks affect application performance. As wireless technologies have progressed, theoretical computation and experimental analysis have measured network performance, usually with a solution in mind to improve performance. Past research has not concentrated on how performance is affected when a network factor is varied through legitimate values or what users could expect from various legitimate network conditions. The present research attempts to do this through the simulation of 3G applications in varying network conditions. Researchers simulated a number of popular 3G applications and focused on throughput and time taken to execute the service.

This research aims to show the effects on performance for each of the above listed applications when the air interface bandwidth, end-to-end delay, and size of target data to be transferred over the UMTS network is varied. The intention is to illustrate the relationship between network factors and application performance, as well as identifying which applications are more prone to which network factors.

1.2 Research Questions

How network factors affect the performance of an application when these factors are varied through a range of legitimate values?

Which network factors is a specific application most prone to?

Are there minimum conditions needed for an application to run acceptably?

What is the recommended number of simultaneous users of the application?

1.3 Simulation Using NS2

Using the popular NS2 simulator with the EURANE extension the performance of application services under various network conditions was investigated. The use of simulators has become important in the study of networks as it is cheaper, faster and easier to analyse network performance [Fledderus and Springer, 2005] and [Hamalainen, 2003]. In this research NS2 is used to simulate the different applications running on a 3G UMTS network. NS2 allowed for easy modifications of network conditions and re-execution of simulations. Results of these simulations for different network conditions were compared to each other and analysed. Present research results are then compared with previous research for validation purposes.

The results are discussed in chapter 6. These give a more clear idea of the performance expected from 3G UMTS networks for different network conditions. Some of the notable results are that in media streaming applications the end-to-end delay has virtually no affect on performance results, while bandwidth has a significant affect on performance (These are taken in zero bit error network). In FTP applications, TCP congestion control manages the flow of data and for small file transfers the TCP window size never grows large enough to utilise a high bandwidth rendering the bandwidth ineffective. But for large file transfers there is a notable difference.

1.4 A Brief History of Wireless Telecommunication Networks

First generation networks were completely analogue and only supported voice data. These first networks were circuit switched and had many throughput and capacity problems. Throughput per a user was around the region of 2400bits/s. Other problems early wireless technology experienced was a high frequency of dropped calls, lack of security and bad reuse of the frequency spectrum [UK Telematics online, no date].

Second generation networks were digital but continued to use circuit switching technology. The benefits of digital technology were higher quality voice traffic and increased capacity. Users were able to obtain data rates of 9.6-14.4Kbps. The introduction of multiple access methods allowed users to share a channel, thus reducing congestion and increasing the capacity. Data services such as fax and short messaging were also supported. TDMA (Time Division Multiple Access) technology allowed for a 3 times gain in capacity compared to first generation analogue networks. [3G.co.uk, no date] and [TeliaSonera, 2004]. There exist 2.5G technologies such as GPRS which was used to bridge the gap between 2G and 3G. GPRS uses packet switching technology and supports speeds of up to 115.2.Kbits/s, marketed as a technology to allow web browsing, MMS, and short video clip downloads [UK Telematics online, no date].

Third generation cellular technology supports full packet switching capability and higher data rates of up to 2Mbps. This allows 3G to offer a larger range of services than previous generations of cellular technologies and at a better performing standard. Chapter 3 discusses UMTS, a 3G cellular technology, in detail.

1.5 Applications in 3G

Generally the performance and quality of services of an application is higher with low latency, low error rates, and high bandwidth. But not all applications are affected to the same degree when these factors are not present. 3G offers four classes of traffic. Each service's traffic will fall into one of these classes. Each class has different quality of service set upon it, making it suitable for different applications [Roccetti et al, 2005]. The four classes are:

> Conversational – This is traffic that needs low latency with non retransmission or error correction code, applications such as voice conversations, video telephony, and real time games fall into this category.
>
> Streaming – This is for video and audio downloads where a high bandwidth is needed, at the start of a download a buffer can be downloaded and used during play back to compensate for a loss of bandwidth.
>
> Interactive – This is for web browsing applications and multiplayer gaming that is not dependent on latency such as turn based games.
>
> Background – This is for SMS, MMS and FTP applications.

The conversational and streaming traffic classes are used to carry real-time traffic flows like audio and video. The interactive and background classes are used for non real-time traditional Internet applications (web browsing, telnet, e-mail and FTP).

1.6 Research Overview

In Chapter 2, previous research done on application performance in wireless networks is discussed and the outcomes presented.

In Chapter 3, UMTS a 3G implementation and WCDMA its respective air interface technology is described.

In Chapter 4, overviews of the applications that will be simulated in this research are given.

In Chapter 5, NS2, EURANE and its components are looked at. This is the simulator used by the research to investigate application performance over 3G UMTS wireless networks.

In Chapter 6, the experiments, results and the analyses of this research is illustrated and discussed.

In Chapter 7, the conclusion of the research is presented and future work stated.

Chapter 2

Literature Review

In this chapter relevant and related research on services, issues and factors of wireless cellular networks are discussed. The performance of SMS and the type of applications it is best suited to are reviewed. Then FTP applications and its performance measurements are inspected. This is followed by the performance measurements of web browsing and media streaming services.

Wireless telecommunication systems were originally designed for voice only data. The first non-voice service to be offered was SMS over GSM networks. Soon after SMS, other data services such as web browsing, file downloading and media services were made available. These services have not enjoyed the same success on wireless networks as they have on the wired Internet; this is due to certain features of wireless networks that hinder data services. One such feature is that of scarce air interface bandwidth [Chan, 2006]. 3G offers air interface rates of up to 2Mbps, but all users need to share this air interface bandwidth. The result is that each user is assigned a small bandwidth.

2.1 SMS – The Instant Messaging Service

SMS is an instant messaging service and the first data service to be offered over a cellular network. Originally SMS was a GSM network service but is now offered as a standard service on all cellular technologies and devices. The SMS protocol uses the control channels to transfer its data. The use of the control channels limits SMS messages to a maximum of 160 characters, making it fitting for notification and short messages. Although it is ineffective at sending large amounts of information between users it is able to send fast reliable messages. A SMS may be received while a mobile device is in the process of a voice call or data connection. SMS is well suited for instant reliable messaging [ADC NewNet, Inc. 1999] and [Enck, 2005].

2.1.1 SMS as a Notification Service

Several experiments have been done measuring various aspects of communication performance of SMS. [Koumpis et al, 1999] investigates the time taken to send SMS notifications to a mobile user. Researchers accomplish this by implementing a SMS notification application and enabling it to connect directly to the SMSC component of 4 commercial cellular service providers. The implemented application is able to send notification SMS messages when an email arrives or a voice message is left.

When an email arrives the application constructs a SMS message containing details about the received email. This SMS is then forwarded to the relevant SMSC via the gateway to be delivered to the user. SMS transfers are then recorded. Findings show that the time taken to transmit a SMS ranged from 2-16 seconds. This only refers to the time to transfer a SMS once a connection to the target SMSC has been established. The time taken to establish a connection to the target SMSC generally took longer than the time taken to transfer the SMS message. Different service providers had different bandwidths and network conditions. As a result different transfer times were obtained by the 4 service providers.

Service providers' available bandwidth ranged from 2400-9600bps. All message sizes were below 160 characters. It appeared that the available bandwidth had an effect on the time taken to transfer the SMS. But researchers found that service providers with the same bandwidth had a difference in transfer time. The researchers believed that network conditions and queuing time were responsible for the large variance in transfer time. The best performing SMS takes a mere 2 seconds over a low 9600bps connection. The worst performing transfer still took only 16 seconds indicating that SMS is good for urgent short messages. A scenario where this has been illustrated is the use of SMS to send commands to a house alarm system.

2.1.2 Sending Commands via SMS

A house alarm system commanded via SMS. [Seixas and Palma, 2006] built a programmable microcontroller which connected to a house alarm system and was able to issue commands to the house alarm as well as receive feed back from the house alarm sensors. This microcontroller was then connected to a mobile cellular phone enabling it to relay information from the house alarm across a cellular network to a target mobile phone. The result is that a user will be able to monitor and control the alarm from anywhere in the cellular network.

The application has a finite number of input commands and output messages. This makes it well suited for the SMS service that has a limited message size that it can transfer. The time taken to send a SMS command or message to and from the house alarm takes the same time as that of a normal SMS from one subscriber to another, indicating the good speed and reliability aspect of the application.

2.1.3 Limitation of the SMS Protocol

In the course of their research [Ferng and Tsai, 2003] showed that SMS is not suited to send large amounts of bursty data. This is because of the nature of the SMS protocol. The SMS protocol has a payload of 140 bytes and establishes a circuit switched connection to the target device each time a packet has to be sent. The research mentions the long delay needed to setup a circuit switch connection before a SMS message can be transferred and conclude that a packet switched protocol will be more suited to transfer large amounts of data. TCP is more appropriate for the transferring of large amount of data as a connection session can be established between source and destination. This connection will only be set up once and all packets can be sent across it. SMS message transfer and voice call message setup share the same control signals and in a loaded network affect each other.

In addition to SMS size affecting performance the network load also plays a role. [Agarwal et al, 2004] studied how long it would take an SMS to be transferred and what would the chance be of the SMS being blocked due to no available Stand-alone Dedicated Control Channel (SDCCH) channels. The control channel over which SMS is sent is used by the location update service, voice call setup service and data connection setup service. It mentions that setup takes about 4 seconds for voice calls before the channel is released. SMS has similar setup times.

This implies that if all SDCCH were in use for a voice call or data connection setup, then no SMS messages will be able to be sent during that time. The message will be returned to the SMSC and delivery attempted at a later time. When exactly the next delivery attempt will occur is dependent on the algorithm the service provider uses.

A formula to determine the time a message will take to be transferred was derived. The formula is 9 x [number of characters in the SMS] = transfer time in milliseconds. This means that a 160 character message will take 1440ms to be transferred. It is presumed that this is the time from the BS to the MS. The formula is based on the size and frame rates of a GSM network.

2.2 FTP Applications

FTP, MMS, and email applications are similar in nature; all three applications transfer a file from a source to a destination. These were some of the first data applications to follow after SMS, since these applications are not time constrained, unlike video/audio data. A poor performing network can still execute these applications successfully, the only cost being a longer waiting period for the file to be transferred. Past research only investigates FTP applications.

2.2.1 Real and Theoretical Rates

[The Shosteck Group, 2001] Produced a white paper in 2001 where it stated the theoretical performance of GPRS and 3G networks, followed by actual performance in real networks. 3G is quoted as being 384kbps and above while GPRS is quoted as having an ideal rate of 115kbps. Tests were then done in a real network and the obtained rates were compared to that of the ideal rate.

SK Telecom of Korea launched a CDMA2000X1 (Code Division Multiple Access) network in 2001 which was capable of an ideal rate of 120kbps. The actual data rates were 70-90kbps. This actual rate is closer to ideal than would be expected, because only 1.5% of SK Telecom subscribers were 3G enabled. As this percentage grows rates are expected drop and be closer to the 30-50kbps mark.

Other GPRS providers expect rates from the 20kbps-40kbps mark. As the number of users of a network increases the available bandwidth per user decreases, since the bandwidth is shared amongst users. Network conditions such as strength of signal and noise also affect a user's throughput. It was found that under bad network conditions rates as low as 8kbps were experienced by subscribers of BT Cellnet. This indicates both the under performance of theoretical values and the affects of network conditions on performance.

2.2.2 MP3 Downloads over Wireless Networks

The time to download a MP3 file was investigated by [Moltchanov et al, 2002]. This was done with the help of a Nokia mobile phone connected to a commercial GPRS network. The GPRS network had an ideal rate of 39.6Kbps. The mobile phone connected to the host MP3 server and downloaded the MP3s across the wireless GPRS network. The time to setup a connection and download the MP3 was recorded.

A three way hand shake took 854.63ms, while a 3Mb MP3 file took 820 seconds to download. The throughput obtained by the MP3 download was 36Kbps, which is extremely good, showing high throughput efficiency. Also various movement patterns were studied and it was found that throughput changed insignificantly. The slow air interface of GPRS is pointed out as being the main bottleneck of the system. Authors expect the WCDMA air interface to improve the performance of wireless networks as it would eliminate the slow GPRS air interface.

2.2.3 The Effects of Delay on MP3 Downloads

Researchers in Bologna Italy developed a wireless MP3 distribution system; see [Roccetti et al, 2005]. The time to download various size MP3s at different background traffic conditions and different end-to-end network delays were measured. The wired network contained replicated servers situated at dispersed geographical regions, Finland, New Zealand, USA and

Japan. A user connected to the Bologna server via a UMTS network had access to the other servers through a gateway implemented. This enabled measuring of download times for each replicated server.

A server's physical distance affected the end-to-end delay between the mobile device and the sending server. It was observed that an increase in delay resulted in an increase in downloading time. This increase occurred on the wired network and was deemed negligible, because the bulk of the time taken to transfer a file occurred across the air interface.

To do the actual download experiments a UMTS simulator was connected to the earlier developed MP3 distribution system. This UMTS simulator was then used to simulate user downloads. Users could either download a single MP3 or a MP3 compilation set. Single MP3 sizes ranged from 3Mb-5Mb, while compilation sets never exceed 50Mb. This research was done while varying the background traffic noise and the speed at which the subscriber moves. The best results reported at the lowest background traffic noise and at stationary movement were 250s for a 3Mb download and about 420s for 5Mb download. The simulator achieved a download rate of 96Kbps. The ideal rate is not stated.

2.2.4 The Effects of File Size on FTP

In an FTP transfer as the file to be transferred increases in size the throughput of downloading the file also increases, as shown by [Chakravorty et al, 2004(A)]. A mobile device was connected to the Internet via a cellular network. A range of files of different sizes was then downloaded over GPRS and 3G networks. The throughput obtained by GPRS and 3G for different file sizes were recorded and then compared to each other.

As the downloaded file became larger both GPRS and 3G obtained a higher throughput for downloading the file. This increase in throughput can be explained by the connection setup using a smaller percentage of total time taken to download the file as the file size increases. Although it is expected that TCP window size growth plays a more important role. The larger the file to be downloaded the more time TCP has to grow the window size to the optimal size.

The ideal rate of devices used in experiments was 39.6Kbps for GPRS devices and 384Kbps for 3G devices. Files downloaded during experiments ranged from 1KB-300KB. At small file sizes it was seen that GPRS and 3G had a similar throughput value. But as the size of the file to be downloaded increased GPRS reached its limit and levelled out, whereas 3G throughput continued to increase. At file size 50KB GPRS approaches its limit and achieved a data rate of 29.7Kbps. When file size is increased to 300KB GPRS's throughput does not change significantly achieving a rate of 30.2Kbps. This indicates that GPRS had reached its maximum throughput. On the other hand 3G showed no signs of levelling out at file size 300KB indicating that 3G had not yet achieved its maximum throughput.

[UK Telematics online, no date] made a comparison between GPRS download times to that of 3G download times and stated the following for a 500KB file download, GPRS takes 120 seconds to download the file, while 3G is stated as taking 10 seconds to download the file. From this it suggests that at the 500KB file size GPRS and 3G have very different performance values. TCP window size has finally grown large enough for 3G to exploit, allowing 3G to perform significantly better than GPRS. TCP window size and TCP congestion control dictates the speed at which file transfers takes place.

2.3 The Effects of TCP on Wireless Networks

The size of the file to be downloaded is not the only factor that affects TCP performance. Network conditions and noise also affect TCP performance. TCP was designed for wired networks which have short delays and low error rates. Any delay experienced by TCP in the network is put down as congestion. The congestion control mechanism of TCP then shrinks the window size according to the congestion experienced. Wireless networks delays and noise vary constantly with distance from the BS. This affects TCP causing it to perform poorly across wireless networks.

It was found that the sensitive and chaotic response of TCP in a wireless network caused unfair sharing of resources [Gurtov and Floyd, 2004]. Increasing the number of concurrent users increases the delay experienced in the network [The Shosteck Group, 2001]. This increase in delay triggers TCP's congestion control mechanism, resulting in TCP throttling throughput. Users sharing the same air interface had large differences in throughput, as much as an 80Kbps difference between users. This indicates the importance of noise and network conditions in the performance of TCP. In a study done on the performance of different TCP implementations it was found that TCP Vegas showed more equity between connections [Dubois, 2005].

2.4 Web Browsing Service

The times to open the home pages of four popular web sites were measured over GPRS and 3G networks. This was done as part of a benchmarking process done by [Chakravorty et al, 2004(A)]. The four websites used were Mail, Yahoo, Amazon and CNN. The throughputs achieved by opening the home page of each of these websites were recorded for GPRS and 3G, while the time taken to open was measured for GPRS only. Webpage contents are normally scattered across multiple servers.

The contents of the web pages concerned were scattered over three to six servers and contained various numbers of embedded objects. These embedded objects affect the throughput of opening a webpage as the HTTP protocol does not get all the embedded objects simultaneously. A separate GET command needs to be issued for each embedded object and after the object is successfully downloaded a new GET command is issued for the next object. This process is inefficient and increases the time taken to open a webpage and thus decreases the throughput obtained opening a webpage.

The ideal connection rate of GPRS and 3G were 39.6Kbps and 384Kbps respectively. The higher bandwidth of 3G allowed it to achieve a higher throughput for all pages opened. As the size of the web page increased it took longer to open the webpage, but throughput did not increase. This suggests that there is no relationship between webpage size and throughput when opening a webpage. It is suspected that the slow GET embedded object process and the larger number of embedded objects in larger web pages decrease the throughput obtainable by large web pages. Various optimisations exist to improve this drawback, such as pipelining, parallel connections and pre-packaged web pages.

2.4.2 HTTP Pipelining

HTTP pipelining allows a web browser to request multiple embedded objects simultaneously over a single connection. Pipelining decreases the waiting period between getting individual embedded objects thus decreasing the time taken to download the webpage. Server state also affected the time taken to download a webpage, as found in a study done by [Timm-Giel, 2004]. It was found that a busy server affected the downloading time of a webpage more than that of pipelining. The busier the server the lower the throughput obtained when downloading a webpage.

If a server's state is relatively consistent then other factors such as pipelining could affect throughput. However some browsers showed the same performance irrespective of whether pipelining was used or not. This indicates that pipelining may not be the most important factor when downloading a webpage. The results for UMTS experiments were ~118Kbps with ideal channel rate 128Kbps. These tests were done on a real network before the network went commercial and thus had very few users, for this reason obtained rates are close to that of ideal rates.

2.5 FTP Outperforms HTTP

Studies have shown that over the same connection FTP transfers normally outperform that of web browsing. This is because a FTP file has different characteristics to that of a web page. A FTP file is a single object that resides at a single server, whereas a web page usually consists of multiple imbedded objects residing on different servers. In the case of FTP file and a web page download throughput was 85-125Kps and 41Kps respectively for a 144Kps ideal connection. This difference is because of the start stop nature of HTTP and the characteristics of a webpage.

A browser sends a separate GET request to download each of the embedded objects in a webpage. These embedded objects making up a webpage can be located on multiple servers. A GET request is sent for an object and the object is completely downloaded before a GET request is sent for the next object. This procedure is not good for high latency connections such as wireless cellular networks. Researchers found the main Yahoo page of size 60.3KB took 12 seconds to download [Chakravorty et al, 2004(A)].

2.6 Comparison of TCP and UDP

Timm-Giel (2004) performed experiments that investigated critical factors affecting application performance in a wireless network and attempted to fine tune them. Feasibility tests, in which an emergency application service ran over a GPRS/UMTS network, were done to check if the application could perform well enough in a wireless environment. Measurements were then done for applications running over TCP, UDP and HTTP protocols. It was found that the asymmetrical link of wireless networks had an affect on the performance. This caused mobile devices used in testing to have a lower uploading rate than downloading rate. Hence a service that uploads data from a mobile device will perform weaker than a service downloading data to a mobile device.

The throughput for downloading a 100KB file was 89Kbps whereas the uploading of 100Kb file was 39Kbps. This proves that mobile devices are asymmetric and perform better at downloading services than uploading services. Results also show that the 100KB file size obtains a higher throughput than that of a 400Kb file size for downloading experiments. Potentially this was due to network conditions as a real network was used. Other experiments performed showed FTP (114Kbps) outperformed that of UDP (105Kbps) for stationary test. The mobile device used in GPRS experiments had a theoretical download speed of 54Kbp, but achieved a maximum rate of 28Kbps for FTP download experiments.

These experiments were done on GPRS and 3G UMTS networks. UMTS far outperformed that of GPRS. Researchers concluded that all emergency services including video services will be possible with UMTS network technology while only a sub set of services can be realised using GPRS [Timm-Giel, 2004].

2.7 The Effect of Channel Allocation on Performance

A UMTS network has many factors governing channel allocation, such as available transmitting power and signal to noise ratio. Base stations have limited power to transmit radio signals. Each mobile device connected to the BS requires a portion of the BS available transmitting power. If a user moves further away from a BS the channel assigned to that user may be surrendered to another user who is closer to the BS. This is because it takes more power to transmit data over a further distance. Constant movement of users and changing of network conditions cause channel switching to occur frequently in a UMTS network.

Investigating the throughput of a GPRS network [Brasche and Walke, 1997] found that it achieved a throughput of ~11Kbps. This throughput remained the same for ideal rates of 15-50kbps. This demonstrated that an increase in assigned bandwidth did not necessarily mean an increase in data throughput.

2.8 Media Streaming in Wireless Networks

Mobile devices have small screens which hide the effects of image resolution and data rates. Researchers in [Song, 2002] examined how frame rate, resolution and data rate affected a user's perceived quality of the video. It turned out that because of the small size of mobile screens users were unable to notice significant changes in video quality when the image resolutions were increased. The study showed that frame rate was the most influential factor when displaying video on small mobile screens, with bit rate being second most important factor. Bit rates of 105Kbps and 200Kbps were proposed by [Ries et al, 2007] for screen sizes QCIF (Quarter Common Intermediate Format) and CIF (Common Intermediate Format) respectively. These are common sizes of mobile screens.

Authors of [Snap9 Corporation, 2006] did a study on streaming in the mobile environment in which they concluded that 9% of data throughput will be used by protocols required for streaming. Taking this into account, expected values of a GPRS network in which a device occupies four download slots will be able to achieve a streaming rate of 36Kbps. An EDGE network using two slots and a 64Kbps UMTS bearer channel will both realise streaming rates of 50Kbps. Higher streaming rates can only be attained by higher bearer channels of UMTS networks. The implications are that only a UMTS network will be able to obtain the required 105-200Kbps recommended to stream video to mobile devices.

2.8.1 Streaming Rate and the Channel Rate

Streaming of real time audio was studied by [Moltchanov et al, 2002]. A reliable internet radio station was used which had the streaming rates of 24Kbps, 64Kbps and 128Kbps. A PC was connected to a GPRS phone, which enabled the PC to connect to the Internet. The GPRS connection had an ideal rate of 36Kbps. Experiments were then carried out on the three streaming rates offered by the Internet radio station.

At the start of the streaming session a 10 second buffer was downloaded. It was found that the 24Kbps streaming rate played back smoothly without the need for re-buffering. But constant re-buffering was needed for 64Kbps and 128Kbps streaming rates, because these streaming rates are higher than the throughput achievable by the GPRS connection. The conclusion is that a faster air interface such as WCDMA would allow for smooth streaming of the 64Kbps and 128Kbps Internet radio channels.

[Weber, 2006] performed tests in a live unloaded UMTS network using a media streaming server, streaming video at 104Kbps, to determine the throughput achieved by 64Kbps, 128Kbps and 384Kbps bearer channels. Mobile devices could be mid-cell or on the outskirts of the cell. Media was streamed using RTP protocol over the UDP protocol. A mobile device connected to the streaming server using one of the three bearer rates and viewed a 5 minute video clip; the re-buffering results of viewing the video clip were recorded.

The average throughput users mid-cell experienced was 96.2Kbps while users located at far edge of the cell experienced 89.7Kbps. This was a live network and channel switching occurred during the viewing of the video clip. The likelihood of a user being switched to a lower data rate channel increases with a user’s distance from the BS. When a user moves to the edge of a cell, the BS may not have the available power to continue transmitting at the current data rate. The BS may then choose to transmit data to the user at a lower data rate. This can cause re-buffering to occur if the channel rate is less than the streaming rate.

The channel rate of 64Kbps was the only channel that caused re-buffering, because it was the only channel rate lower than the server streaming rate. The same was noticed by other researchers when the bearer rate was lower than the server streaming rate [Moltchanov et al, 2002]. To alleviate this problem data is buffered before playback starts. The lower the bearer data rate the longer this initial buffer used before playback starts [Lo et al, 2006].

2.8.2 Buffering in Media Streaming

In an attempt to stream MP3 audio smoothly [Roccetti et al, 2005] calculated the buffer time needed before playback could start. The maximum downloading speed obtained by the system was 96Kbps, while the MP3s were encoded at 128Kbps. This meant that buffering at the start would always need to take place. The buffer time depended on the playing time of the media being downloaded.

The user requested all the songs they wanted to listen to and the system calculated how much data to buffer before initiating playback. If 3 songs having a total playing time of 1263s were chosen, a user would need to wait for 543s of buffering to take place before the music could start playback.

As the playback rate surpassed the channel rate the required buffer increased exponentially for both GPRS and UMTS networks [Chesterfield et al, 2004]. The main draw back to streaming occurs for interactive video where buffers can not be used, in this case the long end to end delay is noticeable especially in GPRS which has end to end delays in the region of 850ms and UMTS around 30ms-350ms. [Basso et al, 2002] identified the ARQ (Automatic Repeat Request) link layer as being the main element affecting the video quality received by the mobile device. It is advised that ARQ take place when streaming media, as this will remarkably increase the video quality. Although a point is made that too many retransmissions can negatively affect quality and a balance needs to be found.

2.8.3 Assigned Dedicated Data Channel

Handover (when a user moves from one cell to another the call of the user is transferred to the new cell) could cause a user's performance to decrease or cause a user to lose their call. [Ferng and Tsai, 2003] considers how resources are allocated to voice and data channels depending on priority and the effects of handovers. Based on the rules the network uses, it is possible for voice calls to get priority over data connections. The result would be when a new voice call enters a cell and there are insufficient resources to facilitate the voice call; but because of the priority rules of the network, resources supporting an existing data connection may be removed and assigned to the new voice call. Such reorganisation of network resources occurs often in 3G networks and assigned data channel rates constantly change affecting applications performance.

Present research investigated the performance of SMS, MMS, Email, FTP, media streaming, media broadcasting and video calling applications over a 3G UMTS network. In particular how network factors such as bandwidth, delay and size of data transferred affected these applications performance. Having had all the applications simulated under the same network conditions helped in comparing them. This indicated which network factors affected which applications most. Simulations were done under realistic network conditions and should be of help when predicting an application performance in a cellular network.

Chapter 3

UMTS/WCDMA Overview

3.1 Overview

Wireless networks face an ever increasing number of users and an increasing number of bandwidth intensive applications. 3G was developed as a solution to this problem. As early as 1992 the telecommunication industry met to discuss the need for a faster technology to cope with the ever increasing wireless traffic [Son Nguyen, 2005]. 3GPP was formed and became the official third generation standardisation body. A call for wireless technology proposals was made. UMTS was proposed by a few regional standardisation bodies and offered data speeds up to 2Mbits a second. In 1999 3GPP accepted UMTS as an official 3G standard and WCDMA as its air-interface technology. 3GPP modifies 3G specification periodically, in 3GPP release 4; modification to the RAN (Radio Access Network) was done. In release 5, HSDPA technology was added, increasing support for packet switching architecture and IP addressing protocol [Hamalainen, 2003] and [Son Nguyen, 2005].

During the time UMTS was being developed in the late nineteen nineties, GSM was the dominant architecture of the time. A conscious effort was made to base UMTS on GSM so as to make it backwards compatible with GSM. This allowed service providers to retain the large investments made in GSM. UE’s radio equipment specification allowed it to communicate with both UMTS and GSM radio interfaces [Jain, 2005] and [UMTS World, no date].

The architecture of UMTS has 3 domains, UE (user equipment) the mobile handset used to send and receive messages, UTRAN (UMTS Terrestrial Radio Access Network) provides the air interface access method to the mobile device and the CN (Core Network) that routes messages and connects UMTS to other networks. Below each component will be explored in more detail.

UE consists of two components. First is the USIM (Universal Subscriber Identification Module), UMTS subscriber identity module holds the identity and service profile of the user. Second is the radio equipment consisting of a transmitter and a receiver, which is used to communicate with the UTRAN via the Uu interface [Wang and Prasad, 2005] and [Ojala, 2000].

3.2 Air Interface

The UTRAN air interface technology is WCDMA. WCDMA is a direct sequence technology that uses spreading codes to allow data rates up to 2Mbits/s. The two types of WCDMA multiple access method used are FDD (Frequency Division Duplex) and TDD (Time Division Duplex) [Son Nguyen, 2005]. The UTRAN consists of a RNC (Radio Network Controller) and a Node B. A Node B handles the communication to and from all UEs in one or more cells. Its responsibility includes the encoding and decoding of radio signals, forward error correction (FEC), closed loop power control, and RRM (Radio Resource Management) amongst other functions. A Node B is connected to RNC via a lub interface. One or more Node B is controlled by a RNC as well as controlling the radio resources of a cell. There are many functional requirements of the RNC, which include radio resource control, admission control, channel allocation control, broadcast signalling, and open loop power control. The RNC provides access to the CN via its lu interface connection.

3.3 UMTS Core Network and Network Elements

The CN resembles that of a GSM network with GPRS [Tektronix Inc, 2004], [Knutsson, 2004] and [TeliaSonera, 2004]. There are two domains in the CN, the circuit switched domain and the packet switched domain. Elements of the CS are Mobile Switching Centre (MSC); Visitor Location Register (VLR); and Gateway Mobile Switching Centre (GMSC). Packet Domain has the following elements, Serving GPRS (3G) Support Node (SGSN); Gateway GPRS (3G) Support Node (GGSN). Then the Equipment Identification Register (EIR); Home Location Register (HLR); and Authentication Centre (AUC) are shared across both domains. Next each element is looked at in more detail.

MSC and GMSC: The MSC is the central element of the circuit switch domain and supports in the circuit switch connection process. It also handles a user's mobility and deals in the handover process that happens between cells. The GMSC is a special MSC that provides a gateway to external networks such as ISDN (Integrated Services Digital Network) networks.

HLR and VLR: The HLR is a database located in the user's home network and holds the user's profile information; such as allowed services, roaming areas, and authentication keys. The VLR is a similar database and contains information of visiting users. When a user switches locations, the VLR is immediately updated. This is done by contacting a user's HLR and transferring all the relevant information to the current network's VLR.

SGSN and GGSN: The main function of the SGSN is to route packets in the packet switching domain. SGSN also manages authentication and current position of a user. The GGSN serves as a gateway to other packet switching networks such as the Internet. As of 3GPP release 5 the main routing protocol will be IP protocol. Routing does not happen directly via IP routing protocol but through GPRS Tunnelling Protocol (GTP).

3.4 Interfaces

All the elements of a UMTS network are linked together by interfaces. Only the interfaces relevant to the connection of elements discussed will be focused on.

Uu: Is the radio interface that exists between the UE and the UTRAN.

luB: Is the interface that connects Node B to RNC.

lu: Is the interfaces that connects UTRAN to CN. There are two types of lu interfaces, luPS and luCS, for the packet domain and the circuit domain respectively.

Gn: Is the interface linking SGSN to GGSN.

Gi: Is the interface that links GGSN to external packet switching network like the Internet.

3.5 UMTS Protocols

Only the protocols of the interfaces discussed above will be looked at. The lu and Uu interfaces have two sets of protocols that exist, user plane protocols and control plane protocols. The user plane protocols consist of a layer protocol structure that facilitates flow control and transfer of information. See Figure 3.1. In addition it provides error handling functionality such as detection, correction and recovery of transferred information. The control plane exists to support and control the user plane functions. This includes requesting the service, controlling different transmission resources and handovers [Wang and Prasad, 2005] and [Tektronix Inc, 2004].

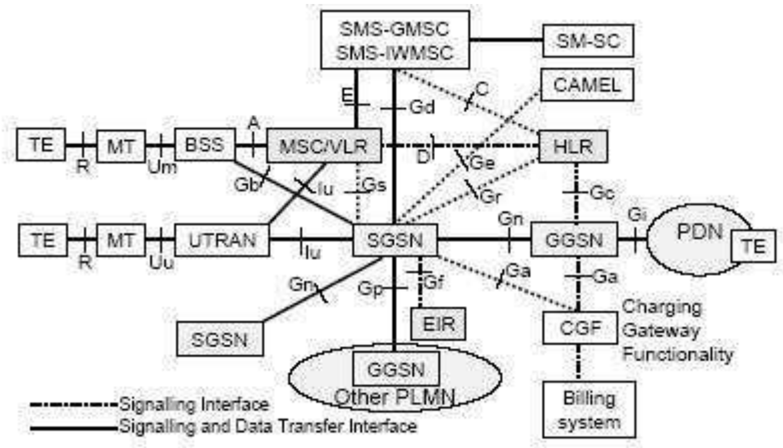

Figure 3.1 Interfaces and protocols of a UMTS network. Source: Anthony Chan EEE401S course notes.

3.5.1 User Plane Protocols

User plane protocols between the UE and UTRAN are PDCP, RLC, MAC and L1; while protocols between UTRAN, SGSN, and GGSN are GTP-U and UDP/IP.

Packet Data Convergence Protocol (PDCP): PDCP offers transparency to higher level protocols by mapping higher level protocol characteristics to the radio interface. This allows high level protocols to be added, removed and modified without affecting the radio interface. Point-to-point (PPP), IPv4 and IPv6 are all supported by PDCP.

Radio Link Control (RLC): RLC provides logical radio interface links; one or more of these links can be assigned to a mobile device. Each link is assigned an identity number to help control and manage it. Defined in 3GPP TS 25.322

Medium Access Control (MAC): MAC is the protocol procedure used to access the radio interface. Defined in 3GPP TS 25.321

Physical Layer (PHY or L1): Offers information transfer services to the MAC and higher levels.

GPRS Tunnelling Protocol for the user plane (GTP-U): GTP-U encapsulates all the PDP (Protocol Data Units) and tunnels data between UTRAN, SGSN and GGSN. Defined in 3GPP 29.060

User Datagram Protocol / Internet Protocol (UDP/IP): Used for routing data and control signal on the backbone of the network.

3.5.2 UMTS Control Plane

User Mobility Management and Session Management (GMM/SM): GMM has the responsibility of attaching, detaching, security and area routing updates.
SM controls the PDP activation and deactivation.

Short Message Service (SMS): SMS protocol is responsible for SMS message origination and termination at the mobile handset.

Radio Resource Control (RRC): RRC is the signalling protocol for controlling and configuring radio resources.

Radio Access Network and Application Protocol (RANAP): RANAP encapsulates and carrier’s higher layer signalling protocols. It is responsible for signalling between the UTRAN and the SGSN, and manages the GTP connection on the lu interface.

RLC, MAC and L1 have both user plane and control plane functionality. The same protocols and interfaces are used, in the one user data is sent and the other control signals.

All the entities and protocols above will have an effect on the performance of a UMTS network. These entities' and protocols' behaviour are implemented in the simulation tool used. Networks created will thus have all these behaviours allowing for a more accurate analysis of network performance.

3.6 Wideband Code Division Multiple Access - WCDMA

WCDMA is the UTRAN air interface technology. Technology in the air interface has progressed a great deal in recent years, but most of the factors limiting bandwidth, throughput and capacity still occur here. It is these improvements that have enabled 3G UMTS networks to offer new services such as broadcast TV, video calling and high speed web browsing. Next the features of WCDMA which enable higher throughput and larger capacities are investigated [Son Nguyen, 2005] and [Wang and Prasad, 2005].

WCDMA is a direct sequence CDMA system on a wide 5MHz radio frequency band. WCDMA comes in two modes, FDD (Frequency Division Duplex) and TDD (Time Division Duplex). In FDD there exists a separate radio spectrum for upload and download. In TDD the same radio spectrum is used for both the upload and download on an alternating basis. FDD is the mode used in present research.

All users of a WCDMA network within a cell use the same radio frequency at the same time. The entire available bandwidth is used by each code channel. To uniquely identify each channel the scrambling code and the spread factor code of the channel is used. Adjacent cells also reuse the same radio frequency and spread factor codes, giving WCDMA a reuse factor of one [Son Nguyen, 2005], [Tektronix Inc, 2004], [Ojala, 2000] and [Akl and Son Nguyen, 2006]. Scrambling codes are used to differentiate which base station a signal is received from.

Code channel separation is accomplished by digitally encoding individual channels, not by frequency separation. A mobile device looks for the unique code assigned to it and the rest of the channels are undistinguishable from noise. Each channel is uniquely identified by the carrier frequency and code. This allows two or more WCDMA service providers to operate in the same area.

Below it can be seen how addition of scrambling codes and spreading occur to allow multiple channels to transmit over the same radio frequency simultaneously.

WCDMA specification allows 3.84MHz bandwidth for signal bandwidth.

> A 9.6kbps channel is modulated over a10kHz radio frequency spectrum.
>
> The data is then spread using 3.84Mbps code rate. This is the fix chip rate of WCDMA and the results are the same as transmitting over a 3.84MHz frequency spectrum.
>
> The mobile device will receive this spread signal along with noise, interference and other channel signal data.
> The mobile device reapplies the process to uncover the original signal intended for it [Tektronix Inc, 2004].

WCDMA has variable length spreading codes ranging from 4-512, the length is known as the spread factor. Each channel is uniquely identified by a combination of scrambling code and an OVSF (Orthogonal Variable Spreading Factor) code. The shorter the spread factor the fewer simultaneous channels but each channel has a higher data rate. The higher the spread factor the larger the number of simultaneous channels but each channel will have a lower data rate. The spread factor and the resultant channel data rate have an inverse relationship [Son Nguyen, 2005].

The scrambling code is mixed before data is transmitted and allows for the identification of the device. Each base station is identified by a unique scrambling code. A base station can transmit multiple channels to multiple devices simultaneously. Each DCH is first multiplied by an OVSF code. OVSF are orthogonal codes and is used to separate traffic. Synchronisation channels P-SCH and S-SCH do not get multiplied by OVSF codes. A mobile device which tries to decode a signal using the wrong OSVF code will interpret the signal as noise. OVSF codes can be reused by each mobile station and mobile devices in an area because each base station and mobile device is uniquely identified by its scrambling. Scrambling codes are not orthogonal and can be a source of interference.

WCDMA was designed to allow many users to efficiently share a single radio frequency by dynamically reassigning spread factor codes, which may be updated every 10ms. A user may only have one OVSF code for a voice call, but may have one or more OVSF codes for data connections.

Table 3.1 Parameters of a WCDMA-FDD air-interface.

Parameter Name	Parameter Value
UTRA Mode	FDD
Multiple Access Scheme	WCDMA
Carrier Spacing	4.4-5.2MHz
Chip Rate	3.84Mcps
Spreading Factor Range	4-512
Modulation	QPSK
Pulse Shaping	Root raised cosine, roll-off = 0.22
Frame Length	10ms
Time Slots Per Frame	15

3G UMTS has brought many improvements to wireless communication but the wireless interface still remains the main bottleneck in the system. The bandwidth is limited and the interface itself has a high delay, our research investigates application performance across different network conditions focusing on changes in delay and bandwidth.

Chapter 4

Application Summary

A few of the applications expected to be popular in 3G are FTP, MMS, SMS and email. These applications share the same basic principle. These applications are examined separately, indicating the process of the application and the factors which affect it. Following this discussion is a description of HTTP web browsing and media streaming applications.

4.1 FTP Overview

File Transfer Protocol (FTP) is the protocol used to transfer a file from one device to another across a network. For the purpose of this research mobile FTP is defined as the transfer of a file across a UMTS network, where the receiver, sender or both are mobile devices.

4.1.1 FTP Downloading

The FTP service is most likely to be used for downloading of files from an Internet server. Uploading from a mobile device will occur less frequently. The benefits of downloading are that a mobile device is usually capable of downloading data at a higher rate than uploading, and less power consumption takes place for receiving data than transmitting data. These factors will usually allow downloading services to outperform uploading services. The trend of commercial FTP service to mobile devices looks to continue to grow as the downloading of MP3s, video clips, and wallpapers are large revenue earners.

4.1.2 Downloading Process

The downloading process is as follows: A file exists on a server and a mobile device connected to a UMTS network indicates its intension to download the target file from the server. A connection is setup between the source server and the mobile device, and the file is then transferred to the mobile device. At the end of the data transfer the connection between the server and destination is closed and a copy of the file now exists on the mobile device.

The performance of a file transfer across a wireless link is lower than across wired networks. It is unlikely that the transfer will fail to take place although this is possible if a connection is lost and not able to be recovered in time. It is more likely that due to time constraints a user will choose not to download files over a certain size across a wireless network or if the network is busy a user will opt not to use the service. These factors that compel the user not to use the FTP service are performance factors. Reasons for poor performance of wireless networks compared to wired networks are investigated.

4.1.3 Factors Affecting FTP Performance

Wireless links have low bandwidth and high delays, it is also more error prone than wired networks. The low bandwidth means less data can be sent at a time across the wireless link, while high delays indicates that it takes longer for the data to be sent across the wireless link. Both these factors obviously lend itself to a more constrained connection, making the wireless interface the constraining link between the source and the destination. The use of TCP (Transport Control Protocol) as the protocol over which FTP is executed further affects the performance as TCP interprets all delays in the connection as congestion in the network and throttles the source by sending data at a slower rate [Dubois, 2005].

In the earlier days of wireless networks the air interface was notorious for its high error rates but since the introduction of link layer retransmissions the error rates have dropped to more acceptable levels of less than 1% [Chakravorty et al, 2004(A)]. However the retransmission by the link layer are seen by higher layers as delays and congestion by TCP.

4.1.4 Factors Affecting Uploading

The uploading of files from a mobile device suffers from all the side effects of downloading, as well as a mobile device most likely having a lower maximum upload data rate than download rate. More power is required to transfer data at higher data rates, the further data needs to be sent across the air the more power it needs. Therefore a mobile device may be in a situation where a base station is too far for it to send data at its maximum data rate due to power constraints; resulting in a lower transfer rate being chosen [The Shosteck Group, 2001].

4.2 Email

4.2.1 Email Overview

Mobile email is defined for the purpose of this research as email that is either sent from and / or received by a mobile device. Email has been a popular internet service since the start of the Internet and actually pre-dates the Internet. Mobile email works much the same as normal wired email, except previously users only had access to their email from a personal computer or laptop, now users are able to access email from PDAs (Personal Digital Assistant), smart phones and later model mobile phones. Email is an asynchronous store and forward service, a user sends an email which is then stored on a mail server and forwarded to the receiver when the receiver is ready to view it. To send an email a user needs an email user agent, a network connection, and an email delivery agent [Isode, no date] and [Mehra, 2008].

4.2.2 Types of Email

Two types of email user agents exist, a standalone email client and a webmail email service. The standalone client can be installed on a mobile phone either by the manufacturer or by the user. These standalone clients are the same as applications such as Microsoft Outlook, Lotus Notes or Novel GroupWise. The other type of email, webmail service, is accessed via an Internet browser and a webpage which contains an email user agent. The web email interface is setup by the providers of the email delivery agent and simply allows the mail server to be accessed via a webpage, both email clients has its benefits. A user is able to access webmail from any computer, making it a highly available and accessible service. An email client application is generally considered more secure and connects to an email server that is usually owned by a corporate entity behind a secure network. An email client also has the option to store its email locally or to have the server store its mail for it. Other benefits of the email client are its ability to synchronise and integrate with other desktop and network applications running on the local machine and network. In the mobile case, the benefits will be the synchronisation and integration of the email client with other applications on the mobile device or network [bizcommunity.com, 2008]. These benefits and security features make email clients more popular for the corporate environment while webmail tends to be more popular for personal use.

4.2.3 Email Process

The email user agent is used to construct an email in the standard email format consisting of a header and body. The header contains details of who the email is to, from, subject and if the email contains any attachments. Any type of file may be attached to an email, which includes document, image, video and application specific files. To help MUA identify which type of file has been sent the Multipurpose Internet Mail Extension (MIME) is used to encode and decode attachments [Crocker, 1982], [Klensin, 2001] and [Marshall and Crosby, 2007].

Once the email is constructed it is sent to the MDA using the Simple Mail Transfer Protocol (SMTP). In a mobile medium the email will be sent from the mobile device across the wireless network to the MDA that exists somewhere on the internet. The MDA then checks the destination address of the email and queries a Domain Name Server (DNS) and then routes the message to the destination MDA.

At this stage many actions could happen. If the email address does not exist on the MDA or the inbox is full, a Daemon process automatically bounces a reply message to the sender to inform it of the error. If the address does exist then the destination MDA stores the message in the MUA's inbox. Once the message is stored the MUA may use POP (Post Office Protocol) to poll the MDA periodically to enquire if any new email has arrived. Or the MDA could be setup to use Internet IMAP (Message Access Protocol), which causes the IMAP either to poll; or a push message may be sent to the MUA. The push message sent can be a simple message informing the MUA that the following email with subject from source has arrived or the entire message can be pushed immediately on to the MUA. The simple message informing the MUA is better suited for a mobile user as the user can decide when a message is important enough to download and view, allowing the network to not waste valuable bandwidth forwarding unimportant messages.

4.3 MMS

4.3.1 MMS Overview

Multimedia Message Service (MMS) is a popular extension of SMS, a user is able to send and receive messages that contain multimedia content, typically the content is video, audio, image or text [WAP Application Protocol Forum, Ltd. 2001]. The message usually originates from a mobile device and is usually destined for a mobile device, although to or from a computer with internet connectivity is possible. The focus of this research was mobile to mobile MMS messaging.

4.3.2 MMS Process

The content to be sent exists on the user's mobile device, the user chooses to send an MMS and all the relevant media is added to the MMS, this includes images, video, audio and text. Once everything is added the MMS can be sent to its destination mobile device [WAP Application Protocol Forum, Ltd. 2001]. The size of the message to be sent is restricted by the operator's WAP gateway; an operator is able to set the maximum size of messages allowed to be sent in the network via the WAP gateway. At the time of writing Vodacom SA gate limit was set at 1Mb.

4.3.3 Network Protocols Used

For the MMS message to be sent the sending mobile device establishes a connection with the MMS Message Switching Centre (MMSC) via TCP/IP, the message is then encoded and encapsulated in a MMS format to be sent to the MMSC [Now.SMS, no date] and [Le Bodic, 2002]. The sending is done using a HTTP post method; the MMSC receives the MMS message, validates it and stores it; the connection with the sending mobile device is then terminated. The MMSC then creates a dynamic link to the newly stored MMS message which is forwarded to the destination mobile device via a WAP push method. The destination mobile device receives the WAP push in the form of a SMS and selects the link to download the MMS message. The destination mobile device then establishes a TCP/IP connection with the MMSC and uses a HTTP GET function to download the MMS message. Once successfully downloaded the connection is closed and the MMS has been successfully sent from the source mobile device to the target mobile device.

4.3.4 Restriction on MMS

Viewing the MMS process it can be seen that a MMS message needs to be sent across the air interface to the MMSC and then again be downloaded from the MMSC to the destination mobile device. This double air interface transfer causes a MMS message to take a while from being sent by the sender to being received completely by the receiver. Not mentioned in the process above is that the network handles content adaption, which is converting a message to a format that is compatible with the receiving mobile device, this conversion adds a small amount of time to the message transfer. The constraint on the size of the message to be sent would usually be attributed to the service provider's gateway.

4.3.5 MMS Characteristics

Mobile devices are usually asymmetric in their downloading and uploading rates, downloading rates are usually higher than uploading. This is because less power is needed for downloading than uploading, as well as downloading being a more popular service than uploading. Therefore in a service such as MMS where there will always be the uploading of data, the uploading will normally take longer than the downloading. The actual data rates achieved is still dependent on the bearer channels assigned to the connection and how often the bearer channels are switched during the transfer. The uploading of the MMS will be more susceptible to lower data rates because of the power constraints and lower maximum data rate achievable for uploading by a device. MMS is split into two stages, an upload stage and a download stage. The two stages are autonomous and stage 1 can be completed a long time before stage 2 starts. This frees a mobile device and network resources when a mobile device is not actively involved in the transfer.

4.4 SMS

4.4.1 Overview

Short Message Service (SMS) [Le Bodic, 2002] and [ADC NewNet, Inc. 1999] was the first data service to be added to the previously voice only mobile phone networks. The protocol of SMS has not changed much since its encapsulation by Implementation of Data and Telematic

Services Experts Group (IDEG) and the first commercial releases in 1993 by a US, a Norwegian and a UK operator. This service, previously limited to GSM, has become extremely popular and is supported by almost every network technology including 3G.

SMS is a service that allows users to send a short text message limited in the number of characters from one mobile phone to another. The size of an SMS is 140 octets, which when using 7 bit ASCI is equal to 160 characters [Le Bodic, 2002]. The limit placed on the size of an SMS is due to SMS using the Mobile Application Part (MAP) of the SS7 protocol which has a payload of 140 octets. A longer SMS message can be sent by combing multiple 140 octet SMS messages together in sequence and giving each 140 octet SMS message a sequence number to enable re-construction of the message on the destination mobile phone. When individual messages are combined the SMS payload is reduced to 139 octets as 1 octet is used for sequencing, enabling a theoretical maximum of 256 SMS messages grouped together, practically only 6-8 messages are grouped together because of the cumbersomeness of reading a large sequences of SMS messages, most users will opt for a voice phone call instead.

4.4.2 SMS Process

A mobile device compiles a text message and sends it to the Short Message Service Centre (SMSC) using the MAP of SS7 protocol. Once the SMSC receives it, it stores the message and then request the information about the current location of the target mobile device from the mobile device's Home Location Register (HLR). The SMSC then sends the SMS along with the information received from the HLR to the Mobile Switching Centre (MSC), which then attempts to route the message to the target mobile device. If routing is unsuccessful because the target mobile device is currently off and not connected to the network then sending of the message fails. The MSC informs the SMSC of the failure and the SMSC then stores the SMS in the hope of delivering it at a later stage. When the target mobile device once again rejoins the network it will inform its HLR, which in turn informs the SMSC of the rejoining. The SMSC checks for any messages destined for the MS and attempts the sending process again. The SMSC sends the SMS with the information from the HLR to the MSC. The MSC then contacts the Visitors Location Register (VLR) to confirm and authenticate that the target mobile device is in its cell. If it is then the message is transferred to the target mobile device and the sending SMSC and the sending mobile device are informed of its successful delivery [ADC NewNet, Inc. 1999] and [Enck, 2005].

4.4.3 Factors Affecting SMS Delivery

SMS messages are normally delivered immediately without any problems, but there are things that can go wrong in the delivery of SMS messages. As stated above the target mobile device can be disconnected from network in this case a message is delayed or if the mobile device is disconnected for too long a SMS can expire and be deleted from the network. Another problem that can occur is that a message is lost in the network or becomes corrupt while being delivered.

SMS messages are normally from a mobile device to a mobile device; however SMS can also be from computers on the internet to mobile devices or from mobile device to wired devices such as land line phones or computers connected to the internet [Le Bodic, 2002].

4.4.3 Restrictions of SMS

During the transfer of a SMS message the task that takes the longest is the looking up of the current location of the target mobile device. The actual transfer of data doesn't take that long because SMS messages are very limited in size [Head Developer of Clickatell, 2007]. A SMS message is transferred across signalling channels and it can be sent and received while a device is occupying a data channel. Voice call setup, data session setup and cell updates all share the signalling channels; all these tasks only use the signalling channel for a small amount of time and can easily co-exist. However it is possible that many connection setups or SMSs being sent could hinder another SMS from being sent. This can be seen on major public holidays when a voice call is not able to be setup.

4.5 HTTP Web Browsing

4.5.1 Overview

Mobile web browsing is defined for our research as the ability to browse the World Wide Web from a mobile device. Most mobile devices come equipped with a default web browser installed by the manufacturer, but there exists third party mobile web browsers which can be downloaded and installed on the mobile device. Mobile browsers have become very advanced and offer all the features of a desktop browser, such as support for CSS, Javascript and AJAX, and added functionality to handle small screen low bandwidth connections. Popular browsers for mobile devices are Opera mini from Opera, Safari from Apple, and Skyfire [Hardy, 2008] and [Virpi, 2006].

4.5.2 History and Present

At the start of mobile browsing mobile browsers used a different technology to normal desktop browsing, these differences often required content to be specifically created for the mobile browser [Virpi, 2006]. Users were unhappy having different content using a mobile compared to using a desktop and thus mobile web browsing technology was striving to access the same content as desktop browsing. The mobile web now displays the same content as the desktop web, with added features of web page scaling to allow pages to display neatly on the smaller mobile screen.

4.5.3 Mobile Browsing Process

To be able to browse the Internet from a mobile device the mobile device needs a mobile web browser and a connection to the internet. A user types in the URL of the target web page, the URL information is sent to a DNS, which then returns the IP address of the server on which the web page resides. The web browser then establishes a connection between itself and the target web server this is usually TCP/IP connection and then uses the Hyper Text Transfer Protocol (HTTP) that runs above TCP/IP to transfer web pages to the browser.

4.5.4 Commands and Protocols Used

The browser uses the GET command of HTTP to request the web page. The GET command and other HTTP commands contains headers that informs the server with information about the browser requesting the web page such as the name of browser, the HTTP version, caching details and cookie information. The server then transfers the web page to the browser based on the information giving by the browser [Fielding, 1999] and [Graphics and Media Lab, no date].

4.5.5 Objects and Pipelining

A web page consists of many objects that make up the web page, objects such as images, video and flash animation. Large web pages such as www.cnn.com and www.amazon.com may consist of a 100 objects all of which need to be downloaded to the browser. The HTTP version being used can affect downloading speed, for example HTTP 1.1 allows for pipelining and persistent connections, whereas earlier versions did not. In pre HTTP 1.1 versions or if a browser does not implement the bandwidth optimisation techniques then each GET and POST pair will require a new connection. This is a synchronous process and takes a long time causing a web page to take a long time to load. What happens is the browser requests an object, sets up the connection, downloads the object and closes the connection, then repeats the process for every object. HTTP 1.1 allows a browser to open a connection and send multiple GET requests and at the same time receive objects from the web server. This greatly improves the time taken to download a web page. Not all desktop browsers support pipelining yet so it is likely that many mobile browsers will not currently support it.

4.5.6 Browser Support

A web page can be cached on the local machine, or on an organisational cache proxy server. The benefits of caching a page is that it takes less time to download it because the page can be retrieved from cache and only objects that have been updated since the cached version need to be downloaded from the server. Another benefit is a large organisation would be saving bandwidth and bandwidth cost. Some mobile browsers such as Opera mini and Skyfire cache formatted versions of web pages, optimising it to be downloaded and displayed on their respective browsers.

4.5.7 Factors Affecting Performance

Once the page is downloaded by the web browser it is up to the web browser to display the contents of the web page. Different formatting and mark-up languages are used to display the web page such as HTML, XHTML, CSS, WAP 2.0 and WML. What the browser supports will affect the way the web page looks and what interactions a user may perform on the page.

A web browsing application will be sensitive to the end-to-end delay of the network if many objects need to be downloaded, wasting a large percentage of the time on establishing new connections for each object to download. In the case when a large web page needs to be downloaded the bandwidth offered to the mobile device can greatly affect the time taken to download the web page.

4.6 Multimedia in 3G

4.6.1 Overview

Mobile media streaming in this research is defined as the streaming of audio or/and video data across a wireless telecom network to a mobile device for listening or viewing purposes. Media could also be streamed from a mobile device; this is less common and is usually as a result of a video call. In media streaming applications data is streamed by a streaming server to a receiving device. The benefits are that the data does not have to be permanently saved and only part of the data needs to be downloaded before playback starts. Three types of media streaming services are defined, namely: media streaming, media broadcasting and video calling. Generally a user needs a 3G enabled phone connected to a 3G network to use media streaming services, although GPRS and Edge phones are also capable of streaming but at lower data rates.

4.6.2 Media Streaming Overview

In the media streaming service data is permanently stored on a server, the data is sent via a streaming server to the end user. The process is as follows: a mobile device with streaming playback software installed (this could be built into a modern mobile web browser or an application on the mobile device) connects to a server via the internet. The user chooses the file to view and selects to stream it; a streaming connection is then setup between the mobile device and the streaming server. The streaming server encodes the media to compress it making it more suitable to be transferred across the network. Video could be encoded in 3GPP format and audio could be encoded in MP3 format, once the client streaming software receives the data, it is decoded before viewing.

4.6.3 Network Treatment and Protocols

Media streaming is classified as real time traffic and the network will attempt to give the service priority status and assign it when possible a high data rate bearer channel. 3GPP standard has marked 64Kbit/s and higher as streaming channels. To transfer the data the Real-time Transfer Protocol (RTP) is used, while Real-time Transfer Control Protocol (RTCP) is used to supply transfer control information and Real-time Streaming Protocol (RTSP) is used to perform functions on the streamed data [TeliaSonera, 2004]. RTP shows the size of the data payload, the sequence number of the packet and the time stamp and delivery monitoring, allowing RTP to guarantee data delivery through retransmissions. RTCP monitors the network conditions and informs the server and client about congestion, the connection can then be adjusted accordingly. RTSP protocol allows the client to issue commands to the streaming server such as pause, play, rewind and fast-forward. All these protocols are implemented above the TCP protocol or UDP protocol. UDP is considered to perform better than TCP as it does not have congestion control of TCP, enabling UDP to stream data freely.

4.6.4 Multi-Cast and Uni-Cast

A media streaming service could use multicast or unicast streaming, it is most likely to use unicast, this means that there would be a separate data stream from the streaming sever to the target device for every device. The reason behind this is that each file on the server and time position of viewing for each user is likely to be different, especially if the user makes use of the RTSP commands. The data is streamed across the Internet using RTP over TCP or UDP via a gateway to the UMTS core network. On arriving in the UMTS network it is processed by the Internet and Multimedia Subsystem (IMS) tagged as real-time traffic and forwarded on to its destination mobile [Magedanz, 2005], [Chakravarthy, 2006] and [Camarillo and

Garcia-Martin, 2006]. The data is then received at the BS where it is sent across the UMTS air interface to the mobile device, this is where most of the delay, error and restriction of bandwidth will occur. The BS then transmits the data to the mobile device and on receiving of data the mobile device decodes the media and plays it for the user.

The main bottleneck of this system would be the air interface, the bandwidth is limited and only a few users are able to use a high data rate channel at a time. Three users are able to use 384Kbit/s channels simultaneously in a single cell, if more users need to use the bandwidth then the network needs to make a decision to drop channel rates or to deny new users network access until current users have released the channels. However a telecom network is more complex than that and the user's data rate depends on many factors, from available bandwidth, number of users, traffic priority class and required power to transmit. All this means that there may be delays in the streaming of data from the BS to the MS affecting the users viewing of the media.

4.6.5 Buffering

Buffering is used to mitigate the affects of delays. Most streaming applications will first implement a playback buffer before starting the play back of data to ensure smooth play back viewing. This is not a guaranteed solution as the playback buffer can run out and the application would need to re-buffer itself, this can be seen often when viewing media where the playback rate is faster than the download rate. It is possible however to stream data to the user at a rate higher than the playback rate and in doing so grow the buffer, which will provide a greater offset against transmission delays. Since media streaming requires high data rates it is resource-intensive and not many users can make use of the service simultaneously. A positive aspect of media streaming is once streaming is complete the resources are freed up and can be re-assigned for another use by the network.

4.6.6 Mobile Broadcasting Overview

Mobile broadcasting for this research is defined as the broadcasting of audio and/or video media to mobile devices in a given region. Mobile broadcasting service is most commonly offered by the service provider itself as the nature of broadcasting is likely to require dedicated channels used for broadcasting purposes, although there are ways to get around this. To use the broadcasting service a user needs a 3G mobile device and subscribe to the broadcasting service, it is possible for non-3G devices to use the broadcasting service but these devices are capable of lower data rates resulting in a decrease in quality [Hartung et al, 2007] and [Bleidt, 2004].

4.6.7 Mobile Broadcasting Process

A server containing the data to be broadcasted multicasts the data to all BS stations that will require it, this all occurs in the wired network over UDP/IP connection. Once the BS receives the data it simply pushes it across the dedicated channels and all the mobile devices tuned into the channel receive it. The security and billing of the system is out of the scope of this research. Many users will receive the same data using the same bandwidth. This allows broadcasting to scale well, an increase in users does not result in an increase of bandwidth.

The cons of broadcasting are that it requires dedicated network resources that can't be re-assigned for other services. Important factors from a user's perspective are that broadcasting does not allow retransmissions or feed back from the user to the server, it is a simple one way communication. Some users may receive good jitter free reception while others may receive bad reception rendering the service unusable; it all depends on the cell conditions. A user near the BS will receive a strong signal while a user on the cell edge is more likely to receive a weak signal; this affects the user's quality of service.

4.6.8 Mobile Broadcasting Standards

A service provider will need a wide variety of channels to make mobile broadcasting popular; this would require large amounts of the scarce bandwidth resource [Hartung et al, 2007]. To solve this issue mobile broadcasting is attempting to make use of the satellite and normal TV broadcasting spectrum [Bleidt, 2004]. Listed below are a few mobile broadcasting standards:

DVB-H (formerly DVB-X) uses part of European DVB digital TV to carry video and audio data to handsets. (Nokia)
3GPP MBMS (Multimedia broadcast Multicast Service) - uses a dedicated channel on 3G network to broadcast video and audio to handsets (Vodafone)

DMB - Digital Multimedia Broadcasting offered by Samsung.
Japanese Handsets with ISDB (DVB variant) digital TV receivers (Sanyo, Nec)

The future of mobile television is promising and by the year 2010 it is expected that 40% of mobile phones will be TV enabled [3G.co.uk, 2006].

4.6.9 Mobile Video Calling Overview

Mobile video calling for the purpose of this research is defined as a video call between two mobile devices. Mobile video calling has been around since the arrival of GPRS technology but has not really taken off. Most research done during the GPRS period concluded that video calling is possible over GPRS but predicted that 3G would raise the quality of video calling making it popular to the mass market. Although 3G has dramatically improved video calling quality the service has not yet gained mass market usage [Bleidt, 2004] and [Mirial, 2007].

To use the video calling service a user needs a network service provider that offers the video calling service, a sending and receiving mobile device that is capable of video calling.

4.6.10 Video Calling Process

The user selects the target mobile device that it wishes to video call and contacts the network to establish a connection. The network's IMS system which deals with video calling and other multimedia functions sets up a video calling session. The call setup uses the SIP (Session Initiation Protocol) for setting up and tearing down of the session, as well as handling the signalling of the session. The network checks the user’s service profile to see if it contains video calling by contacting it’s HSS (Home Subscriber Server) and then proceeds to use SIP to setup the session [Chakravarthy, 2006], [Camarillo and Garcia-Martin, 2006] and [Magedanz, 2005].

SIP does not transfer any data packets. This is done by the RTP protocol, using packet based switching running above a TCP/IP or UDP/IP network layer. This architecture is defined as part of the 3GPP standard and allows for the integration of wireless and wired networks as both systems are built on IP networks and caters for the goal of converging wireless and wired networks. The architecture using the IMS system along with SIP and RTP attempts to provide a high quality of service with minimum delays, but this still can not be guaranteed. For this reason 3GPP have defined 3G-324M a circuit switch standard [Fisher, 2005].

4.6.11 Guaranteeing Delays with 3G-324M Protocol

3G-324M protocol guarantees a fixed delay between two end points making it a suitable protocol for time sensitive connections. In a service such as video calling where data is both created in real time and the users interact with each other, the client is unable to pre-buffer media before playback making the service susceptible to delays. For a well flowing conversation to take place, the delay between statements and replies need to be kept to a minimum, while to increase the quality of audio and video a high bandwidth is required, making video calling an extremely demanding service.

Video calling is a duplex service and the mobile device will need to acquire channels for uploading and downloading, in most UMTS system this should be fine as uploading and downloading occurs on different frequencies. This doesn’t change the fact that a video call would be using from the upload bandwidth and from the download bandwidth. Since buffering can not take place in video calling any problems with the connection would come out as jitter and noise. It is possible due to re-assignment of channels or power constraints that a video call can lose quality intermittently or lose connection completely. The resource-intensive nature of video calling will restrict the number of simultaneous users of the service but once the call is complete the resource will be re-allocated to the network. It remains to be seen how much strain video calling will put on real networks, because as yet video calling is not popular.

Chapter 5

NS2

5.1 Overview

Network Simulator 2 (NS2) was chosen as the simulation environment for the experiments. NS2 is a popular academic research tool used in communication network research. This open source tool supports many network protocols and architectures. New modules are continuously being developed by third parties and made available for public use. NS2 compiles and runs on Linux operating systems or on Windows via the Cygwin emulator. A researcher then uses the simulator to setup a simulation and then executes it. An NS2 simulation produces a results file which shows a log of all events taking place during the simulation. This results file is used in analysis of the simulation. There exists a visual extension to NS2 that allows a user to play back a simulation in a visual animated network. This visual animation shows the movement of packets and acknowledgments through the network [The NS2 Manual, 2007], [NS-2 Tutorial, no date] and [Chung and Claypool, no date].

5.2 NS2 Components

The Major components making up a simulation in NS2 is a script program, the network components libraries, and the scheduler, [Chan, 2006] and [The NS2 Manual, 2007] see Figure 5.1 NS2 entities. To allow NS2 to be more effective at both network setup and simulation execution, NS2 is implemented using two languages, C++ and OTcl.

The script program is written in OTcl, which is an object orientated extension to the Tcl scripting language [Chung and Claypool, no date] and [The NS2 Manual, 2007]. The ability to alter networks quickly through OTcl made NS2 an excellent choice for this research. In Appendix B is an example of an OTcl script used in the simulations.

The OTcl part of the simulator thus deals mainly in the setup and configuration of the simulation. Once the simulation is started, it is the C++ part of the simulator that performs all

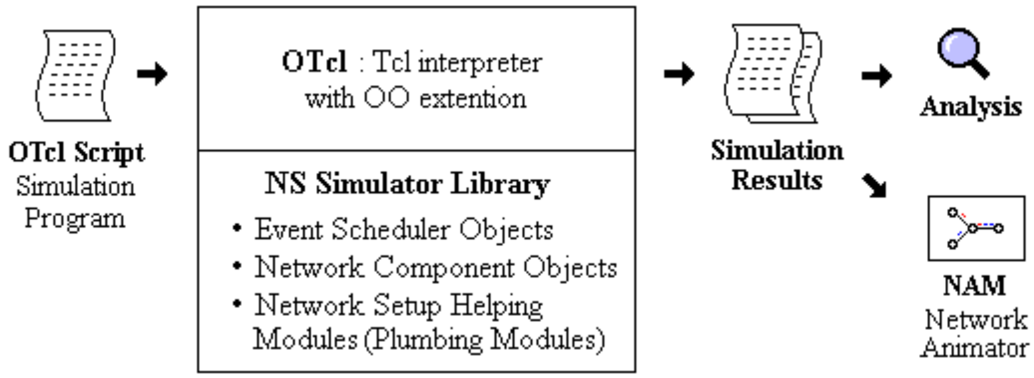

Figure 5.1 Entities making up a NS simulation. Source: NS2 Manual (version dated 12 March 2007).

the packet processing operations. Through the use of C++ processing and execution of the simulation can be done efficiently. Both the scheduler and object component libraries are implemented in C++. The scheduler is the discrete event handler that deals with all the events that takes place during the simulation. This includes when to fire events and who to inform when an event fires. It is also responsible for all timing functions such as to simulate all propagation and processing delays of components. The C++ code can be modified to implement new protocols if needed [The NS2 Manual, 2007]. NS2 however has all the protocols needed by present research and no modification to the C++ code was made.

5.3 EURANE

Enhanced UMTS Radio Access Network Extension EURANE was developed in the SEACORN (Simulation for Enhanced UMTS Access Core Network) project. This extension module to the NS2 implements needed functionality to simulate UMTS wireless networks. The EURANE module added 3 component objects that enable the simulation of UMTS wireless networks. These are UE (User Equipment), BS (Base Station), and RNC (Radio Network Controller) [EURANE Manual, 2005].

5.3.1 EURANE Elements and Protocols

UE: Represents a mobile device, mobile devices are able to change location based on an input script.

BS: The base station has a number of parameters configuring its air interface. A UEs air interface properties are defined by the BS it is connected to. One or more UEs may connect to a BS.

RNC: Connects a BS to the core network consisting of SGSN nodes and GGSN nodes. A bandwidth, propagation delay and queue properties may be set between RNC and BS.

A few of the channels and protocols supported by EURANE are: Radio Link Control (RLC), Acknowledged mode (AM), Unacknowledged mode (UM), Medium Access Control (MAC), Random Access Channel (RACH), Forward Access Channel FACH), Dedicated Channels (DCH), High speed Downlink Shared Channel (HS-DSCH) this is the HSDPA channel.

The EURANE module has limitations that affect the types of simulations that can be performed. Only one UMTS cell can be simulated. There is no cell capacity limitation for this one cell. Wireless links are static and once created during setup will exist for the duration of the simulation. During the simulation the link can not be reassigned to other mobile nodes. All wireless channels can only be configured with the same parameters, the same TTI (Transmission Time Interval) interval, the same bandwidth, same acknowledge mode, and all other parameters are the same. [Wang and Prasad, 2005]

Figure 5.2 below depicts the network topology and the NS2 entities used to construct the network; followed with a brief description of the NS2 entities.

5.4 How NS2 was used

The goal was to simulate 3G applications under realistic network conditions while being able to vary the network factors from simulation to simulation, so as to compare results. NS2 with the EURANE extension would enable this goal to be achieved. NS2 version 2.30 with a EURANE extension patch was installed on a personal computer running Ubuntu Edgy Linux.

This was the latest version at the time of installation. The installation was successful and resulted in a stable simulation environment.

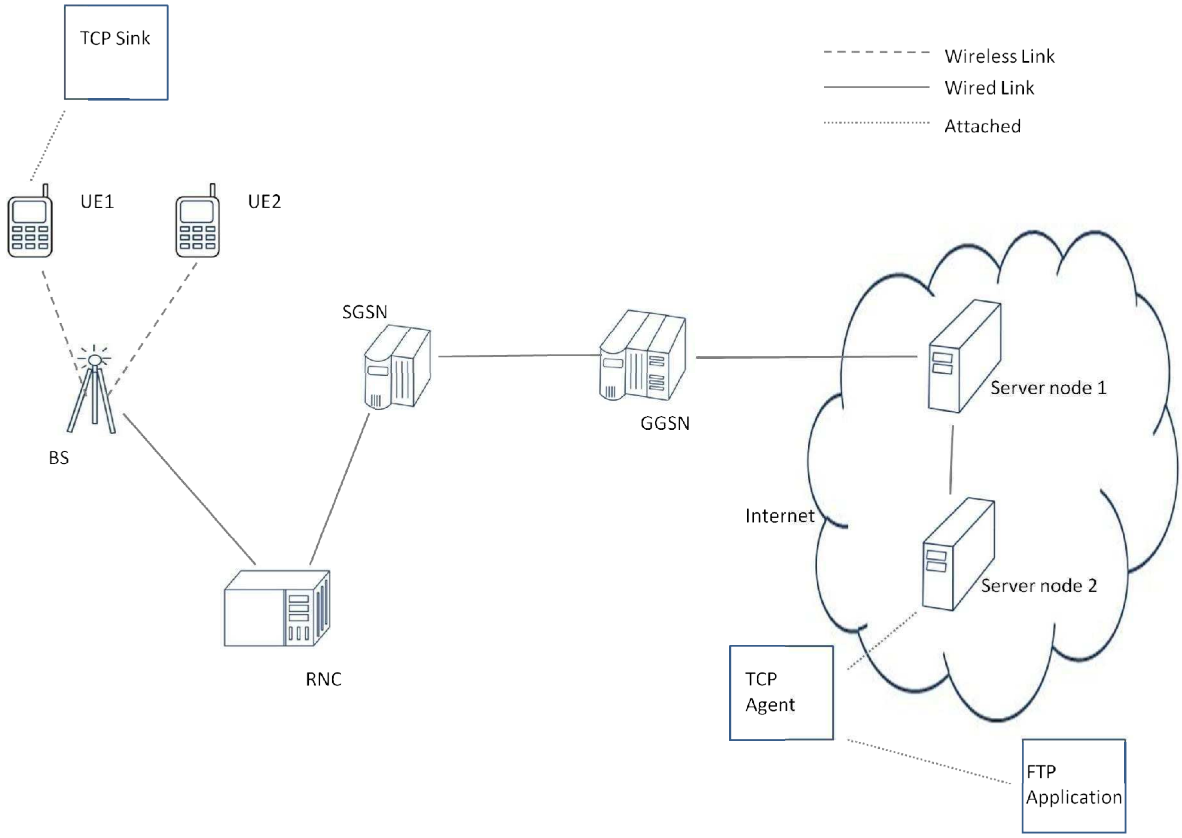

Figure 5.2 Elements and interfaces of a EURANE UMTS network created in NS2.

OTcl was used to construct the topology seen in Figure 5.2. This is a typical UMTS network consisting of mobile devices connected to a base station. EURANE's mobile nodes were used to represent the mobile devices while the EURANE base station node was used to represent the base station. The interface between the mobile node and base station were set up to have typical 3G values, for values such as its TTI and other configurable variables. Please see experiments section to see all configuration settings. Attached to the mobile node is a movement script which indicates the mobile node’s at every time instance during the simulation.

Table 5.1 EURANE elements and their descriptions.

Element	Description
Node	A node is a reactive component and a basic component forming a network. A connection in NS2 will be from node 1 (source) to node 2 (destination) via a number of routing nodes between. Nodes have internal properties performing functions on packets. BS, RNC, SGSN, GGSN, Server 1 and Server 2 are examples of nodes.
Mobile Node	A mobile node is an extension to the basic node object that enables a wireless interface allowing a mobile node to partake in wireless communication. UE 1 and UE 2 are examples of mobile nodes.
Link	A link is a reactive component that connects two nodes to each other. A link has internal properties that affect packets during data transfer. Links implement the behaviour of the network layer down and simulate link propagation delays, network protocol layer queuing and link errors. See the key in Figure 5.2 to identify links in the figure.
Channel	A channel is similar to a link but allows multiple senders to share a link. All protocol behaviours from the application level down to the MAC level are implemented in a channel. There exist both wired channel and wireless channel types. Wired channels have a fixed propagation delay whereas wireless channel propagation delay is dependent on the distance. See the key in Figure 5.2 to identify channels in the figure.
Agent	An agent is a component that can be both active and reactive in a network. An agent may receive a data packet as input, perform the needed function on it and send it as output, the same as nodes and links. But it may also produce its own control signals that affect the communication within the network. TCP is an example of an agent.
Traffic Generator	Generates traffic for the network. Traffic generators are connected to a transport agent, which inserts data into the network based on the type of the traffic generator. The generator types are either distribution type, such as constant, exponential or application type such as FTP or HTTP. FTP is an example of a traffic generator.

RNC element of ERANE is used to represent the Radio network controller and is linked to the Base station. The link between RNC and BS is also configured to acceptable 3G values. The core network consisting of SGSN and GGSN are represented by regular NS2 nodes but configured to that of 3G node standards. SGSN is connected to the RNC and to the GGSN, while the GGSN is connected to a regular NS2 node that represents a wired internet server. The wired internet server in turn could be connected to other wired internet servers. The whole network is set up so that the end-to-end delay is that of a 3G network as well as the network factors such as bandwidth, and other properties. An application agent and a traffic agent are then connected to the internet server node and the mobile node. Once this is done a simulation can be executed and the results recorded in a file.

To allow for performance comparison of different network factors the network configuration needs to be changed and the simulation executed again. This requires a large number of simulations with different configurations to be executed. To allow this to happen quickly, easily and accurately, supporting shell scripts were created that automatically modified OTcl scripts and re-executed it. Other support shell scripts were used to extract and process the NS2 results files. The result is that an extremely large number of simulations were able to be executed and the results extracted and processed. See Appendix C for an extract of a results file. These results can then be used to determine the affects of network factors on application performance so as to have a better idea of how an application will perform in a real network.

Chapter 6

6. Experiments, Results and Analysis

This Chapter describes network simulations to evaluate the performance of 3G applications under different network conditions. These include throughput obtained when transferring data, the time taken to transfer data and the percentage of data that arrives in a timely fashion. Applications simulated are FTP, MMS, email, SMS, web browsing, media streaming, media broadcasting and video calling. Some of the most important results are that media streaming (broadcasting and video calling included) are sensitive to bandwidth changes. A small change in bandwidth has a significant change in performance. File transfer applications are less sensitive to bandwidth and show a small increase in performance with an increase in bandwidth.

Section 6.1 covers implementation details common to all simulations. In Section 6.2 FTP applications are investigated, Section 6.3 looks at email, Section 6.4 at SMS, while in Section 6.5 MMS experiments are covered and Section 6.6 explores web browsing. Sections 6.7, 6.8 and 6.9 examine media broadcasting, video calling and media streaming respectively.

6.1 Common Implementation Details

Certain aspects are common to all of the simulations, these are set out below:

6.1.1 A Network Scenario

A network scenario refers to an instance of a network and the delay of the links connecting elements of the network to each other. Three network scenarios existed for this research. The first scenario had an end-to-end delay of 170.5ms with a large delay of 135ms at the air-interface link. In this scenario the delay bottleneck would be at the slow air-interface link. The bottleneck changes for the second scenario. In the second scenario the bottleneck was moved to the server, the server had a delay of 135ms, while the end-to-end delay was kept at 170.5ms. The third scenario removed the bottleneck completely and the end-to-end delay of the network decreased to 45.5ms. Table 6.1 illustrates the link delay that existed between each element in the network. Applications tested were simulated on each of these scenarios so as to study the effects of delay on the applications performance.

Table 6.1 End-to-end link delays and total network delay shown in milliseconds.

Link	Scenario 1	Scenario 2	Scenario 3	Link Bandwidth
UE-BS	134.5	10	10	16-2048Kbps
BS-RNC	15	15	15	622MBps
RNC-SGSN	0.5	0.5	0.5	622MBps
SGSN-GGSN	10	10	10	622MBps
GGSN-Node1	10	134.5	10	622MBps
Node1-Node2	0.5	0.5	0.5	622MBps
Total end-to-end	170.5	170.5	46.5	-
Round trip delay	341	341	93	-

6.1.2 Network Setup

The wireless network contained a Mobile Station (MS) that connected to a Base Station (BS); the BS was connected to a Radio Network Controller (RNC). The RNC in turn was connected to a Serving 3G Server Node (SGSN), which was connected to a Gateway 3G Server Node (GGSN). The GGSN links the UMTS network to a wired network such as the Internet or to a subsystem in the core UMTS network, which contained an application service server. This is depicted in Figure 6.1.

Each element in the network is linked to another; this link between two elements has properties defining the links behaviour. The most important properties of the link are its delay and its bandwidth. Table 6.1 displays each link and the delay attribute of the link, as well as the bandwidth attributes of each link. When the entire network is viewed the total delay of all the links from the MS to the application server add up to 170.5ms, incurring a round trip delay of 341ms, which is a commonly accepted estimate of a 3G networks round trip delay value [Panian, 2004]. Generally the wired network is associated with short delays and high bandwidths; these wired networks all have a high bandwidth of 622Mbps, while the air-interface bandwidth of the wireless network varies between 16Kbps-2048Kbps.

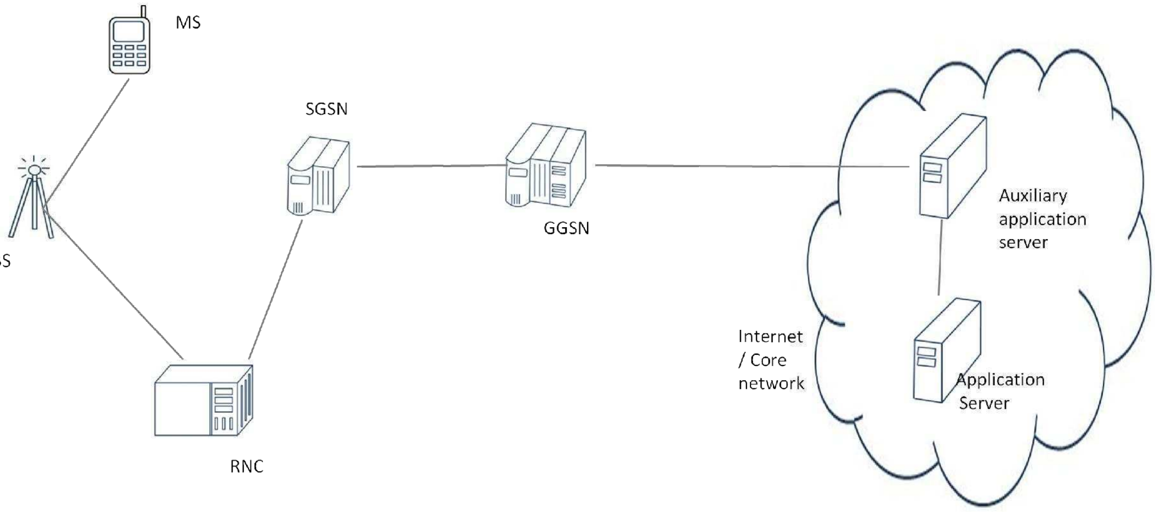

Figure 6.1 The basic network used in simulations.

The actual rates of channels simulated were: 16Kbps, 32Kbps, 64Kbps, 128Kbps, 144Kbps, 384Kbps and 2048Kbps.

6.1.3 Parameters and Protocols

This section describes the network parameters and protocols common to all the experiments.

A sending node runs an application service. This application service will transfer data to the destination node. The data may be a file, message or streaming media, depending on the application being simulated. In the case of a file or message being transferred the NS2 FTP agent was used. The size of the data to be transferred is set by specifying the FTP agent's number of packets to be transferred and the size of a packet.

The FTP application ran above a Transport Control Protocol (TCP) layer that was setup between the sending and receiving nodes. The TCP implementation, the packet size and other factors such as fast restart could be set, although in most cases the TCP implementation controls the window size and congestion control factors. All the congestion control calculations were re-calculated periodically based on the failure, success and time taken to send a packet of data.

Tests were done to identify the most efficient TCP implementation for the experiments. TCP Vegas was chosen as it performed best in these initial tests. [Dubois, 2005] have performed similar experimental research using NS2 to compare TCP implementations, which also found TCP Vegas to either outperform other TCP implementations or compete with the top performing implementation in most situations. TCP Vegas was then used for both uploading and downloading of files. However in cases of uploading large files it was found that TCP Vegas performed poorly. Throughput would drop significantly once data transfers larger than 100KB occurred from mobile devices. Because of this TCP Reno was used for uploading from a mobile device for non-streaming applications. TCP Reno also suffered from a significant drop in throughput once a certain data size was reached, but this drop in throughput occurred at larger file size than that of TCP Vegas. The sudden drop is caused by constant Retransmission Timeouts (RTO).

RTO occurs when a sender sends a packet of data and does not receive an acknowledgement in the retransmission timeout period. TCP then shrinks the window size because the packet delay is seen as network congestion and the sender resends the packet. In the situation where the sender never receives the packet acknowledgement in time, a case of constant RTO is experienced. This is what occurred in some of the upload simulations performed. The same was experienced by [Dubois, 2005]. The long delays, low bandwidth and bursty nature of wireless channels are factors which can cause constant RTO.

TCP Reno was able to transfer larger files (or messages) in upload simulations than TCP Vegas. But after reaching a certain transfer size Reno's throughput would also drop significantly. In both Reno and Vegas this only occurred for small bandwidths, large bandwidths simulations were able to transfer large files without a sudden drop in throughput.

The parameters which changed from simulation to simulation for file and message transfer experiments were the file sizes (or message size), the air-interface bandwidths, and the network scenario used for the simulation. Each network scenario had a different link delay setup, see Table 6.1. All experiments were simulated for air-interface bandwidths ranging from 16Kbps-2048Kbps, the actual channel rates are shown in Section 6.1.2. A simulation was done for each combination of these parameters and the results recorded. To ensure that the best simulation for a combination was recorded, each combination was done for every valid window size up until the max window size based on the bandwidth delay product formula [Panian, 2004].

(Equation 6.1) Bandwidth x Delay = Max Window Size

The packet size used by file and message transfer applications was set to 1024bytes = 1KB. This falls in between the values of common packet sizes transferred across the Internet. Common sizes of packets transferred across the internet generally range from 576-1576Bytes [Chan, 2006].

Streaming media simulations (Media Broadcasting, Media Streaming and Video Calling) were simulated by a NS2 traffic generator, which streamed data at a constant rate. The constant rate could be set. This streaming rate was varied to simulate various frame rate and frame size combinations. The generated traffic was then transmitted across the network over the UDP protocol enabling streaming to take place freely as UDP has no congestion control mechanism that could throttle the flow of data. Streaming media applications do not require acknowledgments making UDP more suited for streaming than TCP.

The parameters which change from simulation to simulation for streaming media simulations were the frame rates, frame sizes, the air-interface bandwidths, and the network scenario used for the simulation. Each network scenario had a different link delay setup, see Table 6.1. The frame rates ranged from 5-15 frames a second, while the frame sizes ranged from 533 bytes to 12993 bytes. The bandwidth at the air-interface ranged from 16Kbps-2048Kbps, the same as FTP simulations. A simulation was done for each combination of these parameters and the results recorded.

The UDP packet size was set to 1024bytes = 1KB. This falls in between the values of common packet sizes transferred across the Internet. Common sizes of packets transferred across the internet generally ranges from 576-1576Bytes [Chan, 2006].

6.1.4 Frame Sizes and Frame Rates

This research has classified video footage into three genres, news, drama and action. The frame size and frame rate is dependent on the genre of video. Action video footage has large frames and requires high frame rates, since scenes change quickly and many fast movements occur. In a news broadcast the scene does not change much as the anchor's head is normally stationary, this causes News styled video footage to perform well with small frame size and low frame rates. Dramas fall in between the other two genres and have an intermediate frame size, while the frame rate does not have to be as high as action footage. Mobile devices have different screen sizes and are capable of different frame resolutions.

[Song, 2002] looks at the different frame sizes (small, medium and large) for a given frame resolution. Using these frame sizes and frame resolution, it was possible to customise frames for three mobile screen sizes, 128X96, 176X220 and 240X320 resolutions, using the same method described in [Bleidt, 2004]. The frame sizes used and its related genre are displayed in a Table 6.2. The frame rate and frame size affects the required bit rate needed to playback the video footage.

(Equation 6.2) [Frame size] X[Frame rate] = [Required bit rate]

6.1.5 Method

Part of this work included writing a Linux shell script which allowed for the automatic modification of network conditions. This shell script modifies and executes OTcl scripts written to simulate the target applications. The OTcl script itself contained all the elements and parameters forming the network described above. The OTcl script was executed on the NS2 simulator and an event log file was output displaying the results of the simulations.

Table 6.2 Screen size, frame rate, frame size and resulting bit rate.

Genre	Screen resolution	Frame size (bytes)	Streaming rate at 5 frames/second	Streaming rate at 10 frames/second	Streaming rate at 15 frames/second
News	128x96	533	21Kbps	41Kbps	62Kbps
	176x220	1806	71Kbps	141Kbps	212Kbps
	240x320	3449	135Kbps	269Kbps	404Kbps
Drama	128x96	1007	39Kbps	79Kbps	118Kbps
	176x220	3415	133Kbps	267Kbps	400Kbps
	240x320	6521	255Kbps	509Kbps	764Kbps
Action	128x96	1997	78Kbps	156Kbps	234Kbps
	176x220	6773	257Kbps	513Kbps	770Kbps
	240x320	12993	508Kbps	1015Kbps	1523Kbps

The results output file of FTP, Email, SMS, MMS and HTTP simulations was input to another Linux script which extracted the relevant information needed to analyse the simulations. Using the final set of information the best performing window size, time taken for download for each file size (for SMS, MMS and Email it will be message size while for web browsing it will be page size), and throughput were calculated.

The result output file of media broadcasting and video calling was input to a Java program which processed the results and extracted the relevant information needed to analyse the simulations, such as the throughput and percentage of frames transmitted below the maximum inter frame delay period (for media streaming the size of the required buffer in seconds was calculated). These results were then analysed and compared for different frame rates, frame sizes, bandwidths and network delays.

6.1.6 Results Layout

The results for each application simulation were grouped into cases. In each case one factor was varied while keeping the other factors constant. The effects on performance for the varied factor could then be measured and analysed. The factors investigated were end-to-end delay, bandwidth, file size (message size or webpage size), for FTP, email, SMS, MMS and web browsing. In media streaming applications (Media broadcasting, video calling and media streaming) the factors investigated were end-to-end delay, bandwidth, frame size and frame rate.

In media streaming and non-media streaming applications, Case 1 examines the effects of end-to-end delay on performance and Case 2 examines the effects of bandwidth on performance. In non-media streaming applications Case 3 explores the file size and its affect on performance. In media streaming applications Case 3 looks at frame rate, while Case 4 investigates the effects on performance due to frame size.

6.2 FTP Experiments

6.2.1 Network Setup

To Simulate an FTP application a EURANE UMTS network connected to a wired network was setup, the wired network represents the Internet and contains a FTP server. Furthermore the network was setup as detailed in the Common Implementation Section 6.1.2 with the application server being a FTP server. The FTP simulation setup is depicted in Figure 6.2.

6.2.2 Parameters and Protocols

A sending node ran a FTP application and transferred files from the source to the destination node. The details are as described in the Common Implementation Details 6.1.3. The file sizes simulated were 20KB, 50KB, 100KB, 150KB, 300KB, 500KB, 750KB, 1MB, 2MB, 3MB, 5MB and 10MB. These file sizes were simulated for each bandwidth. TCP Reno was used for the uploading of files from a mobile device to a server and TCP Vegas used for downloading files from a server to the mobile device. A summary of the FTP applications parameters are presented in Table 6.3.

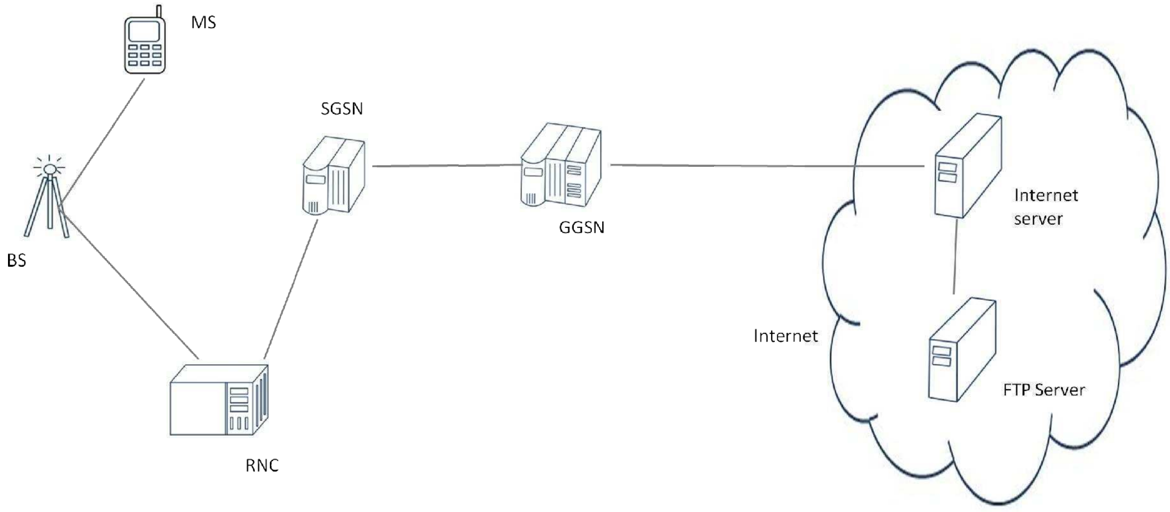

Figure 6.2 The network used for FTP simulations.

6.2.3 Method

The method is identical to that detailed in Common Implementation Section 6.1.5 with the target application server being a FTP server. These experiments were performed for both the uploading and downloading of files ranging from 20Kb-10Mb, which are all valid sizes for a FTP application.

Table 6.3 FTP application parameters.

Parameter	Value
FTP packet size	1024bytes
Number of FTP packets	50 – 10000
Transport protocol	TCP
Air interface	WCDMA
Air interface bandwidth	16Kbps – 2048Kbps
TCP	Vegas (download), Reno (upload)

6.2.4 Results and Analysis

In FTP simulations the throughput and time taken to download a file are used as measurements of performance. The change in performance was investigated due to changes in the end-to-end network delays; air interface bandwidth and file sizes. In each case a single factor was varied while keeping all other factors constant and the effects on performance measured.

Case 1: The effects of changing end-to-end delay while keeping bandwidth and file size constant and how this affects time and throughput performance.

Network delay does affect the time taken to transfer a file. The shorter the end-to-end network delay the shorter the time taken to download a file. This was the result for all the bandwidths tested. Once a packet has been sent the sending node waits for an ACK message from the receiving node before sending the next packet; or a RTO occurs and the original packet is resent. Therefore the shorter the delay the quicker the next packet or resend can occur. Longer end-to-end delays benefit from having a larger TCP window, which negates the effects of the longer delay. Simulations were done for all TCP window sizes and the best simulation results for each end-to-end delay were recorded and compared to each other. These results showed that FTP performed better on the shorter end-to-end delay network.

Where the bottleneck occurs in the network affects performance, if the bottleneck is at the receiving node it has the largest negative affect on the performance. If the receiving node takes long to send an ACK message the sender could experience RTO; whereas if the bottleneck is at the sender, the receiving node does not experience RTO.

[Roccetti et al, 2005] found that it took longer to download MP3 files from a geographically further server which also had a higher end-to-end delay. These results are the same as present research found, a decrease in end-to-end delay increases performance. Also previous studies showed GPRS to perform poorly partly because of its high end-to-end delay [Moltchanov et al, 2002]. 3G has a shorter TTI and a lower end-to-end delay than previous wireless technologies and will allow FTP applications to perform better.

.

Table 6.4 The effects of changing end-to-end delay while keeping bandwidth and file size constant on time and throughput performance. The file size is 1MB and bandwidth is 2048Kbps.

Transfer Direction	**Delay (ms)**	341*	341**	93
Upload	**Throughput (KBp/s)**	21.68	0.21	32.04
	Time (s)	46.134	4655.410	31.210
Download	**Throughput (KBp/s)**	2.28	28.49	30.34
	Time (s)	438.237	35.100	32.956

* Bottleneck at the air-interface. ** Bottleneck at the server on the wire network

In simulations with high end-to-end delay and a large file to transfer, some bandwidths experienced constant RTO with almost no packets making it through. [Dubois, 2005] discusses the occurrence of constant RTO and how increases in RTO can occur. The buffer size was varied but this did not improve the problem, simulations were then continued using default buffer size.

Case 2: The effects of changing bandwidth while keeping end-to-end delay and file size constant and how this affects time and throughput performance. Illustrated in Figure 6.3.

The higher the assigned bandwidth, the shorter the time to download the file and the higher the throughput achieved by the download. TCP uses a larger window size for higher bandwidths based on the time taken and successful delivery of a packet, which in turn shortens the time to download and increases throughput. The higher the bandwidth assigned to a user, the better the performance a user experiences. However bandwidth is limited and needs to be shared by all users and a balance needs to be found between the number of concurrent users and the performance experienced by each user.

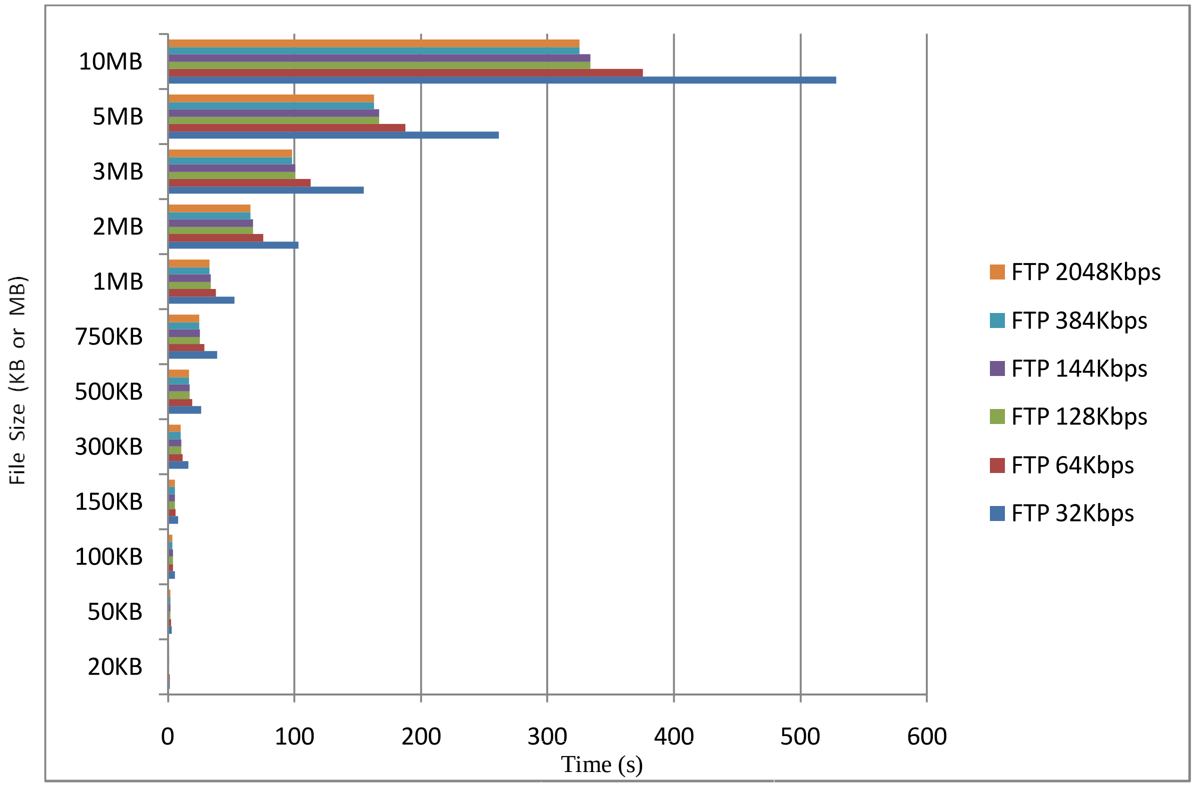

Figure 6.3 The time to download a file by each bandwidth for each file size.

The results show that an increase in bandwidth is insignificant when the file size to be transferred is small. This is because of the small absolute time taken to download a file and the time being to short for TCP congestion control to affect the transfer. In medium file size transfers there is a notable difference in time taken by high bandwidths compared to low bandwidths; whereas in large file transfers a significant time difference is seen between high and low bandwidths. In large file transfers TCP has time to optimise the window size for the bandwidth allowing higher bandwidth to have a large window size, increasing the throughput achieved by the bandwidth. Low bandwidths have a smaller optimised window size, resulting in a lower throughput.

An increase in bandwidth does not always result in an increase in performance; 128kbps performs on par with that of 144kbps. The suspected reason is that both bandwidths have the same optimal window size, resulting in similar performance. [Brasche and Walke, 1997] had a similar experience where all bandwidths from 15-50kbps achieved a throughput of ~11Kbps. Simulations show that an increase in bandwidth results in a logarithmic increase in throughput. But the cost of bandwidth resource may out weigh the increase in performance. It may be better for a few users to have good performance than to have a single user have slightly better performance.

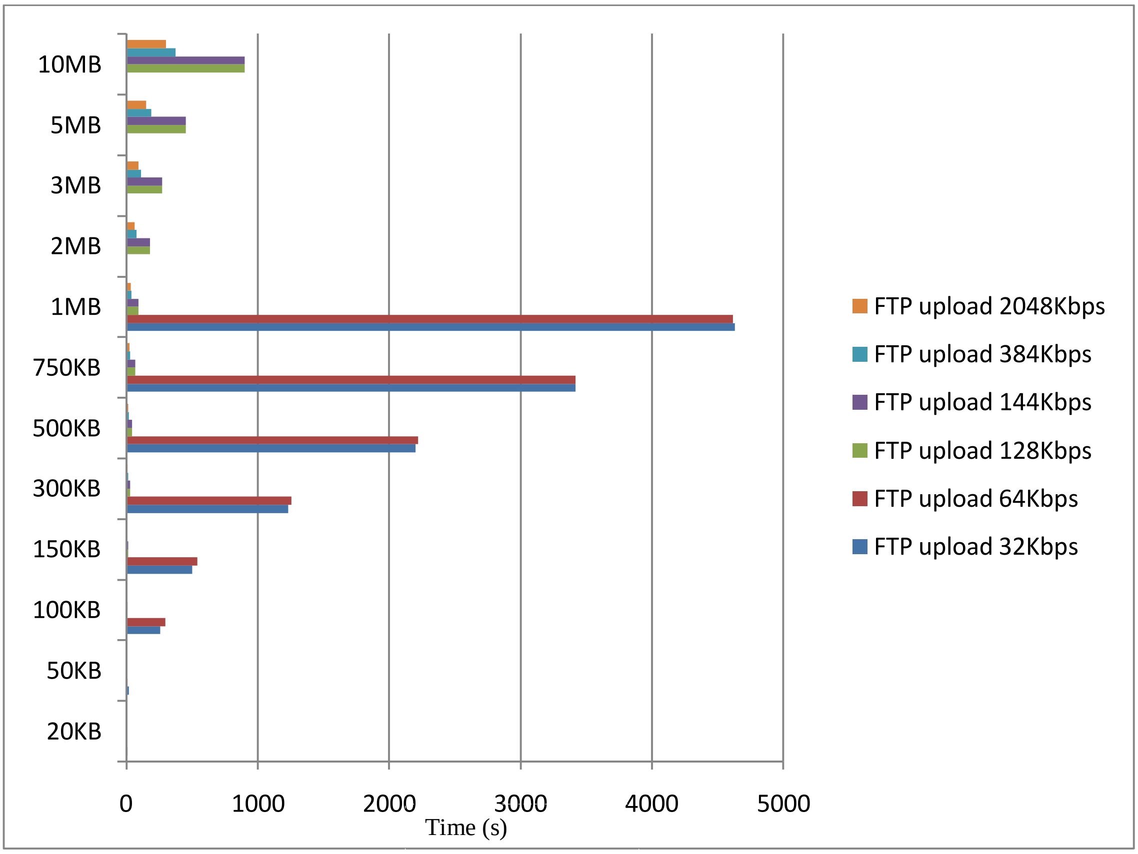

Figure 6.4 The time to upload a file by each bandwidth for each file size.

Uploading

TCP Reno was used in upload simulations. This allowed for larger file sizes to be transferred before constant RTO occurred than the use of TCP Vegas. Uploading had the same effects as downloading up until constant RTO occurs, then a huge kink in performance occurs and throughput drops to close to zero.

Case 3: The effects of changing file size while keeping end-to-end delay and bandwidth constant and how this affects time and throughput performance. This is illustrated in Figure 6.5.

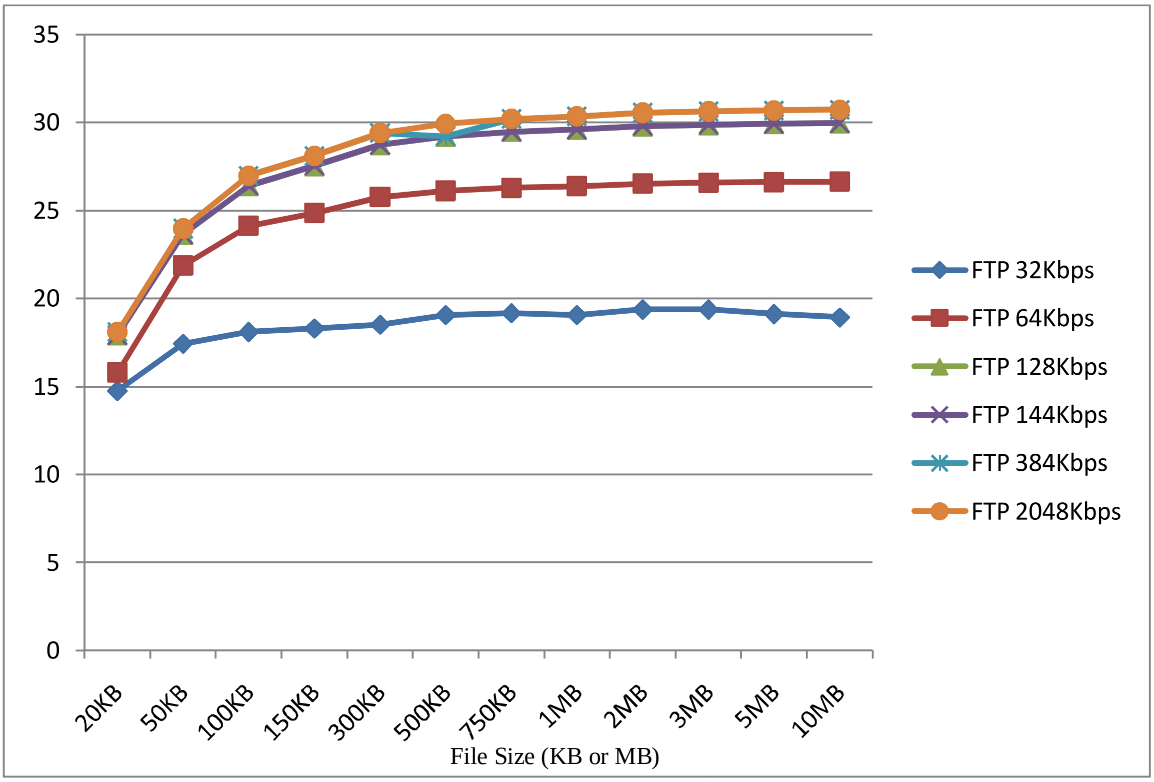

Figure 6.5 Download throughput achieved by each bandwidth.

As the size of the file to be downloaded becomes larger the throughput increases in a logarithmic fashion. All bandwidths perform similar for small files. This is because the time to download a small file is short and TCP window size does not affect the transfer. As the file size increases the throughput increases until the maximum throughput for the channel bandwidth is reached and the throughput levels out. At this stage further increases in file size will yield insignificant increases in throughput. This maximum throughput is reached for low bandwidths at smaller file sizes than for higher bandwidths and is a result of the TCP window size.

This indicates that it would be wasteful to use large bandwidth to transfer a small file. [Chakravorty et al, 2004(B)] and [Chakravorty et al, 2004(A)] discovered the same results that small files have similar performance over low and high bandwidths, whereas higher bandwidths achieve higher throughputs than lower bandwidths for large files sizes.

Case 3 (Uploading): The effects of changing file size while keeping end-to-end delay and bandwidth constant and how this affects time and throughput performance. This is illustrated in Figure 6.6.

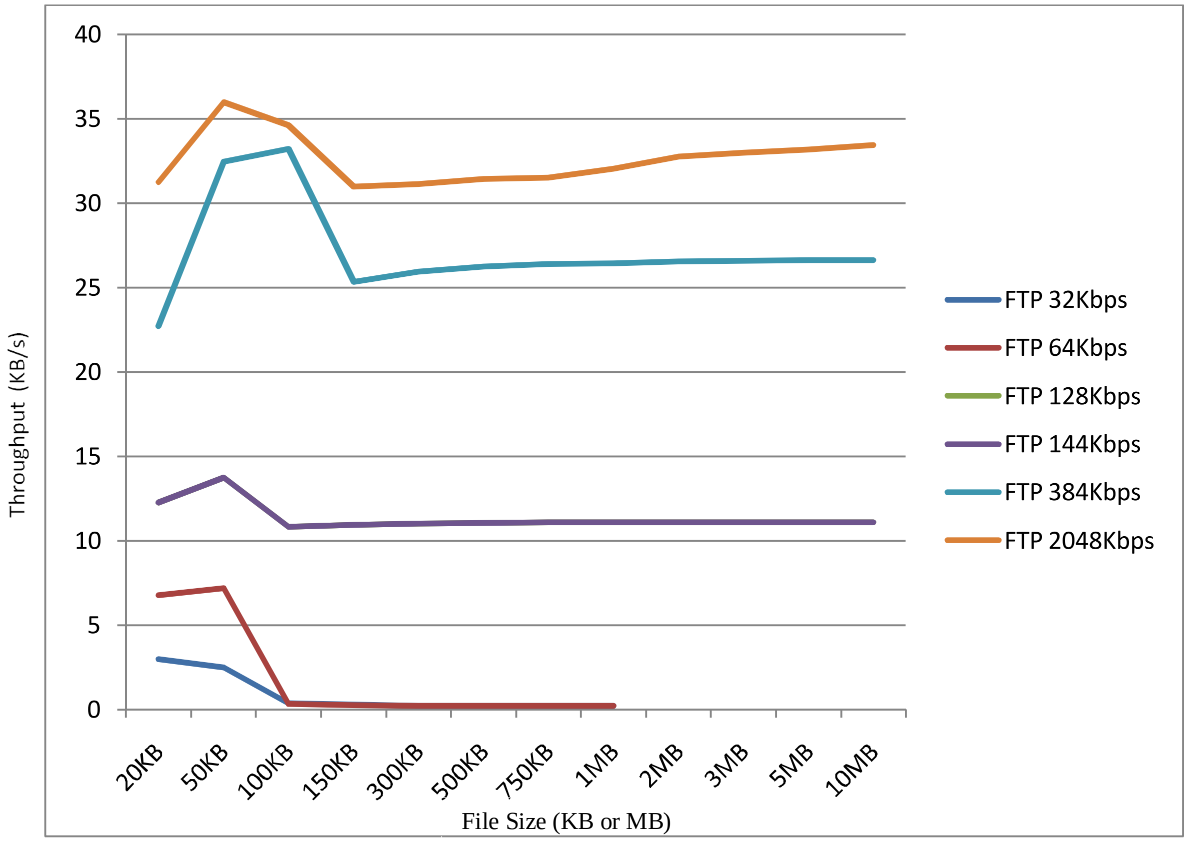

Figure 6.6 Upload throughput achieved by each bandwidth.

The upload simulations used TCP Reno, which reacts differently than TCP Vegas. [Dubois, 2005] TCP Vegas gradually grows the window size whereas TCP Reno is always transferring close to maximum window size. Which meant the file size was insignificant at affecting throughput achieved by a bandwidth. Throughput spikes early on and then decreases and levels out. If initially the window size is larger than optimal, it enables the transfer to briefly achieve throughputs higher than the optimal window size would before the TCP congestion control kicks in and regulates the transfer causing throughput to level out.

A higher bandwidth results in higher throughput, which seems more noticeable for uploading than downloading, because TCP Reno negates the effects of file sizes.

The following formula, Equation 6.3, can be used to work out the maximum number of users sharing the same service equally for a given bit rate and Eb/Io value. Eb is the bit energy and and Io is the power spectral density of thermal noise plus interference. This value is measured at the receiver when the signal is received.

At the receiver for a given data rate and a target BER, the Eb/Io can be measured. The Eb/Io value can then be used in equation 6.3 to calculate the maximum number of concurrent users for a given data rate. In Sungkasap et al the Eb/Io values were measured and the equations used to calculate Table 6.5, which can then be used to determine the maximum number of simultaneous users. M is the maximum number of simultaneous users, F is the ratio of interference from neighbouring cells, and C/I is the signal to interference ratio. R_{chip} is chip rate in cps, and R_{user} is the information bit rate for the service in bps. [Sungkasap et al, 2008]

$$\text{(Equation 6.3) } M = 1 + 1/[(1 + F)(C/I)]$$

$$\text{(Equation 6.4) } C/I = [10^{(Eb/Io)/10}]/[R_{chip}/R_{user}]$$

Table 6.5 Maximum number of users for a given Eb/Io and bit rate

Eb/Io	Bit Rate kbps	Max users
8.1	12.2	30.8
6.8	64	7.5
6.0	128	4.9
6.0	384	2.3
6.0	2048	1

Based on simulation results and maximum users for a given bit rate, 64kbps will be the best bit rate to offer from a service provider's perspective, its performance is close to the higher data rates but more users will be able to enjoy the service simultaneously.

6.2.5 Conclusion

The effect of network factors on the performance of the FTP application was investigated. Simulations were designed which showed how network factors: end-to-end delay, bandwidth and file size affected FTP performance. It was found that a short end-to-end delay always improved the service and a higher bandwidth too always improved the service. The size of file in combination with bandwidth affected the performance. This indicates that the best case for a FTP application will be low end-to-end network delay and high bandwidth.

A higher bandwidth is capable of a higher maximum data rate. To improve the FTP service to allow for higher throughput a higher bandwidth can be used. The return on throughput for an increased bandwidth is logarithmic and it would be better for a network to assign a medium level bandwidth to each user, in such maximising the cells throughput and performance.

The affects of file size on FTP shows that when a small file is downloaded all bandwidths perform the same. This is because TCP is unable to make use of higher bandwidths as the increase in window size occurs to slowly. This indicates that using high bandwidths for small file transfers are wasteful. While in a large file transfer the high bandwidth performs much better than the low bandwidth as TCP is able to increase each window size to its optimal value. A higher bandwidth will have a larger window size and be able to transfer data at a higher rate.

Where the bottleneck occurs in the end-to-end delay affects the performance. If the bottleneck is at the receiving node it has the greatest possibility of causing RTO, because the receiving node will take long to send an acknowledgement to the sender, causing the sender to timeout and retransmit the packet. When the bottleneck is at the sender, it is possible for the receiver to promptly send an acknowledgement to the sender node, which mitigates chance of a RTO from occurring. When TDMA is used in a cell to increase the number of simultaneous users; the base station controller needs to ensure that all users obtain a transfer slot often enough to prevent constant RTO from occurring.

The FTP service will be usable under just about all network conditions, as long as constant RTO does not occur. It is suggested that a cell should use all extra available bandwidth to improve a FTP download. The factor which affects results the most would be bandwidth. Based on the simulation results and number of simultaneous users for a given bandwidth (Table 6.5), it is suggested that the 64Kbps data channel be used as this would maximise network capacity, while not causing constant RTO.

6.3 Email Experiments

6.3.1 Network Setup

The email application was divided into 2 stages. First stage: the MS creates an email and sends the email across the UMTS network to an email server which stores it. Second stage a receiving MS connects to the email server and downloads the email to view it. A EURANE UMTS network was setup which connects to the Internet, furthermore the network is setup as detailed in the Common Implementation Section 6.1.2 with the application server being an email server. The email simulation setup is depicted in Figure 6.7.

6.3.2 Parameters and Protocols

In stage one (uploading emails) a connection is setup between the sending MS and the Email server; to simulate this process a NS2 FTP application was used. The message being transferred represents the email plus any attachments. All other details are as specified in the Common Implement Details 6.1.3. The file sizes simulated are 20KB, 50KB, 100KB, 150KB, 300KB, 500KB, 750KB, 1MB, 2MB, 3MB, 5MB and 10MB. These file sizes were simulated for each bandwidth. TCP Reno was used for the uploading of files from a mobile device to a server. The second stage is the same as the first stage except emails were downloaded from the server to the mobile device and TCP Vegas was used. A summary of the email application's parameters are presented in Table 6.6.

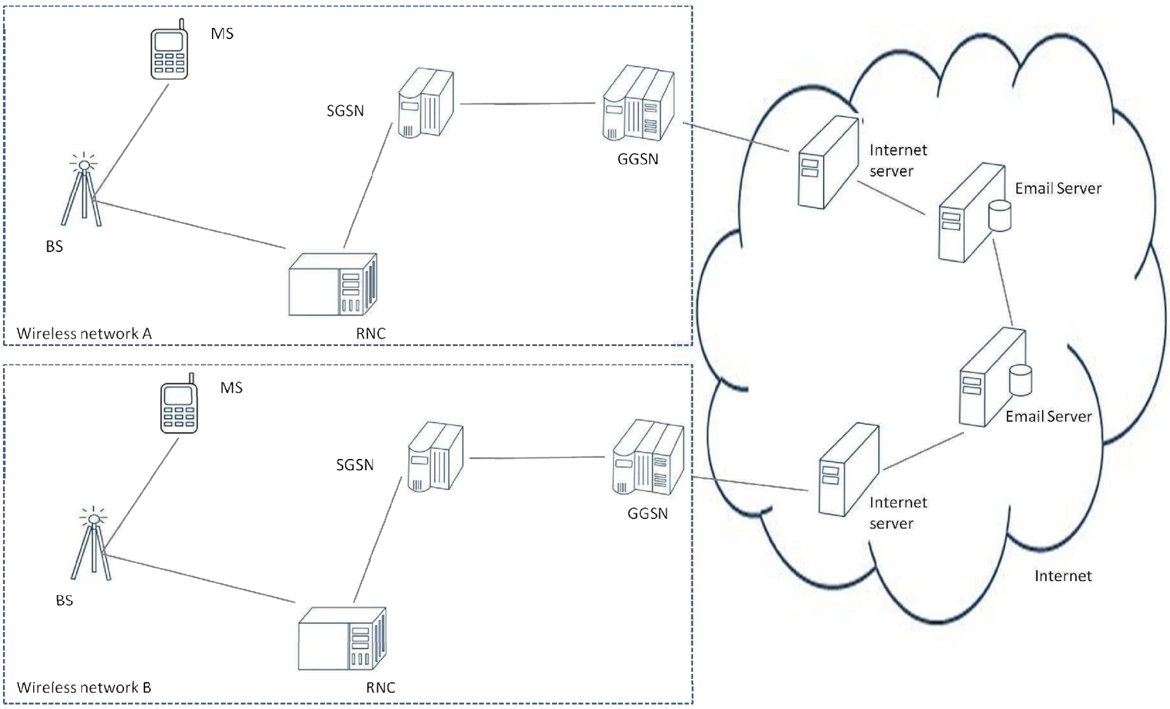

Figure 6.7 The network used for email simulations.

6.3.3 Method

The method is identical to that detailed in Common Implementation Section 6.1.5 with the target application server being an email server.

Table 6.6 Email application parameters.

Parameter	**Value**
Email packet size	1024bytes
Email message size	20KB – 10MB
Transport protocol	TCP
Air interface	WCDMA
Air interface bandwidth	16Kbps – 2048Kpbs
TCP	Vegas (download), Reno (Upload)

6.3.4 Results and Analysis

All cases look at the performance of the entire email process, which includes both the upload and download stages as a single email transfer process. The performance is measured for this transfer process.

Case 1: The effects of changing end-to-end delay while keeping bandwidth and message size constant and how this affects time and throughput performance.

The email server processing time was measured by sending an email between a wired sender and receiver. The processing time was measured to be 4 seconds.

Uploads were done using TCP Reno and downloads were done using TCP Vegas. Time for upload and download stages were shorter for a shorter end-to-end delay. Therefore a shorter end-to-end delay improves email performance.

In the symmetric setup used by the simulations where the upload and download bandwidth is the same, it was found if one stage suffers from constant RTO then the whole email process is affected. The poor performing stage will dominate the time taken to transfer and cause the entire transfer process to perform poorly. In scenario two of the 1Mb message transfer the uploading stage causes the bulk of the time taken to transfer the message, and so affected the entire transfer performance.

Packets take a shorter time to propagate across a short delay network than a long delay network and resends for a damaged or lost packet will occur more quickly. A long delay is helped when TCP increases the window size which then negates the effects of a longer end-to-end delay. But it was still found that a shorter delay performed better. The simulations were done for all window sizes and a short end-to-end delay outperformed a long end-to-end delay. [Roccetti et al, 2005] found that it took longer to transfer data from a geographically further server which also had a higher end-to-end delay.

Table 6.7 the affects of changing end-to-end delay while keeping bandwidth and message size constant on time and throughput performance. The message size is 1MB (and 20KB) and bandwidth is 2048Kbps.

In real networks the uploading time will dominate the time to transfer a message as a mobile device transmits at a lower rate than it receives. This is because of design, function and power. Device manufacturers build the device to a higher receiving rate than transmitting rate. Also the operation of downloading is more common than uploading and finally to transmit at high data rates uses considerable more power than at low data rates [The Shosteck Group, 2001].

Table 6.7 The effects of changing end-to-end delay while keeping bandwidth and message size constant on time and throughput performance. The message size is 1MB (and 20KB) and bandwidth is 2048Kbps.

Message size	Scenario	Upload time (s)	Server processing (s)	Download time (s)	Through put KB/s	Total Time
20Kb	1	2.2	4	9.9	1.65	12.1
	2	1.9	4	2.7	4.35	4.6
	3	0.6	4	1.1	7.41	2.7
1Mb	1	46.1	4	438.2	2.05	288.3
	2	4655.4	4	35.1	0.21	4694.5
	3	31.2	4	33.0	14.66	68.2

Case 2: The effects of changing bandwidth while keeping end-to-end delay and message size constant and how this affects time and throughput performance. Illustrated in Figure 6.8.

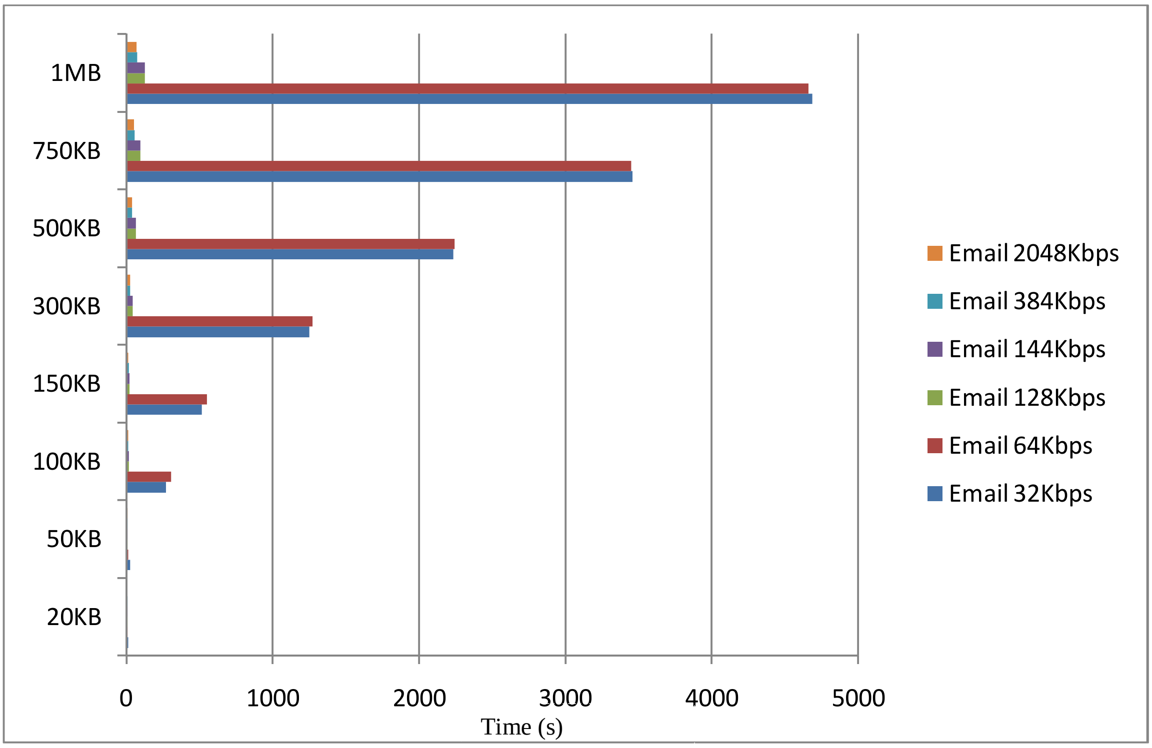

Figure 6.8 The time to transfer an email file by each bandwidth for each message size.

An increase in bandwidth leads to an increase in performance. When the bandwidth increases, the time to transfer the email message decreases while the throughput increases. In the uploading stage bandwidths of 64Kbps and below experience constant RTO for message transfers of larger than 100KB causing the transfer to take very long. When only viewing bandwidths higher than 64Kbps a marked difference in performance can be seen, see Figure 6.8. The 384Kbps bandwidth has a throughput of 14Kbps while 144Kbps bandwidth has a throughput of 8Kbps for a 1Mb message and takes about half the time to transfer the message.

An increase in bandwidth results in better performing uploads and downloads, because TCP and the larger bandwidth together allow for a higher throughput. See Section 6.2.4 Case 2 for details. [UK Telematics online, no date] mentions that it will take the low GPRS bandwidth

120 seconds to download a file that the high 3G bandwidth could do in 10 seconds; referring to the high difference in bandwidth and its effect on performance.

Case 3: The effects of changing message size while keeping end-to-end delay and bandwidth constant and how this affects time and throughput performance. Illustrated in Figure 6.9.

The larger the message size the higher the throughput achieved by the bandwidth. Throughput increase is logarithmic with respect to increase in message size. All bandwidths perform similar for small messages. This is because the time to transfer a small message is short and TCP window size does not affect the transfer. As the message size increases the throughput increases until the maximum throughput for the channel bandwidth is reached and the throughput levels out. TCP controls this maximum window size based on bandwidth and channel conditions. Once throughput levels out, further increases in message size will yield insignificant increases in throughput. This maximum throughput is reached for low bandwidths at smaller message sizes than for higher bandwidths.

This indicates that it would be wasteful to use a large bandwidth to transfer a small message. [Chakravorty et al, 2004(A)] discovered the same results that small files have similar performance over low and high bandwidths, whereas higher bandwidths achieve higher throughputs than lower bandwidths for large files sizes.

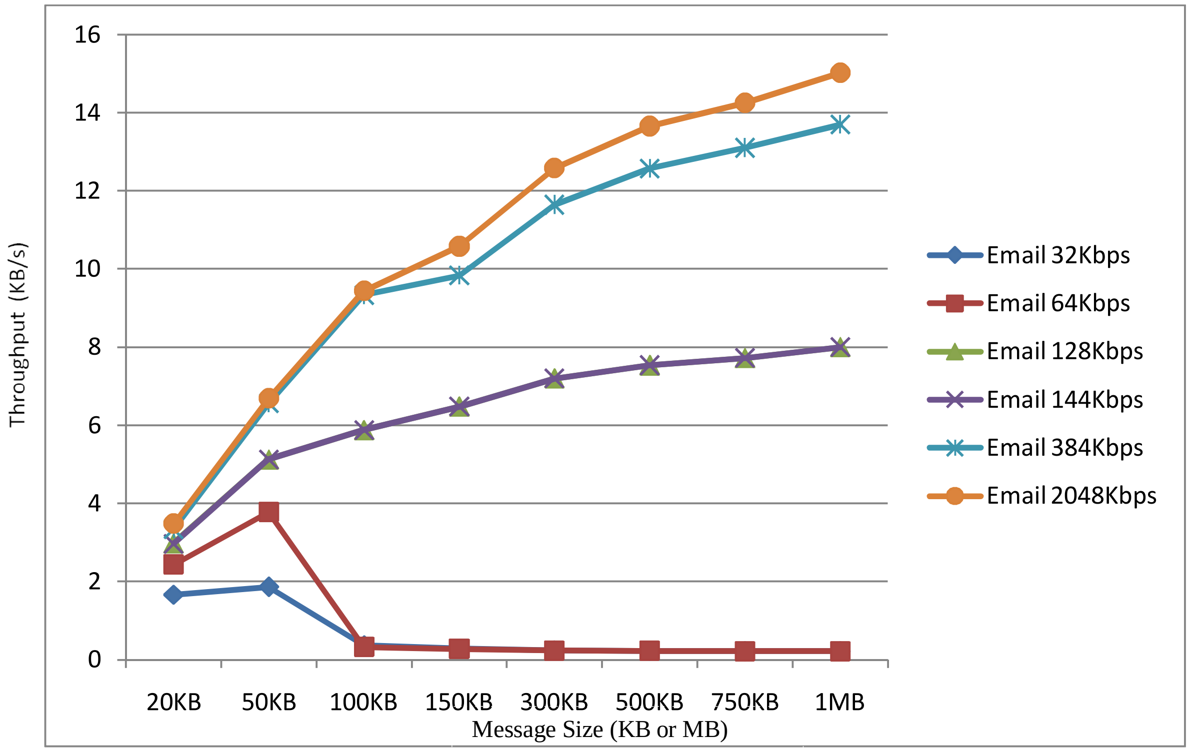

Figure 6.9 Transfer throughput achieved by each bandwidth.

The graph shows that for large bandwidths as the message size increases throughput continues to grow. The graph shape is logarithmic. The 128Kbps/144Kbps begin to taper after 500KB, while the larger bandwidths continue to grow. The 64Kbps and 32Kbps channels rapidly decrease in throughput as message size becomes larger than 50Kb due to constant RTO. [Dubois, 2005] discusses the occurrence of constant RTO and how increases in RTO can occur.

Based on simulation results and maximum users for a given bit rate displayed in Table 6.5 in Section .6.2.4 it is recommended that channel rate 64kbps is used for small emails while 128Kbps channel rate be used for large emails. Its performance is close to the higher data rates but more users will be able to enjoy the service simultaneously.

6.3.5 Conclusion

How network factors affected the performance of email was investigated. Simulations were designed which showed how network factors: end-to-end delay, bandwidth and message size affect email performance. It was found that a short end-to-end delay always improved the service and a high bandwidth always too improved the service. The size of message in combination with bandwidth affected the performance. This indicates that the best case for an email application will be a low end-to-end network delay and high bandwidth.

A higher bandwidth is capable of a higher maximum data rate. To improve the email service to allow for higher throughput a higher bandwidth can be used. The return on throughput for an increased bandwidth is logarithmic and it would be better for a network to assign a medium level bandwidth to each user, thus maximising the cell's throughput and performance.

The effects of message size on email showed that when a small email was sent all bandwidths performed the same. This is because TCP is unable to make use of higher bandwidths as the increase in window size occurs too slowly. This indicates that using high bandwidths for small emails are wasteful. While in large emails the high bandwidth performs much better than the low bandwidth as TCP is able to increase each window size to its optimal value. A higher bandwidth will have a larger window size and be able to transfer data at a higher rate.

Where the bottleneck in the end-to-end delay occurs affects the performance. If the bottleneck is at the receiving node it has the greatest possibility of causing RTO. This is because the receiving node will take long to send an acknowledgement to the sender, causing the sender to timeout and retransmit the packet. When the bottleneck is at the sender, it is possible for the receiver to promptly send an acknowledgement to the sender node, which minimises the chance of a RTO occurring. When TDMA is used in a cell to increase the number of simultaneous users, the base station controller needs to ensure that all users obtain a transfer slot often enough to prevent constant RTO from occurring.

The email service will be usable under just about all network conditions, as long as constant RTO does not occur. It is suggested that a cell should use all extra available bandwidth to improve email transfers. The factor which affects results the most would be bandwidth.

Based on the simulation results and the number of simultaneous users for a given bandwidth (Table 6.5), it is suggested that the 128Kbps channel be used for larger emails and the 64kbps channel be used for smaller email messages, as this will maximise network capacity, while not causing constant RTO.

6.4 SMS Experiments

6.4.1 Network Setup

To Simulate an SMS application we have divided the SMS service into two stages. First stage: the MS sends the message to the UMTS network which stores it. Second stage the UMTS network forwards the message to the recipient MS. A EURANE UMTS network was setup, which contained two extra nodes in the core network representing the Short Message Service Switching Centre (SMSSC). The GGSN links to the SMSSC where the message will be stored before being forwarded to the recipient. The rest of the setup details are the same as described in Section 6.1.2, below is a diagram of the network described. See Figure 6.10. The exact same network setup is used to forward the SMS to the recipient in the second stage of SMS service.

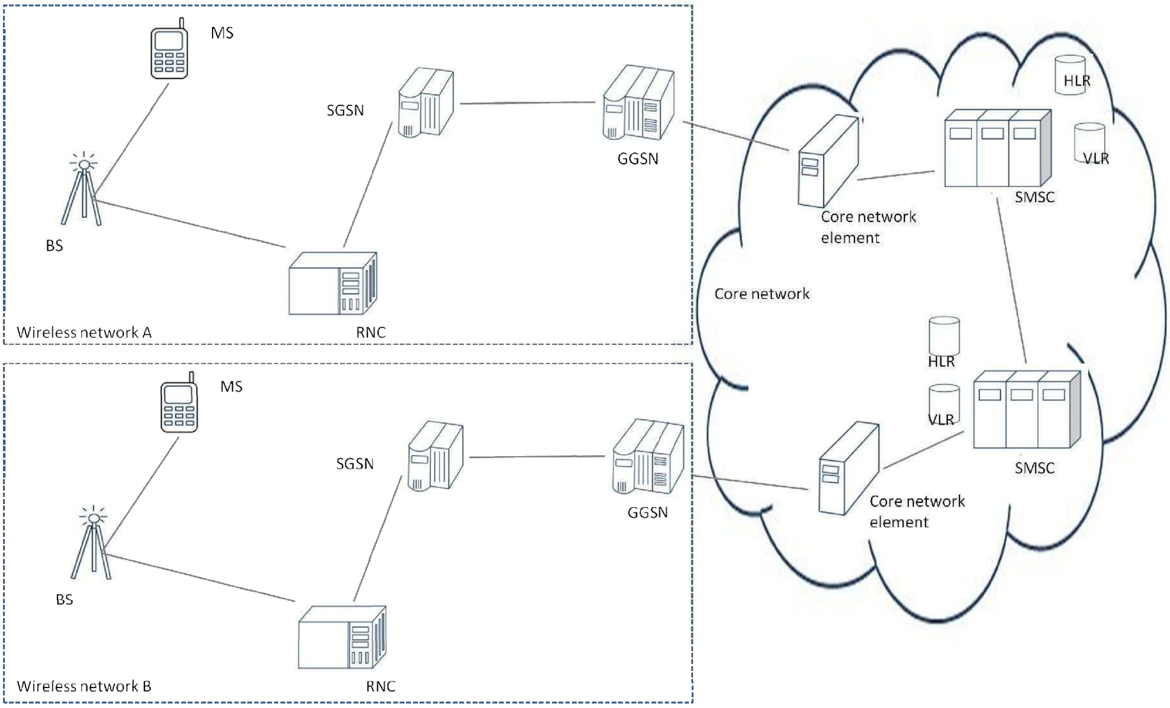

Figure 6.10 The network used for SMS simulations.

6.4.2 Parameters and Protocols

SMS data is transferred over the signalling channels and has a small payload size, containing a maximum of 160 7 bit characters. SMS was simulated using EURANE data channels but setup in a way that better represents the signalling channel. A SMS of 160 characters is represented by a 200 byte packet which includes payload and any overhead incurred by network delivery. A single part SMS, two part SMS and three part SMS were simulated, each SMS part is one packet of data. A one part SMS = 160 characters and three part SMS = 450 characters. In stage one the data was sent from a MS to the SMSC and stored, in stage two the SMSC forwarded the data to the recipient MS, these two stages together simulated the time taken for an SMS to be sent. FTP packet size was 200 bytes and the number of packets was set from one to three. This ran on the TCP connection setup between the source and destination nodes see Common Implementation Details 6.1.3.

The simulation was setup as if SMS parts were delivered sequentially from source to destination. This is because a signalling channel's payload is only large enough to deliver a single SMS part at a time.

Table 6.8 SMS application parameters.

Parameter	Value
packet size	200bytes
SMS parts	1-3
Air interface	WCDMA
Air interface bandwidth	16Kbps – 2048Kbps
TCP	TCP Vegas

6.4.3 Method

The method is identical to that detailed in Common Implementation Section 6.1.5 the target application server would be the SMSC within the core network. Window size is fixed at one in SMS simulations since SMS messages are sent separately across signalling channels.

6.4.4 Results and Analysis

Case 1: The effects of changing end-to-end delay while keeping bandwidth and SMS message size constant and how this affects time and throughput performance. The bandwidth is set at 64Kbps and SMS message size is 160 characters.

A shorter end-to-end delay causes a shorter SMS message transfer time and an increase in throughput. The increase in throughput is insignificant because a SMS message is very small and a user will not notice the improvement. Also the absolute time taken to transfer an SMS is small 3.8 seconds for the long delay network and 3.3 seconds for the short delay network. The half second difference is too small for the user of the service to notice the performance improvement. A SMS consists of a single packet transfer across the signalling channels.

The simulations transferred a single 200 byte packet including over head from the sending node to the receiving node and the network delay affected the packets propagation time. The shorter the propagation time (end-to-end delay) the quicker the SMS packet will be transferred. The difference in network end-to-end delays of a long delay network compared to a short delay network are only a few hundred milliseconds so the difference in time transfer of a single packet SMS will also only be a few hundred milliseconds. The end-to-end delay is insignificant to the performance of SMS.

Table 6.9 The effects of changing the end-to-end delay while keeping message size and bandwidth constant and how it affects time and throughput performance. The SMS size is 160 characters and the bandwidth is 64Kbit/s.

Scenario	Upload time (s)	Network processing (s)	Download time (s)	Throughput KB/s	Total Time
1	0.4	3	0.4	0.05	3.8
2	0.4	3	0.4	0.05	3.8
3	0.1	3	0.2	0.06	3.3

The results to send a message falls within the range found by [Koumpis et al, 1999] of 2-16 seconds. GPRS end-to-end delay is in the region of 600ms and 3G in the region of 200ms. The SMS service will remain unchanged in 3G compared to GPRS. These results backup why users have not noticed an improved in the SMS service in 3G compared to that of GPRS networks.

Case 2: The effects of changing bandwidth while keeping end-to-end delay and SMS message size constant and how this affects time and throughput performance. Illustrated in Figure 6.11.

All the bandwidths take the same time to send an SMS and have the same throughput. The small changes in time and throughput in the graphs are caused by NS2 having fractionally shorter delays for higher bandwidths. A SMS message consists of 140 byte payload and any bandwidth used is capable of delivering this message in a single packet transfer. Hence all bandwidths with the same end-to-end delay will take the same time to deliver a SMS message.

This renders the size of the bandwidth irrelevant, justifying the sending of SMS messages across shared signalling channels as a SMS requires very little resources. This is likely the reason why subscribers do not notice an improvement in SMS performance with the introduction of 3G.

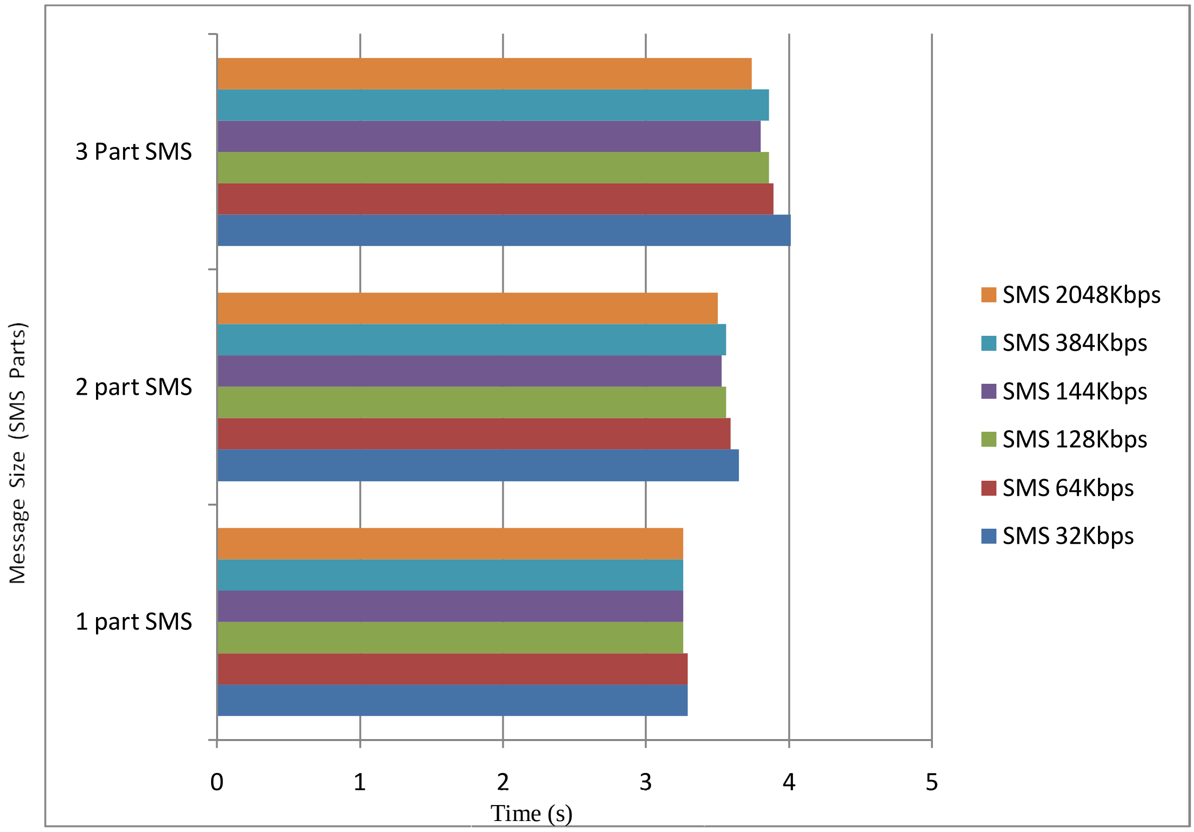

Figure 6.11 The time to transfer an SMS by each bandwidth for each SMS size.

Case 3: The effects of changing SMS message size while keeping end-to-end delay and bandwidth constant and how this affects time and throughput performance. Illustrated in Figure 6.12.

In a hypothetical case that the SMS message could consist of multiple parts that are transferred across the same connection, the number of message parts would not affect the performance. This is because the size of the data being transferred is very small and not enough to allow a distinction in bandwidth performances to occur. The SMS service requires few resources and this is likely why SMS was the first data service.

In simulations message parts were sent sequentially, with a window size of one because the signalling channel path only has a payload of 140 bytes, large enough for one SMS message part. Increasing the number of SMS message parts increased the time taken to deliver the SMS message (SMS message equal all message parts), but had an insignificant effects on throughput.

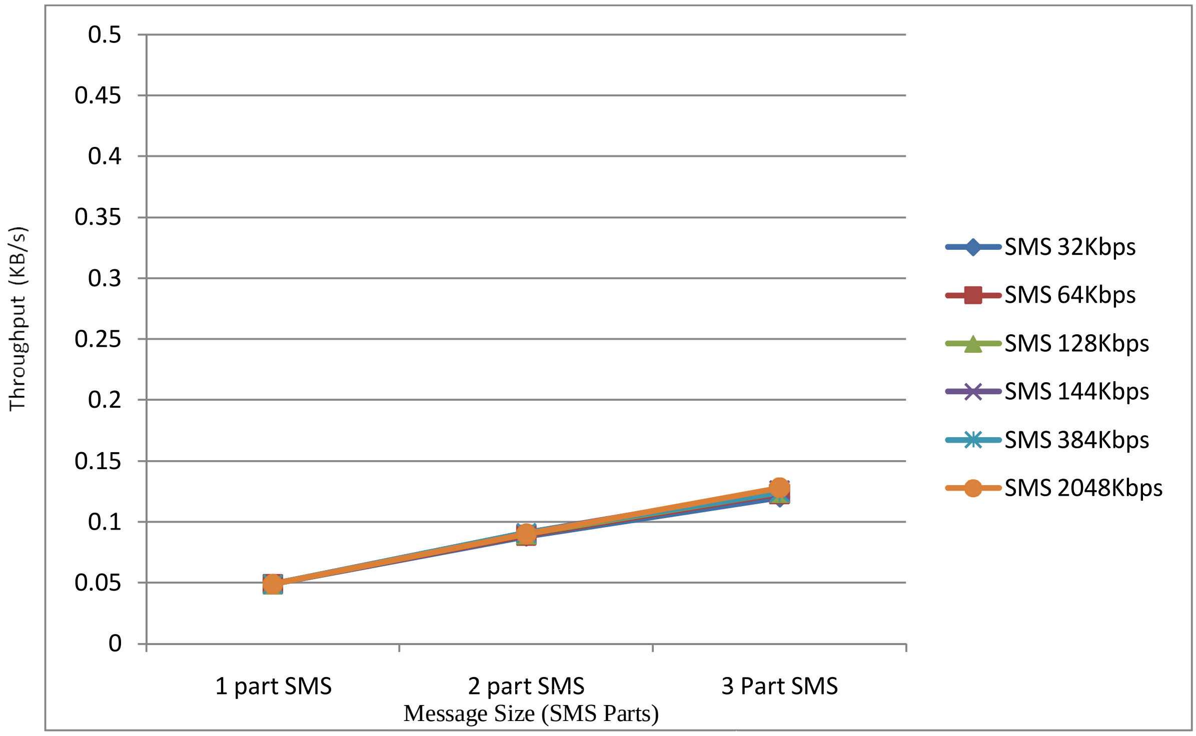

Figure 6.12 Transfer throughput achieved by each bandwidth.

The real time experienced by a user is very small and the user will not notice the increase in time. It is noted that in a real network each message part will be delivered independently from each other. A mobile device can only have one signalling channel connection at a time. A signalling channel connection is established and one message part is delivered and the connection is closed. The same then happens for the second and third message parts. Message parts could be delivered in any order as they are delivered completely separately from each other. The small size of SMS messages justifies the use of signalling channels for its transfer.

6.4.5 Conclusion

How network factors affected the performance of SMS was investigated. Simulations were designed which showed how network factors: end-to-end delay, bandwidth and SMS message size affected SMS performance. It was found that none of the factors had any significant affect on the SMS service. A shorter end-to-end delay only improved message delivery time by a few hundreds of a second. This is because of the nature of a SMS message.

The SMS service is a low resource service consisting of messages of 140 bytes, this size message can easily be transferred by any of the 16-2048Kbps bandwidths in a single transfer. A single transfer difference between a long end-to-end delay network and a short end-to-end delay network will only be a few hundreds of a second. This time difference is insignificant to the user of the service and renders end-to-end delay as an insignificant factor in the SMS service. The same can be said of bandwidth as all bandwidths will take a single transfer for a SMS message. Bandwidth will not affect the SMS service.

Multiple SMS messages are sent independent of each other, and so the size of the SMS message will not affect the performance. If the messages were sent together the size will still not affect performance as the payload of 3 SMS messages total a size of 420 bytes, which is small enough for all bandwidths 16-2048Kbps to handle. The low resource intensity of the SMS service will mean users will not notice any improvement in the service in a 3G network. It also justifies the use of the signalling channels to transfer SMS messages.

To maximise the performance of the SMS service, it should continue to function as is, by transferring SMS messages across signalling channels as soon as a signalling channel becomes available.

6.5 MMS Experiments

6.5.1 Network Setup

MMS application simulations were divided into two stages. First stage was the MS sent the message to the UMTS network which stored it. Second stage the UMTS network forwarded the message to the recipient MS. A EURANE UMTS network was setup which contained two extra nodes in the core network representing the Internet and Multimedia Subsystem (IMS). The GGSN linked to the IMS where the messages were stored before being forwarded to the recipient. The rest of the setup details are the same as described in the Common Implementation Section 6.1.2, below is a diagram of the network described. See Figure 6.13.

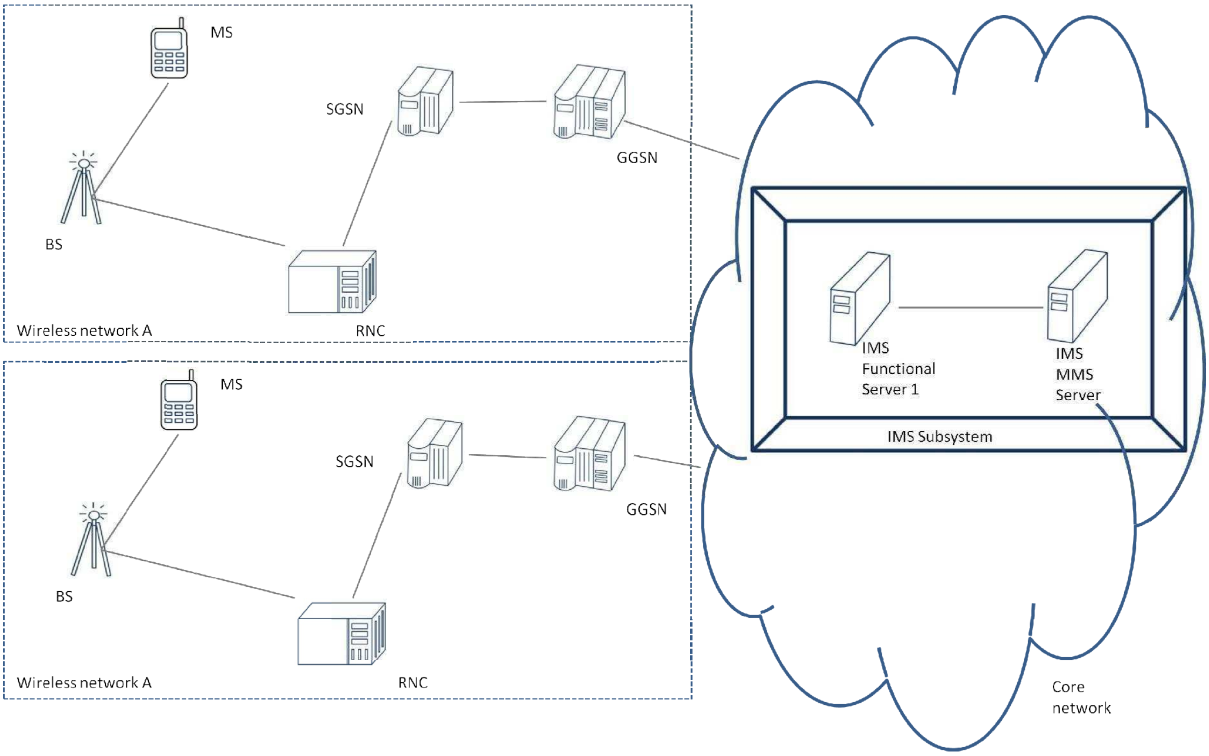

Figure 6.13 The network used for MMS simulations.

6.5.2 Parameters and Protocols

In stage one a connection was setup between the sending MS and the IMS server; to simulate this process a NS2 FTP application was used as described in Common Implementation Details 6.1.3. In the second stage of the MMS simulation the MS downloaded the MMS message from the IMS server. The setup was the same as stage one. The message sizes simulated were 20KB, 50KB, 100KB, 150KB, 300KB, 500KB, 750KB and 1MB.

6.5.3 Method

The method is identical to that detailed in Common Implementation Section 6.1.5 with the target application server being the MMC component of the IMS within the core network.

Table 6.10 MMS application parameters.

Parameter	Value
FTP packet size	1024bytes
Number of FTP packets	20 – 1000
Transport protocol	TCP
Air interface	WCDMA
Air interface bandwidth	16Kbps – 2048Kbps
TCP	Vegas

6.5.4 Results and Analysis

Case 1: The effects of changing end-to-end delay while keeping bandwidth and MMS message size constant and how this affects time and throughput performance.

TCP Reno was used for the upload simulations and TCP Vegas for the download simulations. Simulations show a short end-to-end delay caused a shorter time to transfer a MMS message and a higher throughput achieved, implying a shorter end-to-end delay increases MMS application performance. If the uploading or downloading suffers from constant RTOs then that stage makes up the bulk of the time taken for the transfer and affects the performance of the entire transfer process. A shorter end-to-end delay results in a shorter packet propagation and resend time as explained in previous application simulation Sections 6.2-6.4 Case 1, see these for details.

TCP negates the effects of a longer end-to-end delay by increasing the window size and allowing for more packets to be in the transfer state at a time. In simulations all window sizes were simulated for all networks delays; and in the end a short end-to-end delay network performed best. [Roccetti et al, 2005] found that it took longer to transfer data from a geographically further server which had a longer end-to-end delay.

Table 6.11 The effects of changing the end-to-end delay while keeping MMS message size and bandwidth constant and how it affects time and throughput performance. The MMS message size is 300KB and the bandwidth is 384Kbit/s and 128Kbit/s.

Bandwidth	Scenario	Upload time (s)	Network processing (s)	Download time (s)	Throughput KB/s	Total Time
128Kbps	1	1745.3	6	132.2	0.16	1883.5
	2	1040.4	6	15.9	0.28	1062.3
	3	27.2	6	10.4	6.88	43.6
384Kbps	1	14.8	6	132.2	1.96	153.0
	2	11.2	6	12.4	10.14	29.6
	3	11.6	6	10.2	10.79	27.8

Case 2: The effects of changing bandwidth while keeping end-to-end delay and MMS message size constant and how this affects time and throughput performance. Illustrated in Figure 6.14.

An increase in bandwidth leads to an increase in performance. When the bandwidth increases the time to send a MMS message decreases while the throughput increases. The effects of bandwidth are less on small files, 100KB and smaller, than on large files greater than 100KB. The throughput of 384Kbps bandwidth was close to 12KBps while 144Kbps bandwidth was close to 7KBps and the times taken were 25 seconds and 42 seconds respectively, for a 300KB file. It can be seen that 384Kbps bandwidth performed almost twice as well as the 144Kbps bandwidth.

An increase in bandwidth results in better performing uploads and downloads, because TCP and the larger bandwidth together allow for a higher throughput. The increase in bandwidth gives a logarithmic increase in performance. See Section 6.2.4 Case 2 for details. But this increase in performance comes at a cost, the higher the bandwidth the lower the number concurrent users of the service.

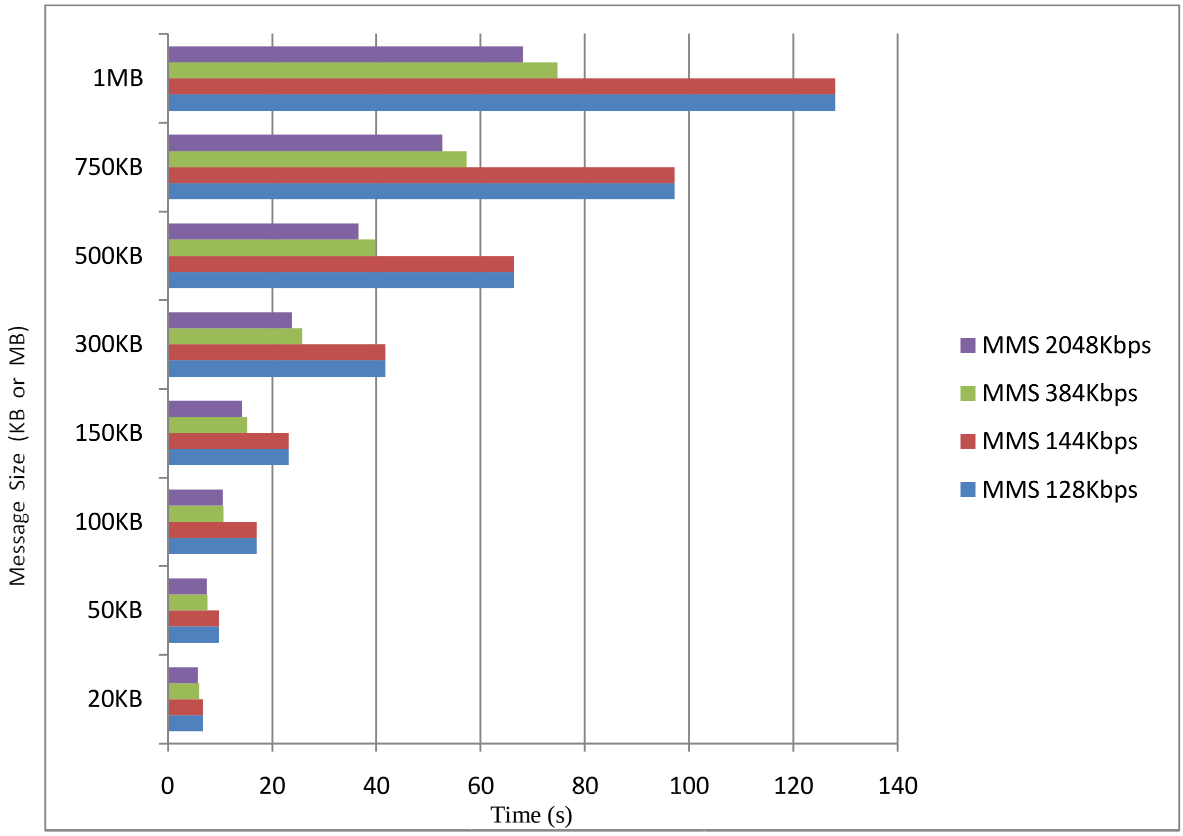

Figure 6.14 The time to transfer a MMS by each bandwidth for each message size.

Case 3: The effects of changing MMS message size while keeping end-to-end delay and bandwidth constant and how this affects time and throughput performance. This is Illustrated in Figure 6.15.

The larger the message size the higher the throughput achieved by a bandwidth. Throughput increase is logarithmic with respect to increase in message size. All bandwidths perform similarly for small messages. This is because the time to download a small message is short and the TCP window size does not affect the transfer. As the message size increases the throughput increases until the maximum throughput for the channel bandwidth is reached and the throughput levels out. TCP controls this maximum window size based on bandwidth and channel conditions. Once throughput levels out, further increases in message size will yield insignificant increases in throughput. This maximum throughput is reached for low bandwidths at smaller message sizes than for higher bandwidths.

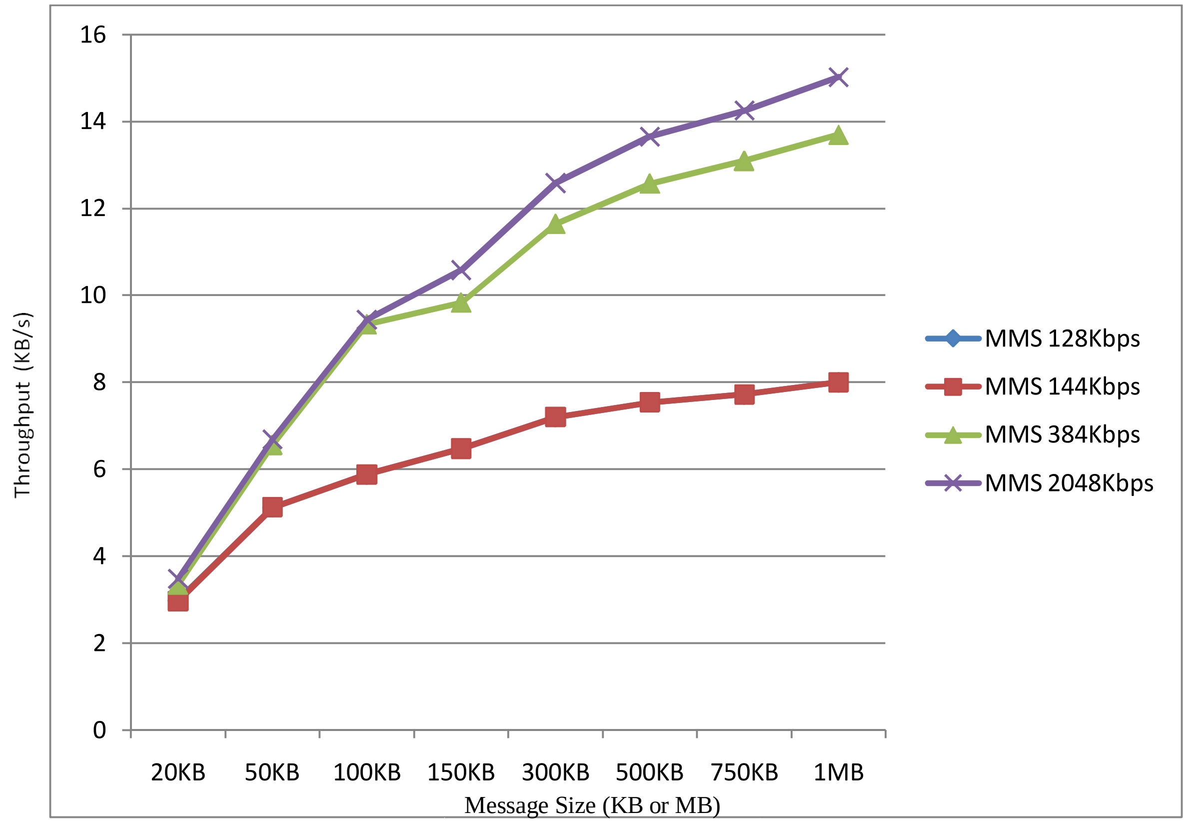

Figure 6.15 Transfer throughput achieved by each bandwidth.

This indicates that it would be wasteful to use large bandwidths to transfer a small message. [Chakravorty et al, 2004(A)] discovered the same results that small files have similar performance over low and high bandwidths, whereas high bandwidths achieved higher throughputs than low bandwidths for larger files sizes. It is recommended that the lowest bandwidth that does not cause RTO be used for the MMS service.

6.5.5 Conclusion

How network factors affected the performance of MMS was investigated. Simulations were designed which showed how network factors: end-to-end delay, bandwidth and message size affected MMS performance. It was found that a short end-to-end delay always improved the service and a high bandwidth too always improved the service. The size of the message in combination with bandwidth affected the performance. This indicates that the best case for the MMS service will be low end-to-end network delay and high bandwidth.

A higher bandwidth is capable of a higher maximum data rate. To improve the MMS service to allow for higher throughput a higher bandwidth can be used. The return on throughput for an increased bandwidth is logarithmic and it would be better for a network to assign a medium level bandwidth to each user, in such maximising a cell's throughput and performance.

The effect of message size on MMS shows that when a small MMS message is sent all bandwidths perform similarly. This is because TCP is unable to make use of the higher bandwidths as the increase in window size occurs to slowly. This indicates that using high bandwidths for small MMS messages are wasteful. While in large MMS messages the high bandwidth performs much better than the low bandwidth as TCP is able to increase each window size to its optimal value. The higher bandwidth will have a larger window size and able to transfer data at a higher rate.

Where the bottleneck in the end-to-end delay occurs affects the performance. If the bottleneck is at the receiving node it has the greatest possibility of causing RTO. The receiving node will take long to send an acknowledgement to the sender, causing the sender to timeout and retransmit the packet. When the bottleneck is at the sender, it is possible for the receiver to promptly send an acknowledgement to the sender node, which mitigates chance of a RTO from occurring. When TDMA is used in a cell to increase the number of simultaneous users; the base station controller needs to ensure that all users obtain a transfer slot often enough to prevent constant RTO from occurring.

The MMS service will be usable under all network conditions, as long as constant RTO does not occur. It is suggested that a cell should use all extra available bandwidth to improve MMS message transfers. The factor which affects results the most would be bandwidth. Based on the simulation results and the number of simultaneous users for a given bandwidth; it is suggested that the 128Kbps channel be used as this will maximise the networks capacity, while not causing constant RTO, based on Table 6.5.

6.6 HTTP Experiments

6.6.1 Network Setup

To simulate a HTTP web browsing application a NS2 UMTS network was setup and connected to a wired network, the wired network represented the Internet and contained a Web server. The GGSN linked the UMTS network to a wired network such as the Internet. This wired network contained Node1 and Node2, which represented two servers on the Internet which are linked to each other. Link and data rate details are described in Section 6.1.2 Common Implementation Details. See Figure 6.16.

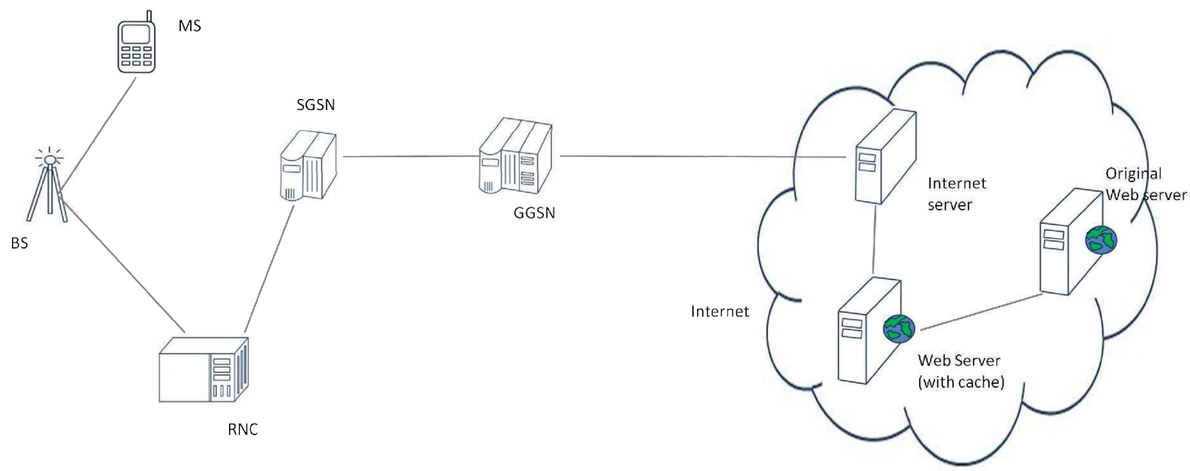

Figure 6.16 The network used for HTTP web browsing simulations.

6.6.2 Parameters and Protocols

An NS2 HTTP application instance was attached to the simulation network to simulate the web browsing service. The web server contained a HTTP application on which the size of the web page could be specified in bytes, the caching option of the web server, the probability of caching and the page expiry time could be set. The HTTP application then controlled the length of time taken to transfer a page based on the size of the page, if it was cached and the bandwidth of the channel it was being transferred over. Internally the HTTP application uses a TcpApp object to establish connections between nodes and to transfer data.

The parameters which changed from simulation to simulation for the HTTP web browsing experiments were the page sizes, the air-interface bandwidths, and the network scenario used for the simulation. Each network scenario had a different link delay setup, see Table 6.1. The page sizes ranged from 0.5Kb-70Kb. The experiment was devised so that the larger 10Kb-70Kb page represented full web pages that are normally pre-structured by popular mobile web browsing web servers, while the 0.5Kb-2Kb represented objects on a web page being downloaded. This will cover the two scenarios which a mobile web browser using a mobile device will encounter.

The exact page sizes simulated are 0.5Kb, 1Kb, 2Kb, 10Kb, 20Kb, 50Kb and 70Kb. The bandwidth ranged from 16Kbps-2048Kbps and can be seen in section 6.1.2. A simulation was done for each combination of these parameters and the results recorded.

6.6.3 Method

The method is identical to that detailed in Common Implementation Section 6.1.5 with the target application server being a web server.

Table 6.12 HTTP web browsing application parameters.

Parameter	Value
Size of web pages to be transferred	0.5Kb – 70Kb
Application protocol	HTTP
Air interface	WCDMA
Air interface bandwidth	16Kbps – 2048Kbps
HTTP Protocol	With and without caching

6.6.4 Results and Analysis

Case 1: The effects of changing end-to-end delay while keeping bandwidth and page (object) size constant and how this affects time and throughput performance. The page size is 70KB and the bandwidth is 384Kbps.

Table 6.13 The effects of changing end-to-end delay while keeping bandwidth and page size constant.

Delay (ms)	341*	341**	93
Throughput (KBp/s)	18.61	18.61	36.84
Time (s)	3.8	3.8	1.9

* Bottleneck at the air-interface. ** Bottleneck at the server on the wire network

A decrease in network delay increases throughput and decrease time taken to download a page. The short delay network took half the time to download than the two longer delay networks. Implying a shorter end-to-end delay increases web browsing performance. A shorter end-to-end delay results in a shorter packet propagation time and resend time; as explained in previous application simulations, see Sections 6.2-6.5 Case 1 for details. The HTTP protocol is sensitive to delay; the short delay network has a throughput of 36.8Kbps while the long delay networks have a throughput of 18Kbps. These results reinforces the use of HSDPA for web browsing which has a 2ms interval. A short delay is good for web browsing.

Case 2: The effects of changing bandwidth while keeping end-to-end delay and page size constant and how this affects time and throughput performance. Illustrated in Figure 6.17.

The higher the bandwidth, the shorter the time to download a webpage and the higher the throughput achieved for the download. This is because higher bandwidths have the potential for higher throughput, the same was found by researches [Chakravorty et al, 2004(A)]. HTTP is an application layer protocol which runs above a TCP transport layer protocol and the increase in performance can be explained by the same reasons given for other TCP applications as described in Section 6.2.1, Case 2.

For small objects of 2KB or less the bandwidth has a trivial affect on performance. If a large object were to be downloaded, like a pre-package web page, a higher bandwidth will perform better than a lower bandwidth, so it would be better to assign a higher bandwidth while transferring the page. [Timm-Giel, 2004] achieved rates of 118Kbps for an ideal rate of 128Kbps. Present research simulations for an ideal rate of 128Kbps was 113Kbps. Researchers found the main Yahoo page of size 60.3KB took 12 seconds to download [Chakravorty et al, 2004(A)]. In the present research the network had a shorter end-to-end delay, 70KB webpage download simulation took 9 seconds. Both simulation results were similar to that found in previous research.

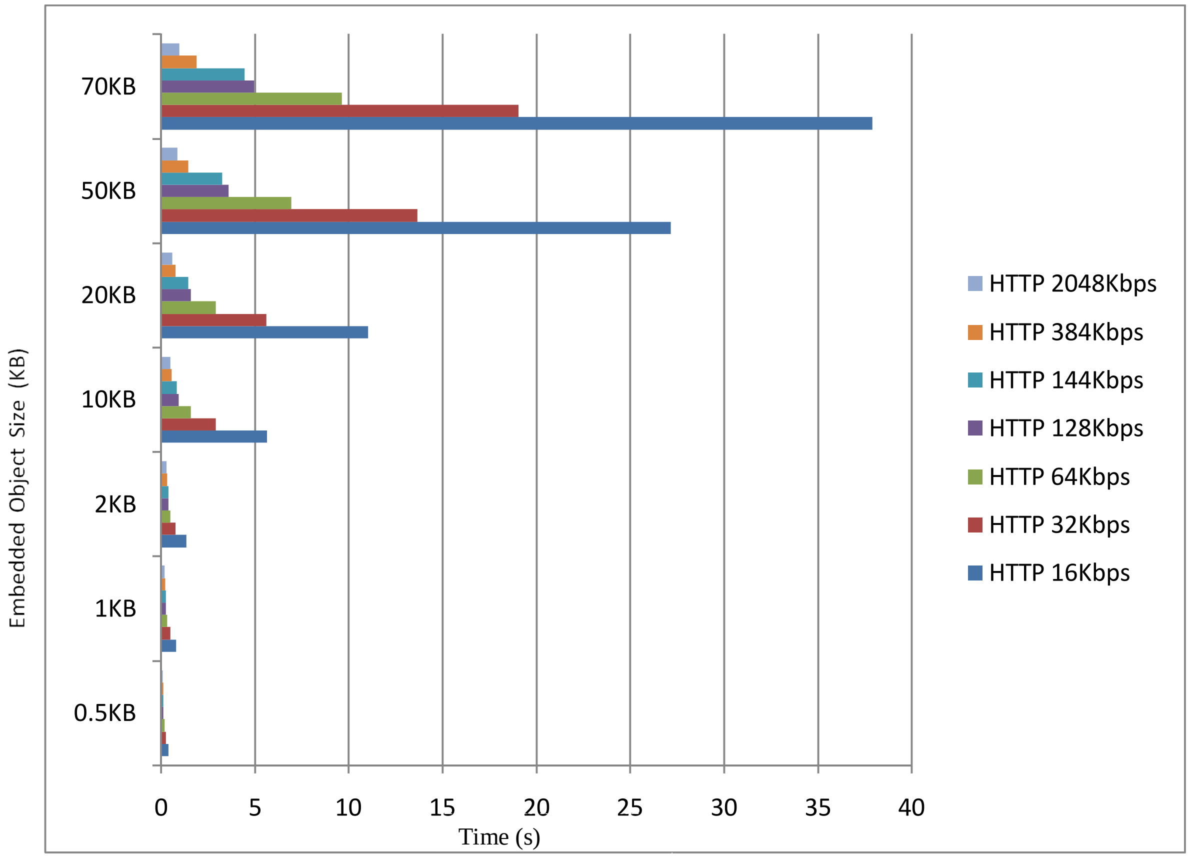

Figure 6.17 The time to download a web page by each bandwidth for each page size.

Case 3: The effects of changing page size while keeping end-to-end delay and bandwidth constant and how this affects time and throughput performance. This is illustrated in Figure 6.18.

As the webpage size to be downloaded becomes larger the throughput increases. There is a notable but small difference in throughput for small object sizes, see 2Kb and smaller objects in the graph, Figure 6.18. This is because the time to download a small object is short and not able to take advantage of the higher bandwidth.

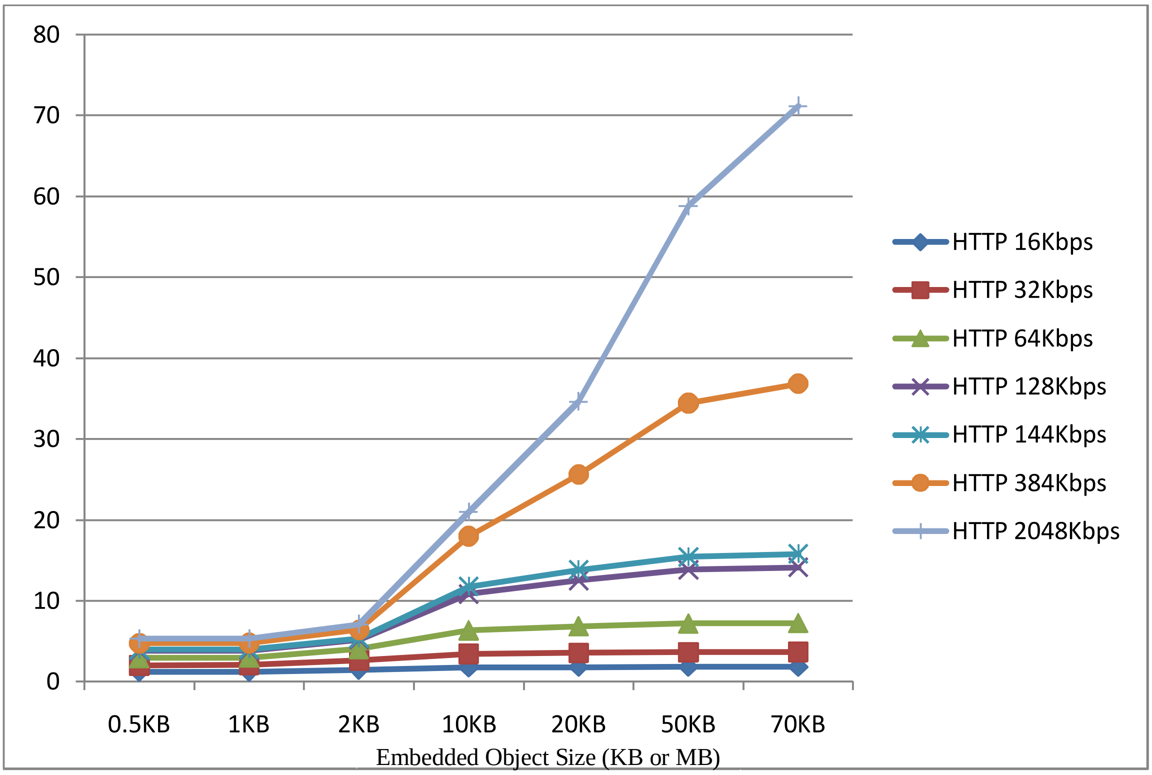

Figure 6.18 Page download throughput achieved by each bandwidth.

As the webpage size increases the throughput increases until the maximum throughput for the channel bandwidth is reached and the throughput levels out. At this stage further increases in webpage size will yield insignificant increases in throughput. This maximum throughput is reached for low bandwidths at smaller webpage (object) sizes than for higher bandwidths. This is controlled by the HTTP protocol and its underlying TCP protocol; this is explained in Section 6.2.4 Case 3. [Chakravorty et al, 2004(A)] also found that for small page sizes both high and low bandwidths performed similarly.

Extra case: comparing a pre-packaged webpage's performance to that of a webpage of the same size but which consists of many objects.

A 70KB page that consists of ten objects of size 0.5KB, five objects of size 1.0KB, five objects of size 2KB, three objects of size 10KB, and one object of size 20KB would have the following throughputs and time to download for each bandwidth.

Table 6.14 The effects of a page consisting of many objects. The combined object size is 70KB and the end-to-end delay is 46ms.

Bandwidth	**Total time (s)**	**Throughput KB/s**	**Time pre-package (s)**	**Throughput pre-package**	**Pre-package % of non-packaged**
16Kbps	42.74	1.64	37.9	1.85	88.7
32Kbps	22.98	3.05	19.0	3.68	82.7
64Kbps	13.46	5.20	9.6	7.27	71.3
128Kbps	8.89	7.87	5.0	14.14	56.2
144Kbps	8.73	8.02	4.4	15.79	50.4
384Kbps	6.10	11.48	1.9	36.84	31.1
2048Kbps	5.30	13.21	1.0	71.14	18.9

When a page containing many objects is downloaded the time to download the page takes longer and the throughput decreases. The actual impact is not as great as expected. The impact is almost constant with a web page taking 4-5 seconds longer to download. This is in scenario 3 which has a very short delay. The time taken to download a 70KB file on this short delay network for 144Kbps bandwidth was 9 seconds (to nearest second) compared well to the 60.3Kb page in 12 seconds experienced by [Chakravorty et al, 2004(A)] which has a longer end-to-end delay. The extra end-to-end delay will be added to each object being downloaded, which will cause an increase in time to download.

Because the difference in time between multiple object webpage and pre-packaged webpage is almost the same in seconds for each bandwidth, it means that a higher bandwidth will be impacted on more than a low bandwidth. A 2048Kbps bandwidth takes 1 seconds pre-packaged and 5 seconds multi-object, while 16Kbps bandwidth takes 38 seconds pre-packaged and 43 seconds multi-object webpage download. The effects are more pronounced on the higher bandwidth.

The effects of updating expired cached objects makes a webpage take longer to download but this is only a fraction of a second longer for each object, and does not have a large effect of the performance. Caching occurs on the wired network and adds a small extra time when a page needs to be fetched from another server. If the server hosting the master copy is very busy causing it to be slow then caching will be a factor that slows down the webpage downloading process, but most likely this will not happen.

6.6.5 Conclusion

How network factors affected the performance of HTTP web browsing was investigated. Simulations were designed which showed how network factors: end-to-end delay, bandwidth and page size affected HTTP web browsing performance. It was found that a short end-to-end delay always improved the service and a high bandwidth too always improved the service. The size of the page in combination with bandwidth affected the performance. The effects of caching will increase performance, while a page consisting of many small objects will decrease performance. This indicates that the best case for the HTTP web browsing service will be low end-to-end network delay and high bandwidth, with a cached web page that consists of a few objects.

The HTTP protocol runs above the TCP protocol and TCP congestion control will affect HTTP in the same manner it does to FTP over TCP. A higher bandwidth is capable of a higher maximum data rate. To improve web browsing to allow for higher throughput a higher bandwidth can be used. The return on throughput for an increased bandwidth is logarithmic and it would be better for a network to assign a medium level bandwidth to each user, in such maximising a cell's throughput and performance.

The effects of the page size on web browsing show that when a small web page is downloaded all bandwidths perform similarly. This is because TCP is unable to make use of the higher bandwidths as the increase in window size occurs to slowly. This indicates that using high bandwidths for small web pages are wasteful. While in large web pages the high bandwidth performed much better than the low bandwidth as TCP was able to increase each window size to its optimal value. The higher bandwidth will have a larger window size and be able to transfer data at a higher rate.

The higher end-to-end delay decreases performance, by adding a few hundreds of a second to the transmission time for each packet transferred. This is offset some bit by the higher end-to-end network delay having a larger potential window size. But the simulations show that a lower end-to-end always outperforms a higher end-to-end, all else equal.

A page consisting of many small objects will take longer to download than an equivalent size page with a few large objects. Objects that are not cached and need to be downloaded from the server of origin also take longer. Both these factors have the same effect as a longer end-to-end delay and decreases performance. Simulations showed that these factors tend to add an almost fixed overhead regardless of the bandwidth used. The fixed overhead as a percentage for a high bandwidth is larger than for a low bandwidth. This is the reason that high bandwidths are affected more by many small objects and cached objects than low bandwidths.

The factor which affects results the most would be the end-to-end delay. The short delay network performed twice as well as the two long delay networks. Based on the simulation results and the number of simultaneous users for a given bandwidth (Table 6.5), it is suggested that the 384Kbps channel be used as this will maximise the networks capacity.

6.7 Media Broadcasting Experiments

6.7.1 Network Setup

To simulate a media broadcasting application a 3G UMTS network that contained a broadcasting server within the IMS subsystem was setup. This network was built from NS2 components and had the same specification attributes of a typical 3G UMTS network. The GGSN linked to the rest of the UMTS core network containing the IMS subsystem, which had two connected elements enabling media broadcasting. Other details are the same as depicted in Section 6.1.2 Common Implementation Details. See Figure 6.19.

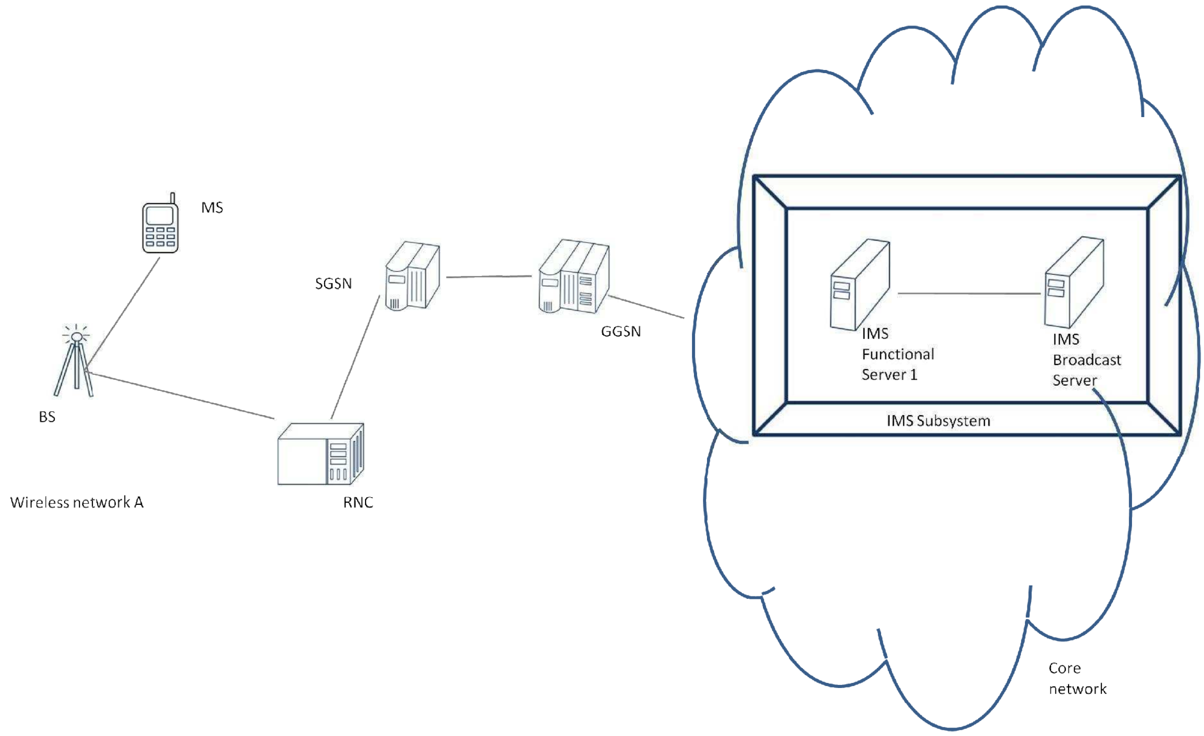

Figure 6.19 The network used for media broadcast simulations.

6.7.2 Parameters and Protocols

The IMS Broadcasting server streams media data to recipients. This was simulated by a NS2 traffic generator over a UDP connection. The Common Implementation Details 6.1.3 describes the details of the setup.

6.7.3 Method

The method is identical to that detailed in Common Implementation Section 6.1.5 with the target application server being the media broadcasting component of the IMS within the core network.

6.7.4 Results and Analysis

Case 1: The effects of changing end-to-end delay while keeping bandwidth, frame size and frame rate constant. The bandwidth is 384Kbps, Frame rate 10 frames a seconds and the Frame size is 3415 bytes, which works out to a bit rate of 267Kbps. The UDP packet size is 1024 bytes. Frame size is equivalent to a medium frame size on a medium mobile display.

Table 6.15 Effects of changing end-to-end delay while keeping bandwidth, frame rate and frame size constant.

Delay (ms)	170*	170**	46
Throughput (KBp/s)	57353.9	57353.9	57454.0
Video Time (s)	100.0	100.0	100.0
% of delayed frames	~0.0	~0.0	~0.0

* Bottleneck at the air-interface. ** Bottleneck at the server on the wire network

The end-to-end delay has no affect on the broadcasting of media data, which makes sense since packets are streamed at a fixed rate with no need for feed back from the receiving node. This means that the service provider does not need to guarantee a minimum network delay to enable the broadcasting service. The advantage of broadcast is that all users will share the same bandwidth channel and there can be an increase in users without an increase in resources, making broadcasting a highly scalable application.

Case 2: The effects of changing bandwidth while keeping end-to-end delay, frame size and frame rate constant. End to end delay is 46ms, frame rate 15 frame a second and the length of video clip 67 seconds, while the UDP packet size 1024 bytes. This is illustrated in Figure 6.20.

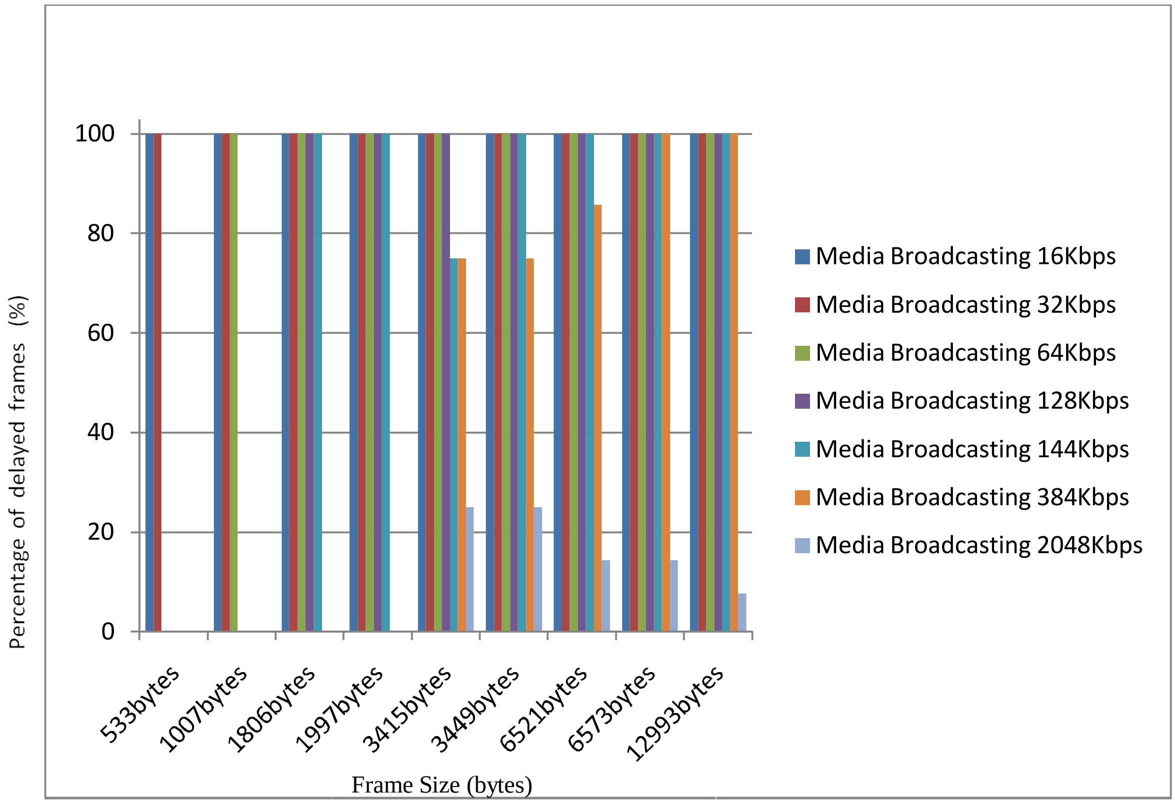

Figure 6.20 The percentage of delayed frames by each bandwidth for each frame size at 15 frames a second.

Increasing bandwidth increases performance. A high bandwidth has a higher throughput and a lower percentage of delayed frames. As the bandwidth at the air-interface is able transfer more data per second, packets will spend less time in the BS queue so as to allow more packets to arrive in time. This is the cause of the lower percentage of delayed frames as bandwidth increases. To allow a user to receive a quality of broadcast good enough to watch without jitter the bandwidth will need to be increased until the percentage of delayed frames are low enough. Once a bandwidth researches ~0% delayed frames further increases in bandwidth is wasteful as the quality of viewing the video changes insignificantly.

[Brasche and Walke, 1997] Ascertain that for higher streaming rates higher bandwidths are required, the bandwidth must be larger than the required streaming rate. Brasche and Walke argue that video over 3G UMTS will be significantly better than GPRS. The data rates of 105Kbps-200Kbps recommended for mobile video can only be attained by 3G channels [Snap9 Corporation, 2006] and [Hewlett-Packard Development Company, 2007]. All users of media broadcasting service share the same bandwidth channel and increasing the bandwidth will improve all user's performance. The minimum bandwidth that can deliver the needed throughput at ~0% frame delay should be used.

Case 3: The effects of changing frame rate while keeping end-to-end delay, frame size and bandwidth constant and how time and throughput performance is affected. End-to-end delay is 46ms, bandwidth is 128Kbps, and frame size is 1806 bytes, while UDP packet size is 1024 bytes.

Table 6.16 Effects of changing frame rate while keeping bandwidth, end-to-end delay and frame size constant.

Frame rate (f/s)	**% of delayed frames**	**Throughput bytes /s**
5	0.1	18120.5
10	50.0	18120.5
15	100.0	18120.5

An increase in frame rate results in an increase in percentage of delayed frames, while throughput remains unchanged. The increase in frame rate decreases the performance. There is a dramatic decrease in performance as frame rate increases, 5 frames a second performed with almost no delays and at 15 frames a second all frames are delayed. The cause is:

(Equation 6.2) Frame size * Frames a second = bit streaming rate.

Which means a small change in frame rate can cause a large change in required bit rate. A drama may require a playback rate of 5 frames, this could then be streamed almost jitter free to a user, as seen in case 3/Table 6.16. While an action movie may need to be viewed at 15 frames a second, if the same bandwidth as the drama were to be used, the user would suffer from 100% delayed frames. The action movie will not be able to be viewed at an acceptable quality.

[Snap9 Corporation, 2006] an increase in required frame rate results in a higher bandwidth needed. In real networks drama and news broadcast could be more accepted by users as it would be of a higher quality than action broadcasts. This means lower bandwidths can be used for news and drama style broadcasts while a higher bandwidth would be required for action broadcasts. Research has been done to attempt to use varying channel rates based on the sequence of video to follow, to determine the required bandwidth. If this is done the channel rate can be kept to the minimum required bandwidth and free up more network bandwidth resources. But that is out the scope of present research.

Case 4: The effects of changing the frame size while keeping end-to-end delay, bandwidth and frame rate constant. End-to-end delay is 46ms, frame rate 15 frame a second and the length of video clip 67 seconds, while the UDP packet size 1024 bytes. This is illustrated in Figure 6.21.

An increase in frame size results in an increase in percentage of delayed frames, while throughput remains unchanged. The increase in frame size decreases the performance. This is because larger frames increase the bit streaming rate.

(Equation 6.2) Frame size * Frames a second = bit streaming rate.

The increase in frame size is multiplied by the number of frames a second and a small increase in frame size can cause a large increase in bit streaming rate.

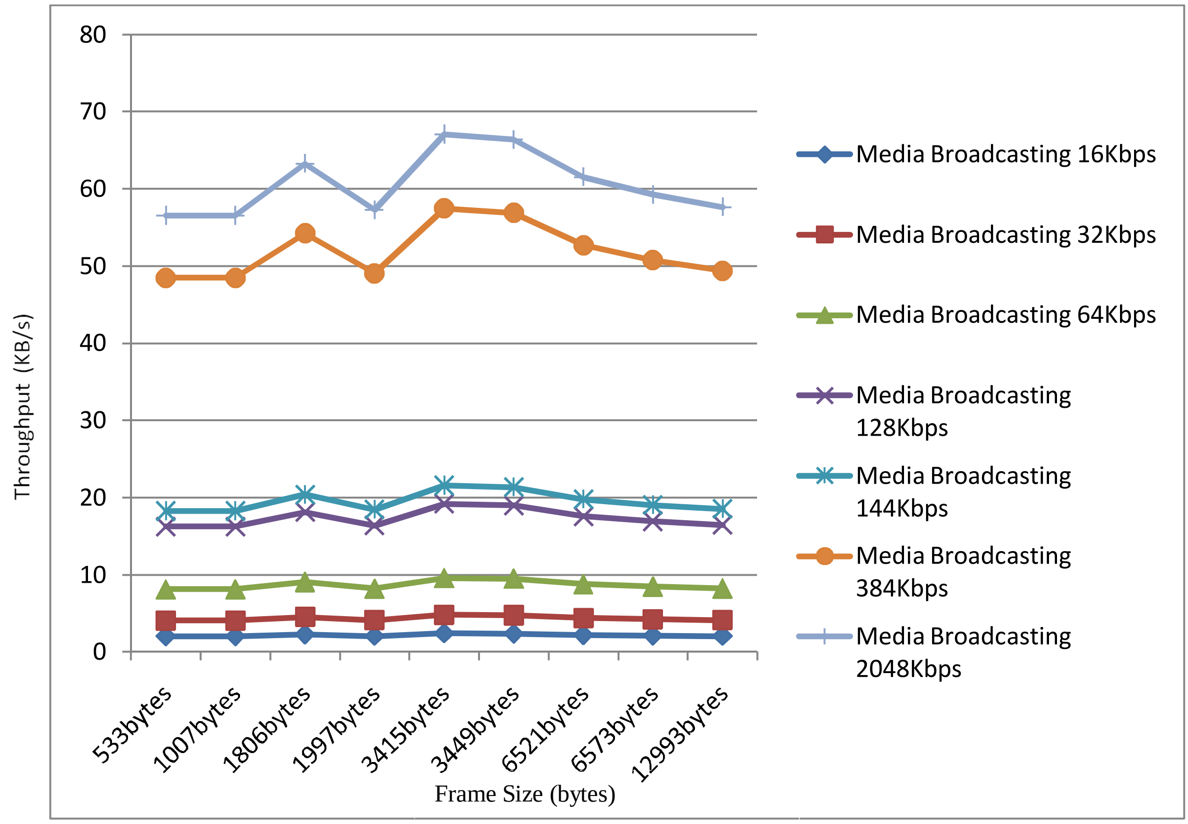

Figure 6.21 Media broadcast throughput achieved by each bandwidth.

A bandwidth channel is capable of a maximum throughput. If the broadcast rate is lower than this maximum then the user will receive data at the broadcasting rate and be able to playback media at the target playback rate [Weber, 2006]. Close to zero packets will be delayed and very little jitter will be experienced. If the broadcast rate is higher than this maximum then the user will receive data at the maximum channel rate. Packets will arrive late and the user will experience jitter. The greater the margin between the broadcast rate and the channel's maximum rate the worst the jitter will be.

6.7.5 Conclusion

How network factors affected the performance of media broadcast was investigated. Simulations were designed which showed how network factors: end-to-end delay, bandwidth, and frame rate and frame size affected media broadcast performance. The bandwidth is important and the throughput of the assigned bandwidth needs to be higher than the required bit rate. The frame size and frame rate affect the required bit rate and in turn affect the bandwidth needed for successful media broadcast. The end-to-end delay does not affect the media broadcast service. A media broadcast service will perform best with a high bandwidth.

Bandwidth has drastic affects on the performance of media broadcast. An increase in bandwidth will lead to an increase in throughput and a drop in the percentage of delayed frames. Once the percentage of delayed frames reaches ~0% then a further increase in bandwidth will not yield performance increases. It is wasteful to increase bandwidth once ~0% delayed frames has been achieved. The obtained throughput will be dependent on assigned bandwidth and broadcast rate (required bit rate).

Media is broadcasted at either the media broadcast rate if this is less than the assigned channel rate or at the assigned channel rate if the broadcast rate is greater than the assigned channel rate. The greater the difference between the broadcast rate and assigned channel rate the greater the number of delayed frames, when the broadcast rate is larger than assigned channel rate. This causes greater jitter in the video received by the viewer.

The required bit rate, which is affected by both the frame rate and the frame size, is giving by the following formula:

(Equation 6.2) Frame size * Frames a second = bit streaming rate.

It can be seen from this equation that an increase in either the frame rate or the frame size will lead to an increase in the number of delayed frames. This was shown in simulations. A small increase in frame rate or frame size can cause a large increase in required bit rate, for this reason changing the frame rate or frame size can take a broadcast from ~0% delayed frames to 100% delayed frames.

The different genres of video need different frame rates to be acceptably viewed by viewers. The frame rate required by news style video is low because scenes don't change often, while action video requires a high frame rate to be viewed acceptably.

The end-to-end delay does not have an effect on throughput or percentage of delayed frames. This is because media is broadcasted at a constant rate without feed back being required from the receiver. Each frame is received relative to the frame before it; this is the cause of end-to-end delay having no affect on the broadcast service.

The bandwidth is the most important factor of the media broadcast service and the bandwidth used is dependent on the required bit rate. There is no maximum number of users of the service; all users tuned in to the broadcast will receive the broadcast. This happens regardless of the bandwidth assigned for the broadcast. It is recommended that the lowest bandwidth which has ~0% delayed frames be used.

6.8 Video Calling Experiments

6.8.1 Network Setup

To simulate a video call application a UMTS network was setup which connected to the 3G core network that contained the video call server within the IMS subsystem. The GGSN linked to the rest of the UMTS core network that contained the IMS subsystem, which had two connected elements that enabled video call. The rest of the network setup details are the same as described in Common Implementation Section 6.1.2, see Figure 6.22.

6.8.2 Parameters and Protocols

The IMS video call server streams media data between recipients. This was simulated by a NS2 traffic generator over the UDP protocol. Details of the setup can be found in the Common Implementation Details 6.1.3. The simulation used FFD channels so that the upload and download bandwidths did not affect each other. The frame rates used in the simulation is 5 frames a second, 10 frames a second and 15 frames a second.

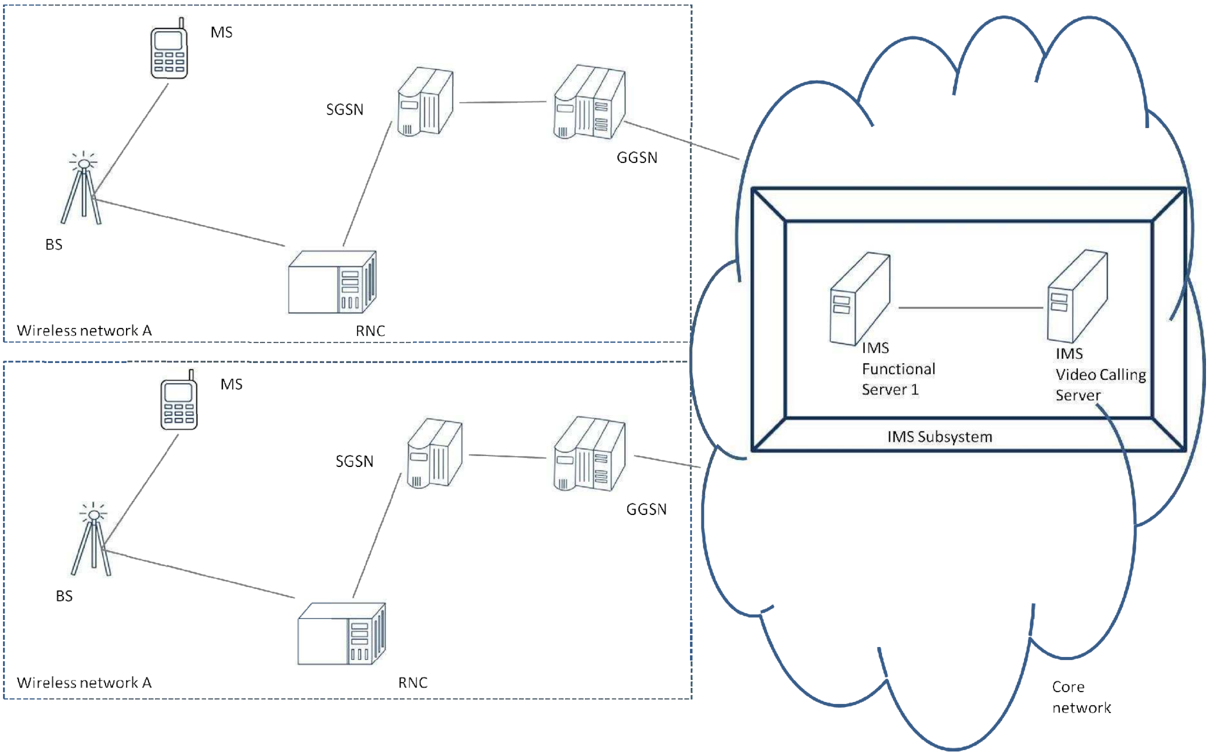

Figure 6.22 The network used for video call simulations.

6.8.3 Method

The method is identical to that detailed in Common Implementation Section 6.1.5 with the target application server being the video call component of the IMS within the core network.

6.8.4 Results and Analysis

Case 1: The effects of changing end-to-end delay while keeping bandwidth, frame size and frame rate constant. The bandwidth is 384Kbps, Frame rate 10 frames a seconds and the Frame size is 1007 bytes, which works out to a bit rate of 79Kbps. The UDP packet size is 1024 bytes. Frame size is equivalent to a small frame size on a medium mobile display. IMS processing is taken at 10ms so as not to impact greatly on the added network delay.

Table 6.17 Effects of changing end-to-end delay while keeping bandwidth, frame rate and frame size constant.

Throughput Up(bytes)	Throughput down (bytes)	% of delay frames up	% of delay frame down	% of total delay frames	Delay upload (ms)	Delay download (ms)	Total Delay (ms)
48189.1	48189.1	0.1	0.1	0.2	*170.5	*170.5	351
48189.1	48189.1	0.1	0.1	0.2	**170.5	**170.5	351
48473.1	48473.1	0.1	0.1	0.2	45.5	45.5	101

* Bottleneck at the air-interface. ** Bottleneck at the server on the wire network

The end-to-end delay does not affect the jitter or the throughput of the video call, but it does show that scenario 1 and 2 which have a high end-to-end delay suffers from a slight propagation lag between user replies. Conversational propagation delay is normally targeted at 200ms or less [Cisco Systems Inc, no date]. The short delay network is lower than this with the longer delay network above this. Even if the end-to-end were greater than scenario 1 the end-to-end does not pose a major problem and will not affect performance severely.

Data is streamed at a constant rate without the need of feedback from the Receiver so end-to-end delay does not affect throughput. A service provider does not need to guarantee a minimum network delay to enable the video call service.

Case 2: The effects of changing bandwidth while keeping end-to-end delay, frame size and frame rate constant. Frame rate 15 frames a second and the length of video clip 67 seconds, while the UDP packet size is 1024 bytes. IMS processing is taken at 10ms so as not to impact greatly on the added network delay. Delay up and delay down is 45.5ms. The experiment uses symmetrical bandwidths for uploading and downloading. Therefore the total end-to-end delay in milliseconds would be 101ms. This is illustrated in Figure 6.23.

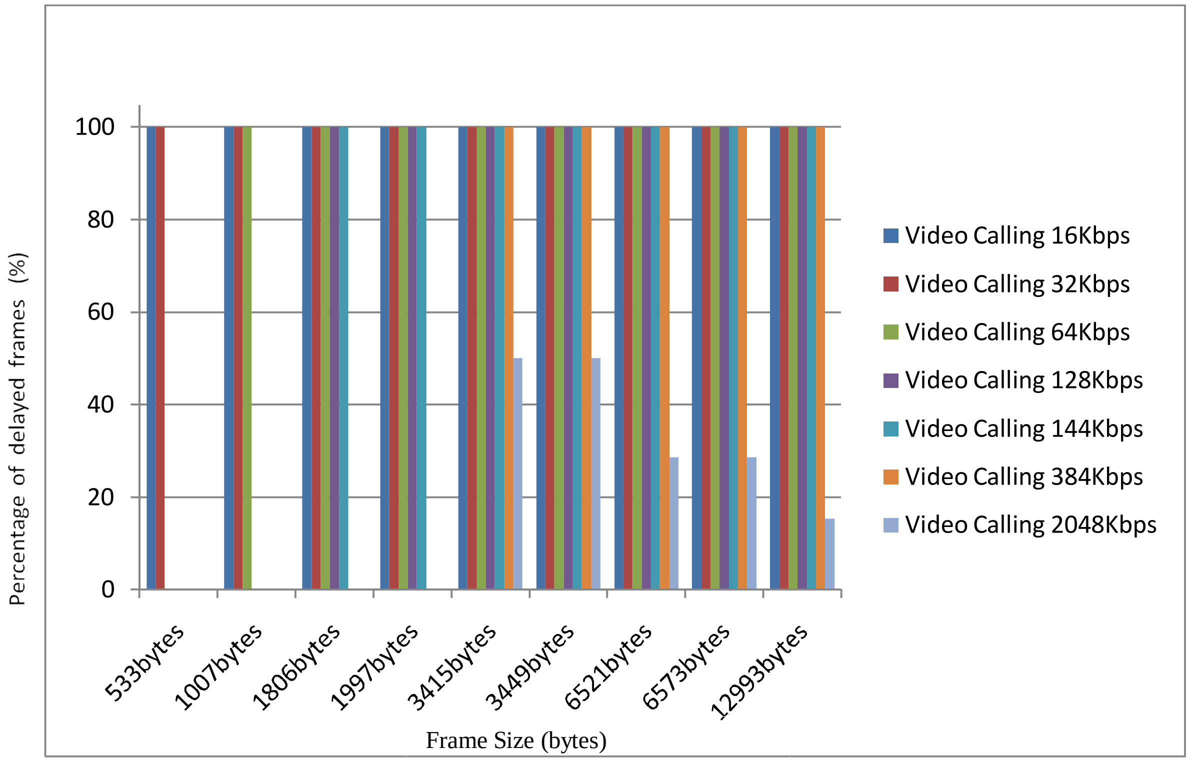

Figure 6.23 The percentage of delayed frames by each bandwidth for each frame size at 15 frames a second.

An increase in bandwidth means less delayed frames and an increase in throughput. Performance is sensitive to bandwidth and raising the bandwidth channel can allow delayed frames to go from 100% delayed to ~0% delayed, see frame size 533 at bandwidth 32Kbps has 100% while 64Kbps has ~0% delayed. The high bandwidth on the wired network allows traffic to flow freely until it reaches the BS. Once at the BS the air-interface bandwidth throttles throughput if streaming rate is greater than air-interface channel bandwidth.

Video frames begin to queue at the BS while waiting to be sent across the air-interface. A high air-interface bandwidth allows traffic to flow more freely, with smaller queues, resulting is less delayed frames. Once a bandwidth reaches ~0% delayed frames further increases in bandwidth is wasteful as the quality of viewing the video will change insignificantly.

Video call service is resource intensive as both uploading and downloading requires bandwidth. Frames can be delayed (or lost) on the upload and on the download. No buffering takes place so if a frame is delayed on the upload it will be delayed on the download. Frames that were sent on time on the upload can become delayed on the download, increasing the number of delayed frames received by the receiver. Also the downloading rate is limited by the uploading rate, as the downloading can not throughput traffic quicker than the uploading is sending it. It is most efficient to have the same upload and download bandwidths so that neither limits the other.

A high bandwidth will perform better, but this will cause the application to be resource intensive. The network will only be able to support a few video calls concurrently. Once the call is completed resources are released.

Case 3: The effects of changing frame rate while keeping end-to-end delay, frame size and bandwidth constant and how time and throughput performance is affected. End-to-end delay is 46ms, bandwidth is 64Kbps, and frame size is 1007 bytes, while UDP packet size is 1024 bytes. IMS processing is taken at 10ms so as not to impact end-to-end delay greatly.

Table 6.18 Effects of changing frame rate while keeping bandwidth, end-to-end delay and frame size constant.

Frames per second	Throughput Up(bytes)	Throughput down (bytes)	% of delay frames up	% of delay frame down	% of total delay frames
5	8125.6	8125.6	0.1	0.1	0.2
10	8125.6	8125.6	100.0	100.0	100.0
15	8125.6	8125.6	100.0	100.0	100.0

An increase in frame rate results in an increase in percentage of delayed frames, while throughput remains unchanged. The increase in frame rate decreases performance. There is a dramatic decrease in performance as frame rate increases, 5 frames a second performed with almost no delays and at 10 and 15 frames a second all frames were delayed. The cause is:

(Equation 6.2) Frame size * Frames a second = bit streaming rate.

This means a small change in frame rate can cause a large change in required bit rate. [Snap9 Corporation, 2006] an increase in required rate results in a higher bandwidth needed.

Low frame rate will have less delayed frames and so have less jitter and better performance. Video call has a news style image, containing mostly head shots with little change in scenery between frames. This is beneficial as it can perform well at low frame rates. A low frame rate in turn will require a lower bandwidth. The network could then be able to deliver a good quality video call at a low bandwidth enabling higher numbers of concurrent users. The assigned bandwidth will need to be higher than the required bit rate, which means the lowest bandwidth greater than the required bit rate a second would be needed, this would normally be the 64kbps or 128kbps bandwidth, resulting in 7.5 and 4.9 concurrent users, based on Table 6.5.

Case 4: The effects of changing the frame size while keeping end-to-end delay, bandwidth and frame rate constant. End-to-end delay is 46ms, frame rate 15 frame a second and the length of video clip 67 seconds, while the UDP packet size 1024 bytes. This is illustrated in Figure 6.24. IMS processing is taken at 10ms as not to impact the end-to-end delay greatly.

An increase in frame size results in an increase in percentage of delayed frames, while throughput remains unchanged. The increase in frame size decreases the performance. This is because larger frames increase the bit streaming rate. See Equation 6.2. The increase in frame size is multiplied by the number of frames a second and a small increase in frame size can cause a large increase in bit streaming rate.

A bandwidth channel is capable of a maximum throughput. If the video streaming rate is lower than this maximum then the user will receive data at the video streaming rate and be able to playback media at the target playback rate. Close to zero packets will be delayed and very little jitter will be experienced. If the video streaming rate is higher than the channel rate

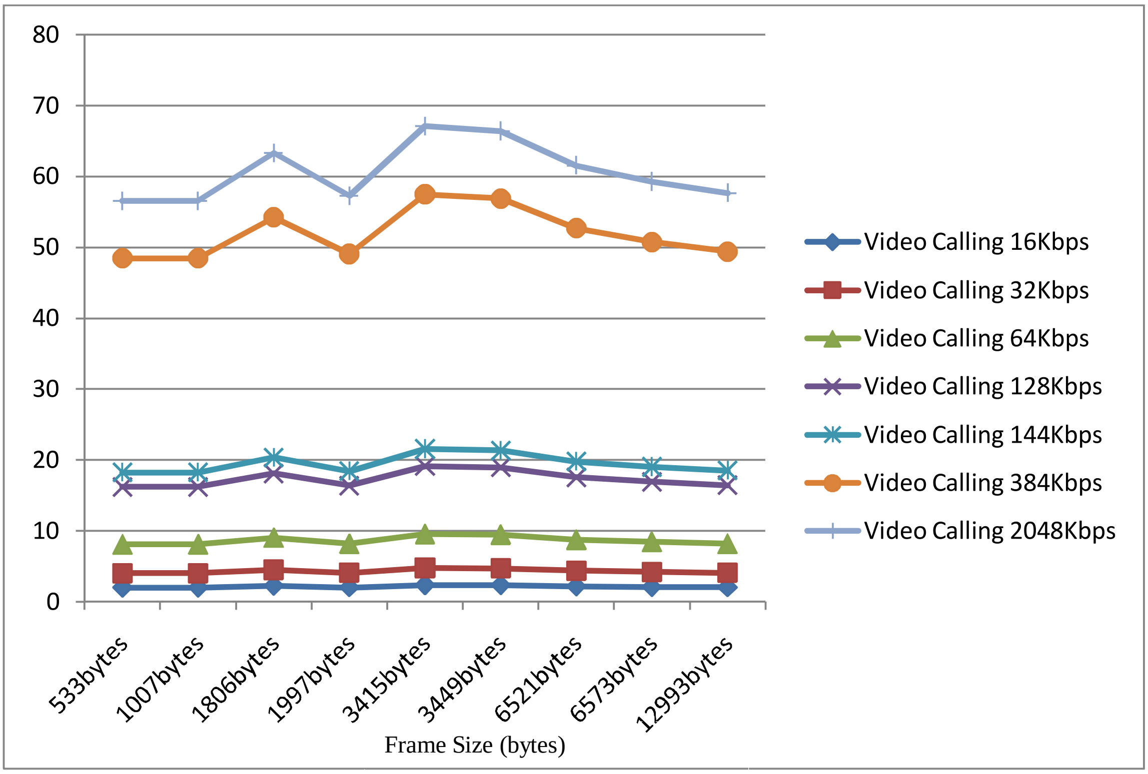

Figure 6.24 Video call throughput achieved by each bandwidth.

then the user receives data at the channel rate. Packets will arrive late and the user experiences jitter. The greater the margin between the video streaming rate and the channel's maximum rate the worst the jitter experienced.

6.8.5 Conclusion

The effect of network factors on the performance of the video call service was investigated. Simulations were designed which showed how network factors: end-to-end delay, bandwidth, and frame rate and frame size affect video call performance. The bandwidth is important and the throughput of the assigned bandwidth needs to be higher than the required bit rate. The frame size and frame rate affect the required bit rate and in turn affect the bandwidth needed for successful video call. The end-to-end delay does not affect the video call service. The video call service performs best with a high bandwidth.

Bandwidth has drastic affects on the performance of the video call service. An increase in bandwidth leads to an increase in throughput and a drop in the percentage of delayed frames. Once the percentage of delayed frames reaches ~0% then a further increase in bandwidth does not yield an increase in performance. It is wasteful to increase bandwidth once ~0% delayed frames has been achieved. The obtained throughput is dependent on assigned bandwidth and broadcast rate (required bit rate).

The video call is streamed at either the video call rate if this is less than the assigned channel rate or at the assigned channel rate if the video call streaming rate is greater than the assigned channel rate. The greater the difference between the video call streaming rate and assigned channel rate, the greater the number of delayed frames, when the broadcast rate is larger than assigned channel rate. This causes greater jitter in the video received by the viewer. In a service like video call delayed frames are dropped, and current frames are sent through to help the flow of the conversation, this will come through as jitter in the video call conversation.

The required bit rate, which is affected by both the frame rate and the frame size, is giving by the following formula:

(Equation 6.2) Frame size * Frames a second = bit streaming rate.

It can be seen from this equation that an increase in either the frame rate or the frame size will lead to an increase in the number of delayed frames. This was shown in simulations. A small increase in frame rate or frame size can cause a large increase in required bit rate, for this reason changing the frame rate or frame size can take a video call from ~0% delayed frames to 100% delayed frames.

Video call has mostly news styled image and is capable of decent performance with low frame rates and small frame sizes as the scene does not change often. This means a video call can be made with a relatively low bandwidth and be of an acceptable quality.

The end-to-end delay does not have an affect on throughput or percentage of delayed frames. But the end-to-end delay does affect conversation propagation and a long end-to-end delay can cause a lag between statement and response while conducting a conversation. A smooth conversation requires a round trip delay of 200ms. A longer round trip delay is not a severe problem as the conversation can still take place, but with a slight lag, as is experienced with international phone calls.

The bandwidth is the most important factor of the video calling service and the bandwidth needed is dependent on the required bit rate. A video call would use a low frame rate with either a small or medium size frame, which will result in a required bit rate of 21-71kbps based on Table 6.2. Using the lowest bandwidth that has ~0% delayed frames. The experimental simulations show that the 64Kbps bandwidth be used for a small frame size, while the 128Kbps bandwidth should be used for the medium size frame resolution.

6.9 Media Streaming Experiments

6.9.1 Network Setup

A media streaming application was simulated by setting up a UMTS network connected to the wired Internet containing a media streaming server. This network was built from NS2 components and has the same attributes of a typical 3G UMTS network. The GGSN linked to the Internet containing the media streaming server. The rest of the details are the same as described in the Common Implementation Section 6.1.2, see Figure 6.25

6.9.2 Parameters and Protocols

The media streaming server streams media data to recipients at a rate specified by the application controlling the server. The rate could be set higher than that of playback so as to transmit all media as quickly as possible buffering it at the recipient. Media was chosen to be streamed at the playback rate. Media streaming is simulated by a NS2 traffic generator over the UDP protocol.

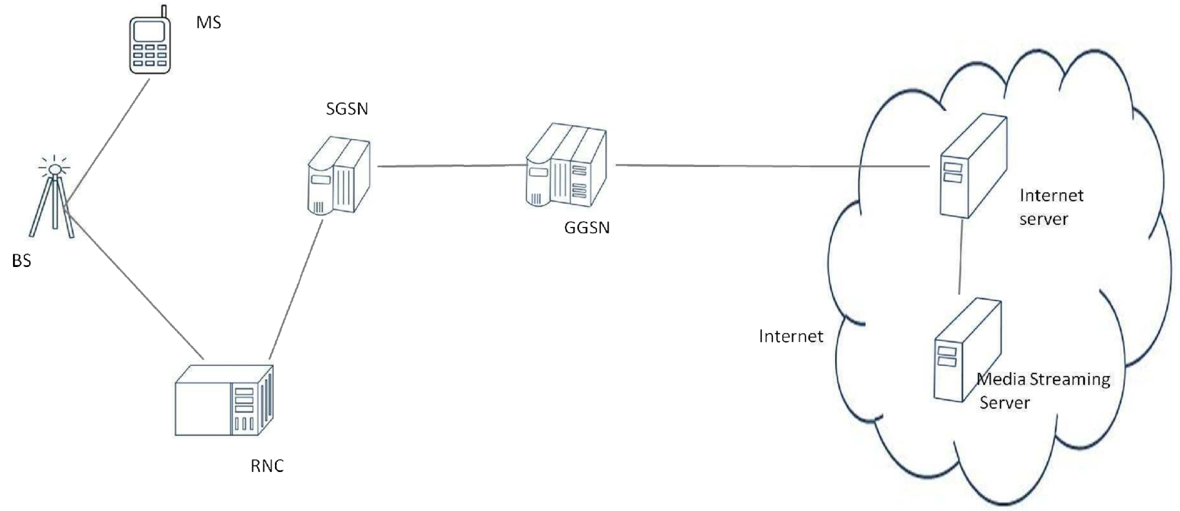

Figure 6.25 The network used for media streaming simulations.

6.9.3 Method

The method is identical to that detailed in Common Implementation Section 6.1.5 with the target application server being a media streaming server.

6.9.4 Results and Analysis

Case 1: The effects of changing end-to-end delay while keeping bandwidth, frame size and frame rate constant. The bandwidth is 2048Kbps, Frame rate 15 frames a seconds and the Frame size is 12933 bytes, which works out to a bit rate of 1523Kbps. The UDP packet size is 1024 bytes. Frame size is equivalent to a large frame size on a large mobile display.

Table 6.19 Effects of changing end-to-end delay while keeping bandwidth, frame rate and frame size constant.

Delay (ms)	170*	170**	46
Throughput (KBp/s)	57619	57619	57650
Video Time (s)	66.7	66.7	66.7
Buffer Time (s)	162.4	162.4	162.4

* Bottleneck at the air-interface. ** Bottleneck at the server on the wire network

The end-to-end delay has no affect on the streaming of data, which makes sense since packets are streamed at a fixed rate with no need for feed back from the receiving node. The throughput stays the same and the required buffer size stays the same. This means that the service provider does not need to guarantee a minimum network delay to enable the media streaming service.

Case 2: The effects of changing bandwidth while keeping end-to-end delay, frame size and frame rate constant. End-to-end delay is 46ms, frame rate 15 frame a second and the length of video clip 67 seconds, while the UDP packet size 1024 bytes. This is illustrated in Figure 6.26.

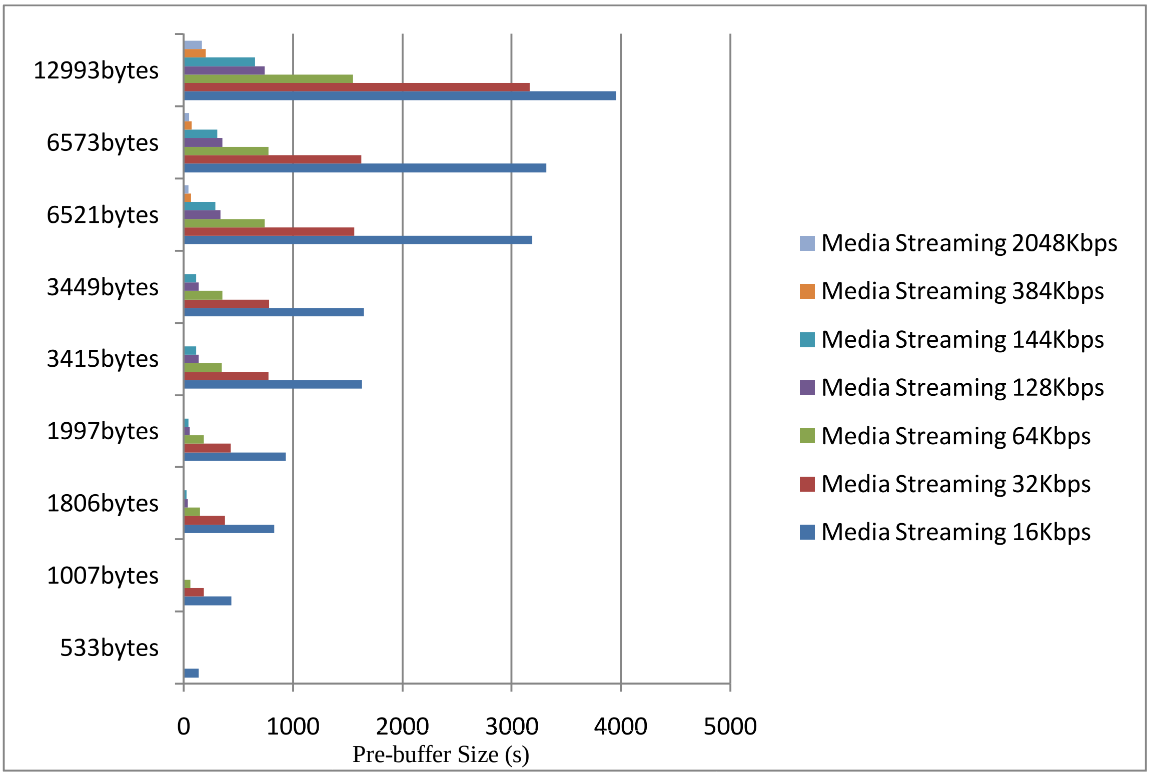

Figure 6.26 Required buffers in seconds required by each bandwidth for each frame size at 15 frames a second.

Increasing bandwidth increases performance. A high bandwidth has a higher throughput and a smaller required buffer size. To allow a user to playback video smoothly, a buffer needs to be put in place to mitigate jitter caused by delayed frames. Buffer time increases exponentially for a decrease in bandwidth. Exponential increase in buffer size was also found by [Chesterfield et al, 2004] and [Weber, 2006]. A high bandwidth is capable of supporting a higher required throughput. Below is an example which demonstrates this exponential buffer increase.

Example: Media is streamed across the wired network to the BS where it is then transmitted across the wireless channel. In situation one, media was streamed at 10 packets a second on the wired network and then transmitted at 9 packets a seconds over the wireless channel for period of 10 seconds. Then the buffer required would be 1 second to ensure smooth playback. In the next situation the wired streaming rate is doubled to 20 packets a second and the results investigated.

Situation two, media is streamed at 20 packets a second on the wired network and then transmitted at 9 packets a second over the wireless channel for a period of 10 seconds. The required buffer size for smooth playback would then be 11 seconds. This indicates that increasing the required streaming rate be a factor of 2 has increased the required buffer size by a factor of 11.

Once a bandwidth requires a close to zero buffer size further increases in bandwidth is not needed as the quality of viewing the video changes insignificantly. In present research simulations use a streaming rate equivalent to the playback rate, which means an increase in bandwidth will not cause an increase in throughput once the bandwidth is greater than the required streaming rate. The 384Kbps to 2048Kbps did not increase performance for the end user. But a 144Kbps to 384Kbps bandwidth change improved the service dramatically. See Figure 6.26.

[Brasche and Walke, 1997] Ascertain that for higher streaming rates higher bandwidths are required, the bandwidth must be higher than the required streaming rate. Brasche and Walke argue that video over 3G UMTS will be significantly better than GPRS because of 3G's higher available bandwidth. Decent quality video would require a bit rate in the region of

100-200Kbps which means a 128kbps or greater bandwidth channel would be needed resulting in 4.9 or less concurrent users enjoying the service concurrently, based on Table 6.5.

Case 3: The effects of changing frame rate while keeping end-to-end delay, frame size and bandwidth constant and how time and throughput performance is affected. End-to-end delay is 46ms, bandwidth is 128Kbps, and frame size is 3415 bytes, while UDP packet size is 1024 bytes. IMS processing is taken at 10ms as not to impact the end to end delay greatly.

Table 6.20 Effects of changing frame rate while keeping bandwidth, end-to-end delay and frame size constant.

Frame rate (f/s)	Buffer time (s)	Throughput bytes /s	Video length (s)	Buffer time % of video length
5	0.0	19177.5	200	0.0
10	93.6	19177.5	100	93.6
15	133.6	19177.5	66.7	200.3

Throughput is unaffected as frame rate increases, but the required buffer size increases exponentially. This is because the frame rate increases the required bit rate, which in turn increases the size of the initial required buffer, explained in the example in Case 2. [Weber, 2006] and [Chesterfield et al, 2004] also found that increases in required bit rate exponentially increase the required buffer.

Case 4: The effects of changing the frame size while keeping end-to-end delay, bandwidth and frame rate constant. End-to-end delay is 46ms, frame rate 15 frame a second and the length of video clip 67 seconds, while the UDP packet size 1024 bytes. This is illustrated in Figure 6.27. IMS processing is taken at 10ms as not to impact the end to end delay greatly.

An increase in frame size results in an increase in required buffer size, while throughput remains unchanged. The increase in frame size decreases the performance. This is because larger frames increase the bit streaming rate.

(Equation 6.2) Frame size * Frames a second = bit streaming rate.

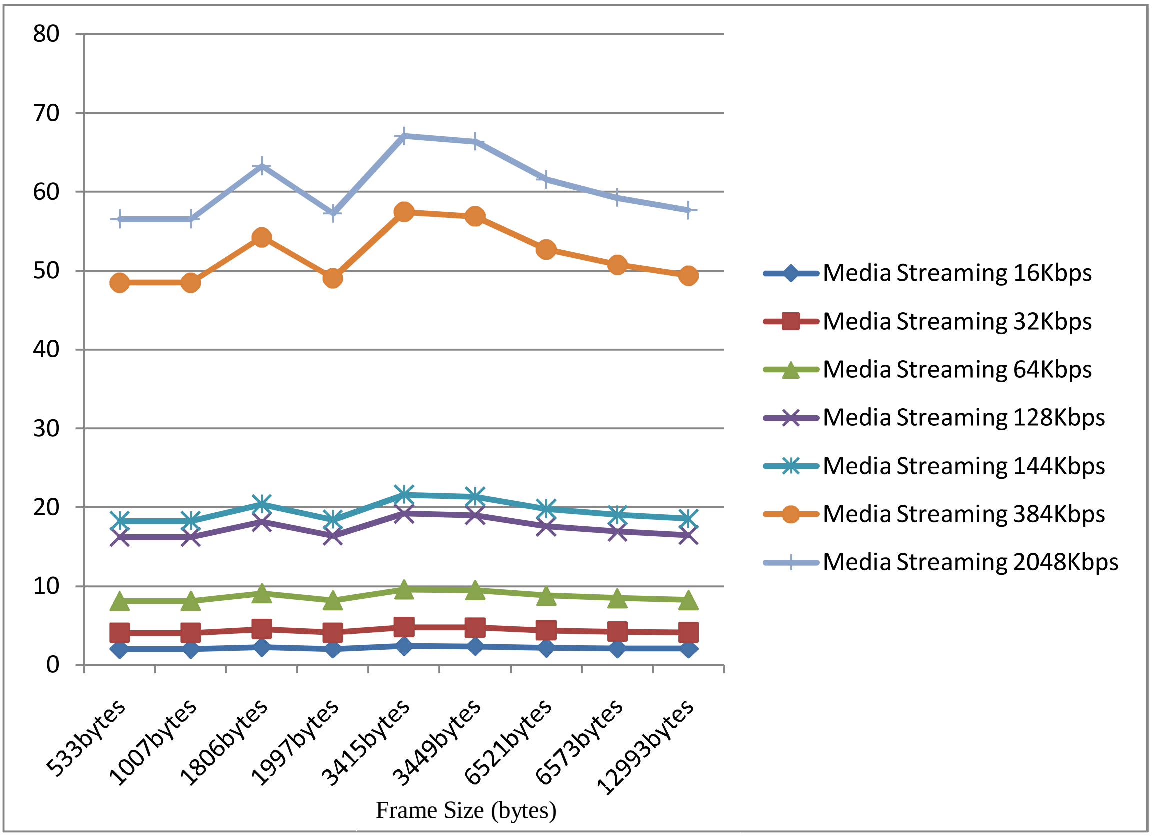

Figure 6.27 Media streaming throughput achieved by each bandwidth.

The increase in frame size is multiplied by the number of frames a second and a small increase in frame size could cause a large increase in bit streaming rate.

A bandwidth channel is capable of a maximum throughput. If the media streaming rate is lower than this maximum then the user receives data at the media streaming rate and is able to playback media at the target playback rate. Only a small buffer size will be required. If the media streaming rate is higher than the maximum channel rate then the user receives data at the maximum channel rate. The media then needs to be buffered to allow for smooth playback. The greater the margin between the media streaming rate and the channel's maximum rate the larger the required buffer needs to be.

6.9.5 Conclusion

How network factors affected the performance of media streaming was investigated. Simulations were designed which showed how network factors: end-to-end delay, bandwidth, and frame rate and frame size affect media streaming performance. The bandwidth is important and the throughput of the assigned bandwidth needs to be higher than the required bit rate. The frame size and frame rate affect the required bit rate and in turn affect the bandwidth needed for successful media streaming. The end-to-end delay does not affect the media streaming service. The media streaming service will perform best with a high bandwidth.

Bandwidth has drastic affects on the performance of media streaming. An increase in bandwidth leads to an increase in throughput and a decrease in the size of the required buffer. Once the size of the required buffer is close to zero seconds, then a further increase in bandwidth will not yield performance increases. It is wasteful to increase bandwidth once ~0% delayed frames has been achieved. The obtained throughput will be dependent on assigned bandwidth and media streaming rate (required bit rate).

Media streaming will take place at the video streaming rate if this is less than the assigned channel rate or at the assigned channel rate if the video streaming rate is greater than the assigned channel rate. The greater the difference between the video streaming rate and assigned channel rate the greater the size of the required buffer, when the video streaming rate is larger than the assigned channel rate. This causes re-buffering, the playback of the video is paused and the receiver buffers video for a predefined number of seconds before resuming playback. The greater the difference between the assigned channel rate and the video streaming rate the more often re-buffering takes place.

The required bit rate, which is equivalent to the media streaming rate, is affected by both the frame rate and the frame size. The following formula gives the required bit rate:

(Equation 6.2) Frame size * Frames a second = bit streaming rate.

It can be seen from this equation that an increase in either the frame rate or the frame size leads to an increase in the need for re-buffering. This was shown in simulations. A small increase in frame rate or frame size can cause a large increase in required bit rate, for this reason changing the frame rate or frame size could take a video streaming from smooth playback to many pause and re-buffering sessions.

The different genres of video need different frame rates to be acceptably viewed by viewers. The frame rate required by news style video is low because scenes don't change often, while action video requires a high frame rate to be viewed acceptably.

The end-to-end delay does not have an affect on throughput or the required buffer size. This is because media is streamed at a constant rate without feed back being required from the receiver. Each frame is received relative to the frame before it; this is the cause of end-to-end delay having no affect on the media streaming service.

The bandwidth is the most important factor of the media streaming service and the bandwidth used is dependent on the required bit rate. The maximum number of users of the system will depend on which type of video each user is streaming. The range will be 105-200Kbps which will require 128Kbps bandwidth with a small amount of pre-buffering or 384Kbps and above for ~0% buffering.

Chapter 7

7 Conclusion and Future Work

In an attempt to measure network factors affect on application performance in 3G UMTS networks, simulation experiments were setup. During simulations network factors were varied through legitimate values and the outcomes recorded. These outcomes showed how each factor affected each application and gave an insight to the required conditions for an application to run acceptably. Simulations were done using NS2 simulator and the EURANE extension. Thousands of network combinations were used and the results recorded. This was done by automating scripts which edited and re-executed NS2 OTcl scripts and then extracted the results. The results were analysed to show how each network factor affected the performance of each application.

Performance is defined for this research as the throughput achieved by an application. The network factors examined were: air interface bandwidth, end-to-end delay and size of target data to be transferred over the UMTS network. The effects of these factors were investigated for all the following applications: FTP, SMS, MMS, email, web browsing, media streaming, media broadcasting and video calling. The results were then recorded and analysed.

How network factors affect the performance of an application when these factors are varied through a range of legitimate values?

Which network factors have the most significant impact on a specific application?

Are there minimum conditions needed for an application to run acceptably?

What is the recommended number of simultaneous users of the application?

The results were discussed in chapter 6. These gave a more clear idea of the performance expected from 3G UMTS networks for different network conditions. Some of the notable results are that in media streaming applications the end-to-end delay has virtually no effect on performance, while bandwidth has a significant effect on performance (These are taken in zero bit error network). In FTP applications, TCP congestion control manages the flow of data and for small file transfers the TCP window size never grows large enough to utilise a high bandwidth rendering the bandwidth ineffective. But for large file transfers there is a notable difference.

The FTP, Email and MMS applications are similar in nature and the results obtained from their simulations were similar too. All three applications pass a file from a source to a destination and run above the TCP protocol. The following network factors effects on these applications were investigated: end-to-end delay, bandwidth and file size (email or message size). It was found that a shorter end-to-end delay always improved the service and a higher bandwidth too always improved the service. The size of the file (email or message size) in combination with bandwidth affected the performance. This indicates that the best case for these applications will be low end-to-end network delay and high bandwidth.

A high bandwidth is capable of a higher maximum data rate than a low bandwidth. To improve the FTP, email and MMS services to allow for higher throughput a higher bandwidth can be used. The return on throughput for an increased bandwidth is logarithmic. It would be better for a network to assign a medium level bandwidth to each user, in such maximising the cells throughput and performance.

The effects of file size on FTP, email and MMS showed that when a small file (email or message) was transferred all bandwidths performed similarly. While in a large file (email or message) transfer the high bandwidth performed much better than the low bandwidth. Where the bottleneck in the end-to-end delay occurs affected the performance. If the bottleneck was at the receiving node it had the greatest possibility of causing RTO.

FTP, email and MMS services will be usable under all network conditions, as long as constant RTO does not occur. It is suggested that a cell should use all extra available bandwidth to improve these services. Bandwidth is the most important factor for these services. Based on the simulation results and number of simultaneous users for a given bandwidth, it is suggested that FTP use the 64Kbps data channel, that email use the 128Kbps channel for large emails and the 64kbps channel for small emails, that MMS use the 128Kbps channel. This would maximise network capacity, while not causing constant RTO.

SMS service is a low resource service and performance is affected insignificantly by changes in end-to-end delay, bandwidth and size of the message. 3G will have no noticeable effect on the performance of the SMS service. A SMS message takes only a few hundreds of a second longer in a long end-to-end delay network compared to a short end-to-end delay network. The size of a SMS is very small and a change in bandwidth has no affect on the performance.

In the case of a multiple part SMS, each SMS part is sent independently and so it does not impact on performance. The nature of SMS with its small message payload and independent delivery make it suitable to be delivered via the signalling channels. To maximise the performance of the SMS service, it should continue to function as is, by transferring SMS messages across signalling channels as soon as a signalling channel becomes available.

The effects of network factors on HTTP web browsing show that a shorter end-to-end delay always improved the service and a higher bandwidth too always improved the service. The size of the page in combination with the bandwidth affected the performance. The effects of caching increased performance, while a page consisting of many small objects decreased performance. This indicates that the best case for the HTTP web browsing service will be low end-to-end network delay and high bandwidth, with a cached web page that consists of a few objects.

The HTTP protocol runs above the TCP protocol and TCP congestion control affects HTTP in the same manner it does to FTP over TCP. A higher bandwidth is capable of a higher maximum data rate, while the return on throughput for increasing the bandwidth is logarithmic. A small page took the same time to download on both high and low bandwidths, while there was a notable difference in performance for bandwidths when a large page was downloaded.

The higher end-to-end delay decreases performance, by adding a few hundreds of a second to the transition time for each packet transferred. Both caching and the number of objects a page consist of; has the same effect as a longer end-to-end delay and decreases performance. Objects that are not cached need to be downloaded from the server of origin, which then increases the time taken to download. Simulations showed that these factors tend to add an almost fixed overhead regardless of the bandwidth used. The fixed overhead increases download time as a percentage more for a high bandwidth than a low bandwidth. For this reason high bandwidth is affected more by many small objects and cached objects than low bandwidth.

The factor which affects results the most would be the end-to-end delay. The short delay network performed twice as well as the two long delay networks. Based on the simulation results and the number of simultaneous users for a given bandwidth, it is suggested that the 384Kbps channel be used as this will maximise the networks capacity.

How network factors affected the performance of media broadcast, video call and media streaming was investigated. The factors examined were end-to-end delay, bandwidth, frame rate and frame size. The bandwidth is important and the throughput of the assigned bandwidth needs to be higher than the required bit rate. The frame size and frame rate affects the required bit rate and in turn affects the bandwidth needed for successful media services. The end-to-end delay does not have an effect on the performance of these services. Video call is only affected by the propagation delay which leaves a lag in a conversation if the round trip delay is greater than 200ms. These services will perform best with a high bandwidth.

Bandwidth has a dramatic effect on the performance of media services. An increase in bandwidth will lead to an increase in throughput and a drop in the percentage of delayed frames of broadcast media and video calling services. While an increase in bandwidth will lead to a smaller required buffer size for the streaming media service. Once ~0% frames are delayed or a ~0 second buffer is achieved it becomes wasteful to increase the bandwidth any further. The obtained throughput will be dependent on the assigned bandwidth and media service rate (required bit rate). This will be the broadcast rate for broadcast, video call rate for video calling, and the streaming rate for media streaming.

When the required rate is greater than the assigned channel rate the performance suffers and frame delay occurs. The performance decreases exponentially with a greater difference in required rate and assigned channel rate. This results in jitter of the received video for broadcasting and video calling, decreasing the quality of service received by the viewer. While for media streaming it causes re-buffering, playback stops and the application re-buffers, this also decreases the quality of the service received by the viewer. An increase in required bit rate is likely to reduce performance.

The required bit rate, which is affected by both the frame rate and the frame size, is giving by the following formula:

(Equation 6.2) Frame size * Frames a second = bit streaming rate.

It can be seen from this equation that an increase in either the frame rate or the frame size will lead to an increase in the number of delayed frames. The different genres of video need different frame rates to be viewed acceptably by viewers. The frame rate required by news style video is low because scenes don't change often, while action video requires a high frame rate to be viewed acceptably.

There is no maximum number of users of the broadcast service; all users tuned in to the broadcast will receive the broadcast. It is recommended that the lowest bandwidth which has ~0% delayed frames be used. The maximum number of users of the media streaming service will depend on which type of video each user is streaming. The range is 105-200Kbps which requires 128Kbps bandwidth with a small amount of pre-buffering or 384Kbps and above for ~0% buffering.

A video call would use a low frame rate with either a small or medium size frame, which will result in a required bit rate of 21-71kbps based on Table 6.2. Using the lowest bandwidth that has ~0% delayed frames. The experimental simulations show that the 64Kbps bandwidth be used for a small frame size, while the 128Kbps bandwidth should be used for the medium size frame resolution.

The dissertation contributes in several aspects. It serves as a good introduction to the field, giving an overview of 3G UMTS networks and wireless applications. It also contributes the answers of the research questions investigated. Finally, it serves as a good first stage for potential future research.

A brief history of wireless telecommunication is given, followed by the structure and features of 3G UMTS networks. This gives the reader a quick grasp and understanding of how 3G developed and the needs which drove 3G's development. Then components constituting a 3G network are outlined and individual elements and protocols described. This highlights the inherent constraints of 3G, like the limited radio spectrum, limited power and relatively high latency compared to wired networks. The eight applications simulated are described and the results reported, this informs interested readers about how these applications work and its expected performance.

The major contribution is the answers of the proposed research questions. How network factors affect the performance of an application when these factors are varied through a range of legitimate values? This research presents the effects of three important network factors on eight popular applications, which includes a discussion and conclusion of the results. It is beneficial to have these discussions and results contained in a single source, making it an asset to anyone interested in the topic. At the start of this research no such single source existed.

The resourcefulness of the source is improved by the answers of three additional research questions. A few interesting results were discovered and theoretical expected results confirmed. As a source it could be useful to someone interested in the subject, a network designer or someone looking for a result comparison. This research also serves as a base for future research.

Future research could focus on modifying the protocols used by an application to tailor the protocol for the target application to improve its performance. Also, later generations of wireless applications can perform the same experiments and compare the results and in showing the performance increase gained by the new generation. This dissertation will be useful for anyone interested in the field of application performance in a 3G UMTS network.

This research and these simulations show how a network factor affects an application's performance, which gives a certain degree of insight into the applications nature. Future research would be required to investigate the transport level protocol and investigate how it can be modified to optimise a target application's performance. Each application would need to be analysed to identify modifications expected to improve performance. Once the transport level protocol modification has been made the simulations can be re-run and the results compared to this original results. The results will discuss if and why these modifications were successful, or why it was not. Discoveries made will hopefully in future pass on to real 3G applications improving its performance.

References

3G.co.uk. (No Date). Introduction to 3G, What 3G does, Where launched, UK Special. [Online]. Available URL: http://www.3g.co.uk/All%20About%203G.htm, February, 26, 2006

3G.co.uk . (14 February 2006). TV Phone Sales To Escalate To A $30 Billion Megavendor-Dominated Market. [Online]. Available URL: http://www.3g.co.uk/PR/Feb2006/2618.htm, February, 23, 2006.

ADC NewNet, Inc. (1999) Wireless Short Message Service Tutorial. Published by ADC NewNet. Shelton City, USA. [Online]. Available URL: http://www.mobilein.com/SMS_tutorial.pdf, September, 11, 2008.

Agarwal, N., Chandran-Wadia, L. and Apte, V. (2004). Capacity Analysis of the GSM Short Message Service. Proceedings of National Conference on Communications (NCC 2004). Bangalore, India, 2004.

Akl, R. and Son Nguyen (2006). Capacity allocation in multi-cell umts networks for different spreading factors with perfect and imperfect power control. Proceedings of the IEEE Consumer Communications and Networking Conference, 2006, 928-932.

Basso, A., Kim, B. J. and Jiang, Z. (2002). Performance evaluation of MPEG-4 video over realistic EDGE wireless networks. Proceedings from Wireless Personal Multimedia Communications, the 5th International Symposium 2002. **3**, 1118-1122.

bizcommunity.com. (11 September 2008). Nokia brings MS Exchange ActiveSync corporate mobile email solution to 80 million mobile devices. [Online]. Available URL: http://www.bizcommunity.com/Article/196/78/28273.html, September, 13, 2008.

Bleidt, R. (2004). Mpeg-4 and The Future of Mobile Video. Software Development Forum Multimodal SIG. Presentation originally given the March 2004. A copy available URL: http://streamcrest.com/SDF%20Final1.pdf

Brasche, G. and Walke, B. (1997). Concepts, Services, and Protocols of the New GSM Phase 2+ General Packet Radio Service. Communications Magazine, IEEE, **35** (8), 94-104.

Camarillo, G and Garcia-Martin, M. (2006). The 3G IP Multimedia Subsystem (IMS): Merging the Internet and the Cellular Worlds. Second Edition. Chichester: John Wiley & Sons.

Chakravarthy, C. V. (2006). IP Multimedia Subsystems: A Tutorial. IMSEXPO. San Diego, USA

Chakravorty, R., Banerjee, S., Rodriguez, P., Chesterfield, J., Pratt, I. (2004). Performance Optimizations for Wireless Wide-area Networks: Comparative Study and Experimental Evaluation. Proceedings of the 10th annual international conference on Mobile computing and networking. Sponsored by SIGMOBILE. Published by ACM, New York, USA. 159-173.

Chakravorty, R., Chesterfield, J., Rodriguez, P. Banerjee, S. (2004). Measurement Approaches to Evaluate Performance Optimizations for Wide-Area Wireless Networks. Fifth Passive and Active Measurement (PAM) Workshop. Sophia-Antipolis, France. Published by Springer-Verlag, Berlin, Germany. 257-266.

Chan, Y. (2006). Evaluation of Traffic Prediction Based Access Control Using Different Video Traffic Models in 3G CDMA High Speed Data Networks. Unpublished Masters thesis. University of Waterloo.

Chesterfield, J., Chakravorty, R., Crowcroft, J., Rodriguez, P. and Banerjee, S. (2004). Experiences with multimedia streaming over 2.5G and 3G Networks. Proceedings from IEEE BroadWiM conference. San Jose, USA.

Chung, J. and Claypool, M. (No Date). NS by Example. Tutorial. Worcester PolyTechnic institute. [Online] Available URL: http://nile.wpi.edu/NS, May, 23, 2007.

Cisco Systems Inc., (No Date). Voice over IP Overview. Published by Cisco Systems Inc. [Online]. Available URL: http://www.cisco.com/univercd/cc/td/doc/product/access/acs_mod/1700/1751/1751swg/intro.pdf, December, 30, 2008

Crocker, D.H. (1982). RFC822: Standard for ARPA Internet Text Messages. RFC published by The Internet Society.

Dubois, X. (2005). Performance of Different TCP Versions over UMTS Common/Dedicated Channels. Unpublished Masters thesis. University of Namur.

Enck, W., Traynor, P., McDaniel, P, and La Porta, T. (2005). Exploiting Open Functionality in SMSCapable Cellular Networks. Proceedings of the 12th ACM conference on Computer and communications security. pp393 – 404. New York, USA.

EURANE: Enhanced UMTS Radio Access Networks Extensions for NS2. (User Guide Release 1.6). (version dated 22 September 2005).

Ferng H., Tsai Y. (2003). Channel allocation and performance study for the integrated GSM/GPRS system. Wireless Communications and Networking, 2003. WCNC 2003. 2003 IEE, **3**, 1861-1865. New Orleans, USA.

Fielding, R., Gettys, J., Mogul, J., Frystyk, H., Masinter, L., Leach, P., Berners-Lee, T. (June 1999). Hypertext Transfer Protocol -- HTTP/1.1, RFC published by The Internet Society.

Fisher, A. (2005). How to Build a "Lean and Mean" Video Gateway Using 3G-324M–over–IP. Published by Surf Communication Solutions, Yokne'am, Israel.

Fledderus, E., Springer C. (2005). Wireless Network Simulation – Your Window on Future Network Performance. Wireless Personal Communications, **33**, 319-325.

Graphics and Media Lab at Michigan University, (no date). Hyper Text Transfer Protocol. [Online], Available URL: http://graphics.cs.msu.ru/courses/wp_el00/Internet/HTTP/article.html, September, 21, 2008.

Gurtov, A. And Floyd, S. (2004). Modelling Wireless Links for Transport Protocols. ACM SIGCOMM Computer Communications Review, **34** (2), 85-96.

Hamalainen, S. (2003). WCDMA Radio Network Performance. Unpublished Masters Thesis. University of Jyvaskyla

Hardy, E. (28 January 2008). Skyfire Mobile Web Browser Preview. [Online]. Available URL: http://www.brighthand.com/default.asp?newsID=13761, September, 14, 2008.

Hartung, F., Horn, U., Huschke, J., Kampmann, M., Lohmar, T and Lundevall, M. (2007). Delivery of Broadcast Services in 3G Networks. IEEE TRANSACTIONS ON BROADCASTING, **53** (1), 188-199.

Hewlett-Packard Development Company, (2007). Accelerating 3G mobile video communications. Published by Hewlett-Packard Development Company. [White Paper]. Available URL: http://h20208.www2.hp.com/opencall/library/solutions/mv/4aa1-1548enw_mv_wp.pdf, September, 21, 2008.

Isode. (No Date) IETF and OMA Architectures for Mobile Email. [White Paper]. Available URL: http://www.isode.com/whitepapers/oma-ietf-mobile.html, September, 10, 2008.

Jain, R., bin Tariq, M., Kempf, J. and Kawahara, T. (2005). The All-IP Next-generation Network Architecture. Etoh, M. Next Generation Mobile Systems 3G and Beyond. Chichester, John Wiley & Sons, Ltd.

Klensin, J. (April 2001) Simple Mail Transfer Protocol (RFC2821). RFC published by The Internet Society.

Knutsson, B. (2004). Simulation of radio resource management for UMTS. Unpublished Masters thesis. University of Linköping.

Koumpis K., Cvetkovic S. and Peersman G. (1999). Performance evaluation of SMS-based email and voicemail notification architecture. In Proc. of the 5th Workshop on Emerging Technologies in Telecommunications, 282-286. Bayona, Spain.

Le Bodic, G. (2002). Mobile Messaging Technologies and Services: SMS, EMS and MMS. New York: John Wiley & Sons, Inc.

Lo, A., Heijenk, G. and Niemegeers, I. (2006). Performance Evaluation of MPEG-4 Video Streaming over UMTS Networks using an Integrated Tool Environment. Proceedings of SPECTS, International Symposium on Performance Evaluation of Computer and Telecommunication Systems 2005 conference. Philadelphia, USA.

Magedanz, T., Witaszek, D., Knuettel, K. and Weik, P. (2005). The IMS Playground at FOKUS - An Open Testbed for Next Generation Network Multimedia Services S. Proceedings of the First International Conference on Testbeds and Research Infrastructures for the Development of Networks and Communities, **1**, 2-11. Published by IEEE Computer Society, Washington, USA.

Marshall, B. and Crosby, T. (18 October 2007). How E-mail Works. [Online]. Available URL: http://communication.howstuffworks.com/email.htm, September, 13, 2008.

Mehra, M. (29 April, 2008). Mobile Email versus SMS Accessing Emails On Mobile Phones. [Online]. Available URL: http://www.promotionworld.com/e-mail/articles/080429mobileemailversus.html, September, 13, 2008.

Mirial, (2007). 3G-to-TV Video Calling: The latest trends in participation TV and best-practice implementation for 3G video broadcasting TV. Published by Mirial. Available URL: http://www.mirial.com/pdf/Whitepaper/IP-to-TV_Video_Calls.pdf, September, 8, 2008.

Moltchanov, D., Koucheryavy, Y. and Harju, J. (2002). Performance Evaluation of Multimedia Services in Heterogeneous Wireless Environment Under Different Mobility Patterns. Proc. Conf. on Network Control and Engineering for QoS, Paris, France.

Now.SMS. How MMS Works. [Online]. Available URL: http://www.nowsms.com/howmmsworks.htm, September, 9, 2008.

NS-2 Tutorial. (No Date). [Online]. Available http://216.239.59.104/search?q=cache:LL55aKfGcL8J:pages.cpsc.ucalgary.ca/~gongm/CPSC441/NS-2.ppt+%24ftp+produce+%3Cn%3E+ns2&hl=en&ct=clnk&cd=4&gl=za, May, 23, 2007.

Ojala, K. (2000) 3G Radio Network. White paper published by Teliasonera, Finland. Available URL:

Panian, J. (2004). TCP Window Size for WWAN. Portable Computer and Communications Association, Newark. Published by Qualcomm. Available URL: http://www.pcca.org/standards/architecture/tcp_window_size.pdf, December, 8, 2007.

Ries, M., Crespi de Arriba, C., Nemethova, O. and Rupp, M. (2007). Content Based Video Quality Estimation for H.264/AVC Video Streaming. Proceedings of IEEE Wireless and Communications & Networking Conference, Hong Kong.

Roccetti, M., Salomoni, P., Ghinim V. and Ferretti, S. (2005). Bringing the Wireless Internet to UMTS Devices: A Case Study with Music Distribution. Multimedia Tools and Applications. **25** (2), 217-251.

Seixas, M. and Palma, J. (2006). Remote Alarm Command System for Residential Domotics Through GSM – SMS. Industrial Journal of Computer Applications in Technology **25** (4), 227-233.

Snap9 Corporation. Cellular Phone. (2006). [Online]. Available URL: http://www.thecellularphoneguide.com/ September, 6, 2006.

Son Nguyen, B. S. (2005). Capacity and Throughput Optimiztion in Multi-cell 3G WCDMA Networks. Unpublished Masters thesis. North Texas University.

Song, S., Won, Y. and Song, I. (2002). Empirical Study of User Perception Behavior for Mobile Streaming. International Multimedia Conference. Proceedings of the tenth ACM international conference on Multimedia. Juan-les-Pins, France.

Sungkasap, M., Malisuwan, C. S. and Ungvichian, V. (2008). WCDMA Mobile Internet in High-Mobility Environment Case Study on Military Operations of the Royal Thai Armed Forces. International Journal of the Computer, the Internet and Management, **16** (3), 34-40.

Tektronix Inc., (2004). WCDMA/UMTS wireless Networks. Published by Tektronix Inc., USA. [Online] Available URL: http://www.tek.com/Measurement/App_Notes/2EW_17289/eng/2EW_17289_0.pdf, May, 2, 2007.

TeliaSonera. (2004) Streaming in Mobile Networks. White paper published by TeliaSonera. Finland. Available URL:

http://www.medialab.sonera.fi/workspace/StreaminginMobileNetworksWP.pdf, April, 4, 2007.

The NS2 Manual. (version dated 12 March 2007). The latest NS2 manual is available at URL: http://www.isi.edu/nsnam/ns/doc/ns_doc.pdf, May, 23, 2007.

The Shosteck Group, (2001). GSM OR CDMA: The Commercial and Technology Challenges for TDMA Operators. Published by CDMA Development Group, Costa Mesa, USA.

Timm-Giel, A. (2004). UMTS/GPRS Performance Measurements and Evaluation. ITG FG521-Workshop 20. February 2004.Hamburg, Germany.

UK Telematics online. (No Date). Wireless Wide Area Networks (WWAN) [Online]. Available URL: http://www.uktelematicsonline.co.uk/html/wireless_wan_s.html September, 21, 2006.

UMTS World. (No Date) UMTS Capacity Planning. [Online]. Available URL: http://www.umtsworld.com/technology/capacity.htm, July, 26, 2007.

Virpi, R. (2006). Web Browsing on Mobile Phones - Characteristics of User Experience. Unpublished PhD thesis. Helsinki University of Technology.

Wang, H. And Prasad, D. (2005). End-2-End QoS Provisioning in UMTS networks. Unpublished Masters thesis. Aalborg University.

WAP Application Protocol Forum, Ltd. (2001) WAP MMS Architecture Overview. Published by WAP Application Protocol Forum. [Online]. Available URL: http://www.openmobilealliance.org/tech/affiliates/wap/wap-205-mmsarchoverview-20010425-a.pdf, September, 8, 2008.

Weber, R., Guerra, M., Sawhney, S., Golovanevsky, L. and Kang, M. (2006). Measurement and Analysis of Video Streaming Performance in Live UMTS Networks. Proceedings of WPMC 2006 Conference. San Diego, California, USA.

Figure 3.1 Source Anthony Chan EEE401S course notes.

Figure 5.1 Source NS2 Manual (version dated 12 March 2007).

Appendix A

Appendix A defines the meaning of bandwidth as used in this research. Bandwidth is defined as rate of data transfer, throughput or bit rate, measured in bits per second. Bandwidth is the same as the dedicated channel assigned to the mobile device for data transfer. Bandwidth implies ideal rate.

Appendix B

Appendix B contains an example of an OTcl script used in this research.

```
#These simulation puts the round trip delay of the network at 340ms. And
the delay can be either is the radio interface or server. bottlenecks.


global ns
```

```
# Remove all Packet headers and add only those that are required.
# This significantly reduces the memory requirements of large simulations
remove-all-packet-headers
add-packet-header MPEG4 MAC_HS RLC LL Mac RTP TCP IP Common Flags

set ns [new Simulator]

$ns color 1 blue
$ns color 2 red

set f [open out.tr w]
$ns trace-all $f
set nf [open out.nam w]
$ns namtrace-all $nf

proc finish {} {
      global ns
      global f
      global nf
      $ns flush-trace
      close $f
      close $nf
#     exec nam out.nam &
      puts " Simulation ended."
      exit 0
      }

$ns node-config -UmtsNodeType rnc

# Node address is 0.
set rnc [$ns create-Umtsnode]

$ns node-config -UmtsNodeType bs \
            -downlinkBW 384kbs \
            -downlinkTTI 134.5ms \
            -uplinkBW 384kbs \
            -uplinkTTI 134.5ms \
            -hs_downlinkTTI 134.5ms \
            -hs_downlinkBW 384kbs \

#Node address is 1.
set bs [$ns create-Umtsnode]

#Interface between RNC and BS
$ns setup-Iub $bs $rnc 622Mbit 622Mbit 15ms 15ms DummyDropTail 2000

$ns node-config -UmtsNodeType ue \
            -baseStation $bs \
            -radioNetworkController $rnc

# Node address for ue1 and ue2 is 2 and 3, respectively.
set ue1 [$ns create-Umtsnode]
set ue2 [$ns create-Umtsnode]

# Node address for sgsn0 and ggsn0 is 4 and 5, respectively
set sgsn0 [$ns node]
set ggsn0 [$ns node]

# Node address for node1 and node 2 is 6 and 7, respectively.
set node1 [$ns node]
set node2 [$ns node]
```

```
# Connections between fixed network nodes
$ns duplex-link $rnc $sgsn0 622Mbit 0.4ms DummyDropTail 2000
$ns duplex-link $sgsn0 $ggsn0 622MBit 10ms DummyDropTail 2000
$ns duplex-link $ggsn0 $node1 622MBit 10ms DummyDropTail 2000
$ns duplex-link $node1 $node2 622Mbit 0.5ms DummyDropTail 2000

# Routing gateway
$rnc add-gateway $sgsn0

# Agent set-up for ue1
set tcp0 [new Agent/TCP/Vegas]
$tcp0 set fid_ 0
$tcp0 set prio_ 2
$tcp0 set window_ 17
$tcp0 set packetSize_ 1000
$tcp0 set slow_start_restart_ false

# Agent set-up for ue2
set tcp1 [new Agent/TCP]
$tcp1 set fid_ 1
$tcp1 set prio_ 2

# Attach agents to a common fixed node
$ns attach-agent $node2 $tcp0
$ns attach-agent $node2 $tcp1

# Create and connect two application to their agent
set ftp0 [new Application/FTP]
$ftp0 attach-agent $tcp0
$ftp0 set type_ FTP


set ftp1 [new Application/FTP]
$ftp1 attach-agent $tcp1

# Create and attach sinks
set sink0 [new Agent/TCPSink]
$sink0 set fid_ 0
$ns attach-agent $ue1 $sink0

set sink1 [new Agent/TCPSink]
$sink1 set fid_ 1
$ns attach-agent $ue2 $sink1

# Connect sinks to TCP agents
$ns connect $tcp0 $sink0
$ns connect $tcp1 $sink1

$ns node-config -llType UMTS/RLC/AM \
            -downlinkBW 384kbs \
            -uplinkBW 384kbs \
            -downlinkTTI 134.5ms \
            -uplinkTTI 134.5ms \
            -hs_downlinkTTI 134.5ms \
            -hs_downlinkBW 384kbs

# Create HS-DSCH and attach TCP agent for ue1
$ns create-hsdsch $ue1 $sink0

# Attach TCP agent for ue2 to existing HS-DSCH
```

```
$ns attach-hsdsch $ue2 $sink1

#Loads input tracefiles for each UE, identified by its fid_
$bs setErrorTrace 0 "UE1_trace_file3"
$bs setErrorTrace 1 "UE2_trace_file3"

# Load BLER lookup table from file SNRBLERMatrix
$bs loadSnrBlerMatrix "SNRBLERMatrix"

# Tracing for all HSDPA traffic in downtarget
$rnc trace-inlink-tcp $f 0
$bs trace-outlink $f 2

# UE1 Tracing
$ue1 trace-inlink $f 2
$ue1 trace-outlink $f 3
$bs trace-inlink $f 3
$ue1 trace-inlink-tcp $f 2

# UE2 Tracing
$ue2 trace-inlink $f 2
$ue2 trace-outlink $f 3
$bs trace-inlink $f 4
$ue2 trace-inlink-tcp $f 2

$ns at 0.0 "$ftp0 produce 5000"
#$ns at 0.0 "$ftp0 start"
$ns at 3000.1 "$ftp0 stop"
$ns at 3000.1 "finish"

puts " Simulation is running .. please wait ..."

$ns run
```

Appendix C

Appendix C contains an extract of a results file used in the research.

```
Frames a seconds: 5.0, Packet size: 107.0, Frame size: 533.0

Percentage less than min delay: 99.98

Percentage Greater than min delay: 0.02

Target min delay:0.0401500938086304

Average time:0.0022749142

Mean packet time: 0.0022289999999998145

Mean packet amount: 2248

The buffer time up only: 0.2324839061913696

The buffer time up and down:-189.37589804314317

Throughput per seconds: 47034.74091462439
```

The total number of packets: 5000

Frames a seconds: 10.0, Packet size: 107.0, Frame size: 533.0

Percentage less than min delay: 99.98

Percentage Greater than min delay: 0.02

Target min delay:0.0200750469043152

Average time:0.0022749142

Mean packet time: 0.0022289999999998145

Mean packet amount: 2248

The buffer time up only: 0.25255895309568477

The buffer time up and down:-89.00066352157262

Throughput per seconds: 47034.74091462439

The total number of packets: 5000

Frames a seconds: 14.992503748125937, Packet size: 107.0, Frame size: 533.0

Percentage less than min delay: 99.98

Percentage Greater than min delay: 0.02

Target min delay:0.013390056285178236

Average time:0.0022749142

Mean packet time: 0.0022289999999998145

Mean packet amount: 2248

The buffer time up only: 0.25924394371482173

The buffer time up and down:-55.57571042588564

Throughput per seconds: 47034.74091462439

The total number of packets: 5000

Frames a seconds: 5.0, Packet size: 267.0, Frame size: 533.0

Percentage less than min delay: 99.95

Percentage Greater than min delay: 0.05

Target min delay:0.10018761726078801

Average time:0.005687289

Mean packet time: 0.005542000000000158

Mean packet amount: 483

```
The buffer time up only: 0.175786382739212

The buffer time up and down:-189.00065652157787

Throughput per seconds: 46946.79661961965

...

...
```