Mobile Multiplayer Gaming

CHRISTIAN WESTERMARK

KTH Information and
Communication Technology

Master of Science Thesis
Stockholm, Sweden 2007

COS/CCS 2007-20

Mobile Multiplayer Gaming

Christian Westermark
chrwes@hotmail.com

Stockholm
24th of June 2007

Supervisor & Examiner
Professor Gerald Q. Maguire Jr.
School of Information and Communication Technology (ICT)
KTH / Royal Institute of Technology

Industry Advisor
Tord Westholm
Ericsson AB, Research

Abstract

The last ten years, online multiplayer games have become very popular. During the same time period mobile terminals and cellular networks have undergone a tremendous technical evolution. Therefore it is natural to wonder why we have not seen an online mobile multiplayer gaming revolution yet. The answer to this question is of great value for companies selling mobile systems. This answer is important in order to understand how to fill up today's empty networks with traffic and what kind of traffic these games will generate.

This thesis is a continuation of Mattias Åkervik's thesis. It gives the reader an understanding of what kind of wireless technologies are on the market today and how they perform. Given this performance background, some suitable games were chosen to examine how they perform over a particular cellular network and to determine the perceived gaming quality that a user experience. The thesis also examines the particular packet traffic characteristics generated by these games to gain a better understanding of how to better adapt cellular networks towards gaming.

Finally the market will be analyzed. Not only how large this potential market is, but to examine if there are some market issues preventing the revolution in network cellular on-line multiplayer games.

Abstract på Svenska

De senaste tio åren har multiplayer gaming blivit väldigt populärt. Under samma tidsperiod har cellulära terminaler och de mobile nätverken genomgått en stor teknisk evolution. Därför kan man undra varför inte mobile multiplayer gaming har slagit igenom än. Svaret på denna fåga är värdefullt för föreatag som säljer mobila system, men även mobiloperatörerna, dådet kan ge en hint hur man bättre kan fylla ut dagens tomma 3G system med trafik.

Detta examensarbete är en fortsättning på Mattias Åkerviks arbete. Det ger läsaren en förståelse i dagens trådlösa teknologier och vad de kan prestera. Med detta i bagaget kommer ett antal spel undersökas och hur deras spelupplevelse influeras av begräsningarna som de mobila nätverken har. Arbetet kommer också att behandla vilken trafik dessa spel genererar, då detta kan ge en bättre förståelse i hur man kan anpassa nätverken i ett gamingsyfte.

Slutligen kommer marknaden att analyseras. Inte bara det potentiella värdet av gamingmarknaden, men även om det finns några marknadsstrukturella orsaker som bromsar utvecklingen av mobila multiplayer spel.

Acknowledgements

This thesis would not have been possible without certain people. I want to thank Professor Gerald Q. Maguire for taking his time reading, correcting, and coming up with new useful ideas and suggestions for my thesis, always in a very fast time. I also want to thanks my industrial advisor Tord Westholm for giving me this opportunity to explore this very interesting field, in a world known company such as Ericsson, and providing a nice working environment. I also want to thank him for his comments, meeting bookings, and the contacts he gave me inside the company. I also wish to thank all the other employees of Ericsson who never hesitated to help me in a very useful way and also those who took their time to participate in the gaming survey. Finally I want to thank my wife. Without your support at home the result of this thesis would never had been as good as it became.

Table of Contents

List of figures

1 Introduction

Cellular phones have been around for a several decades, but only since the 1990s, thanks to the introduction of GPRS, has affordable data transfer been possible through GSM based cellular telephony networks. In the end of the 1990s the first mobile games came preloaded into the cellular terminals. Since that time cellular phones have become more advance and today are more a minicomputer rather than a phone. Thanks to this evolution today they offer an improved gaming experience together with games that can be installed and deleted on the terminals, just as on a regular computer. Also the means of data transfer through the cellular networks have become more sophisticated, with today's cellular network to a greater extend starting to reach the same performance as a fixed cable connection of a decade ago. Therefore one may wonder why we do not see any mobile multiplayer games in today's market.

1.1. Gaming trend

A "gamer" is not simply the end user of a video game, but the gamer can also be a player of traditional role playing games, games played with dice, or paper & pen. The stereotypical picture of a gamer is a young male who spends most of his spare time playing (video)games. This statement, however, is not supported by facts, which indicate that the average age of an American game buyer is 40 years old, with 38% of these players being females [1]. There are many types of gamers and they can be divided into different categories in order to get a better understanding of the gaming market.

The most common gamer is a casual gamer who plays sometimes, usually when he or she has some spare time. A casual gamer doesn't want to put a lot of work into learn the basics of a game. He or she wants to sit down with a new game and be able to play at once. Therefore many game developers incorporate an introduction tutorial into their more advanced games. This is done so that they can attract these casual gamers, who are a rather larger fraction of the end users who are interested in playing games.

Then we also have the hardcore gamers who are individuals who devote most of their time to play video games. Unlike the casual gamer, these gamers do not mind that there exist complexities in the games and many of them are proud that they have mastered them. Even if these gamers play a lot they only make up about 20-25% of today's gaming market [2].

With the internet revolution these gamers have moved away from the walkthroughs and FAQs provided by the gaming press, towards forums, where hardcore gamers publish a lot of information to gain respect. These hard core gamers also produce a lot of postings: describing what is good and bad about a game or what could have been done better. They usually complain a lot and compare them to other games which are better. These postings of different topics are common among hardcore gamers and create free publicity for the game. Therefore, even if these gamers are hard to satisfy and they don't constitute much market share, the game developers know that they have to satisfy these gamers to create a hit [2]. The forum postings, describing what is in and what is out, have caused hardcore gamers to have a similar attitude and opinions towards things. Nerveless all of these gamers do not play the same kind of games.

Competitive gamers are found among both hardcore and casual gamers. These gamers play multiplayer games online where they can compete against other gamers. For these players, this interaction gives the gaming experience a whole lot of more fun. The hardcore competitive gamers enjoy competing, and wining, so that they can then brag and tease the other gamers. Thanks to the internet revolution online multiplayer games where gamers can compete have become more and more popular. Today 44% of the most frequent players play their games online [1]. Another trend among competitive gamers, especially in Asia, is to compete with each other in televised tournaments, often with a big audience. The viewers are allowed to see the battle from both perspectives, and just as in a sporting event they have their favorites gamer who they cheer for. Therefore a few gamers have gained a lot of fame and have hundred of thousands of fans. This fame makes them very attractive from a commercial point of view, which generates a lot of money for the gamer. Also through the tournaments prize money the gamer can build up sustainable wealth.

Mobile gamers are gamers who play on their handheld telephones. Mobile gaming is a rather new invention and it really first began in 1997 when the game Snake was made available in handsets. A stereotypical picture of a gamer is a person who kills 3-5 minutes playing a mobile game while waiting in the supermarket, for the bus, or for class. However, this is not true. There are both hard core and casual gamers who enjoy playing mobile games for more than a few minutes. A study [3] from 2005, conducted by Sorrent shows that 60% of gamers are playing at home, that 65% play more than once a day, and that mobile gaming is gender neutral. Their research also states the following: [3]

- Both men and women (66 percent of men and 68 percent of women) play games at least once a day.
- 34% play more than 3 times a day, both male and female.
- Woman are slightly more likely to play at home (68% female-60% male)
- More than 60% play longer than 10 minutes each time, both male and female. This percentage also includes mobile gamers playing as much as 2 hours each time.
- More than 60% are singles between 18-26 in age.
- Most downloads of games are based on information the gamer got through “word of mouth”

Looking at what games mobile gamers play, there are some differences between the sexes. While there is an equality among genres like retro, casino, strategy/RPG, board, and trivia games, men play more sports (41% male-20%female) and action/adventure (39% male - 26% female). Women tends to play more classic games (40% female – 33% men) and puzzles games (64% female - 55% male) [3].

Mobile gamers are usually gamers who also play via other media such as PC, MAC and game consoles. Research conducted by Ziff Davis [4] concluded that 80% of all mobile gamers also own a gaming console; and 20% of the PC and console owners have purchased and downloaded mobile games to their cell phone terminals. The survey also suggests that the average game session is 17 minutes and that a mobile gamer plays 4.4 hours a week [2].

Thus we can conclude that mobile gamers play in both short and long game sessions, they play many different genres, gamers are both hardcore and casual gamers, and the gaming is rather neutral among the sexes (although there are differences by genre). Also if we widen the definition of mobile gamer, we also have to have in mind that a mobile gamer is a gamer who plays on any mobile device, and not specifically on a cellular phone.

Over last 10 years the gaming market has tripled and in the US alone games were sold for 7 billion USD in the year 2005, which are around 228 million units. [5] Roughly 69% of American heads of household play one or more computer game, and 44% of them play online. [1] Among the 56% who do not play online there is a large portion of casual gamers, who play single player games via their gaming consoles or PC without bothering with competition. This playing just for enjoyment is both commonly done alone or for socializing with the family. These gamers may read a gaming magazine to find out about new games, but do not put a lot of effort into posting in different online forums. Of course there are also hardcore gamers who play a lot of single player games, frequently posting pictures of new games, strategies, and discussions with other hardcore gamers. However, this is a small portion of the 56%.

The 44% of the gamers who play online are to some extent competitive gamers. Even if they are casual gamers, they will search for information and post in different online forums; this is because they are interested in winning, hence creating much more forum postings than a single player gamer. Of course there are also many hardcore gamers who enjoy playing online games and just as in the single player mode, they post a lot in different forums. Also in comparison to the traffic that the actual game play creates, one forum posting literally don't create any data transfer at all. These are a number of reasons why online gamers create a lot more internet traffic than single player gamers, even if you don't take into account the traffic that the actual game play generates, which is the biggest part.

Though companies like Ericsson who provide the market with top of the line cellular networks, are interested in what kind of traffic mobile gaming produces, this thesis will focus on these online multiplayer gamers, which as mentioned produces much more traffic than single player gamers. Though forum posting only produces a small fraction of the total gaming traffic, this thesis will only consider the traffic generated by the gaming session.

1.2. Problem statement

This thesis will illustrate the potential mobile traffic that gaming produces. We examine the PC-market where Ericsson has an interest, due to the newly developed UMTS modems connectable to stationary computer or laptops, which sends the traffic over 3G networks. Sadly cellular networks are not as reliable as fixed cable connections, so can today's cellular networks deliver an acceptable gaming quality for online gaming? In an operators point of view it would also be interesting to know what traffic patterns online gaming produces so that they better can adapt their networks towards gaming. This adaptation of course produces an extra cost, but maybe the revenue is bigger. Therefore it would also be interesting to know; what traffic online gaming produces and how much the gaming market is worth? Another question that will be answered throughout the report is what impedes on the development of mobile gaming?

1.3. Method

First this thesis will treat aspects what theoretical impact different network parameters have on gaming. Then it gives an insight how much today's cellular networks can perform. With this knowledge in hand a gaming survey will be conducted where gamers grade the gaming experience during different emulated network parameters. The result of the survey will be presented through graphs to easily see how gamers perceive the gaming experience under different circumstances. The same chapter will also compare these results with what today's cellular network performs, so that the reader easily can see what performance the gamers can expect under certain networks. The following chapter will give an insight to how cellular games must be adapted to the terminals and how the terminals can be improved towards gaming. Finally the market will be examined; the value chain presented to see if there are any market structures that hold the gaming phenomena back.

2. Gaming Networks

There are many network parameters that influence the gaming experience. This chapter will present them, and what can be done to avoid a declining gaming experience due to lack of network performance.

2.1. Network parameters

There are three main network parameters that affect a user's perception of a multiplayer online game: bandwidth, packet loss and latency. Additionally what kind of network you choose to setup, peer-to-peer or client-server, has an impact on the user's gaming experience. Therefore this chapter gives the reader a short introduction to these alternatives, so that he or she can understand the conclusions and the rest of the report.

2.1.1. Latency

Regarding mobile on line gaming the most crucial network aspect is latency. Latency is the time it takes for the terminal to send information to a server or to receive the reply. The round trip latency from the terminal to the server and server to terminal, is usually called the round trip time (RTT). This round trip latency varies greatly from network to network, wired networks to wireless mobile networks, and even as a function of congestion on a given network. Of course, it chiefly depends on where the server is situated with respect to the user's terminal. If you use the server in the same local network, you will of course have lower latency than if the server is on the other side of the world. This is why you usually play online games with others from the same continent.

Figure 1 and Figure 2 shows the ping times from a client to two different servers. The first one resolver1.opendns.com (208.67.222.222) is situated in London, and its RTT is around 31ms. The second one w2.rc.vip.scd.yahoo.com (66.94.234.13) is situated in USA, and therefore you should expect a longer RTT and after a looking at the ping times, that all are around 179ms, you understand that the significantly greater propagation time leads to a longer round trip time.

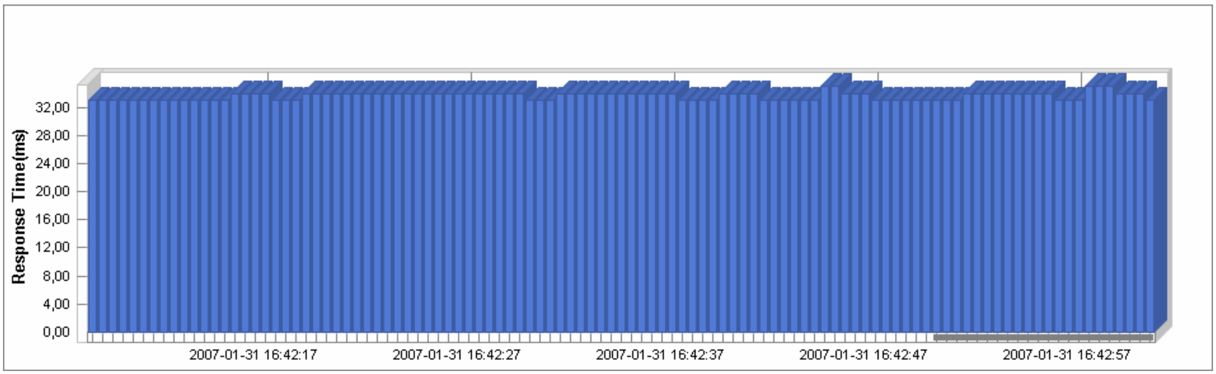

Figure 1: *Round trip time from a client in Sweden (at 147.214.27.17) to the server 208.67.222.222 (resolver1.opendns.com) in London, UK.*

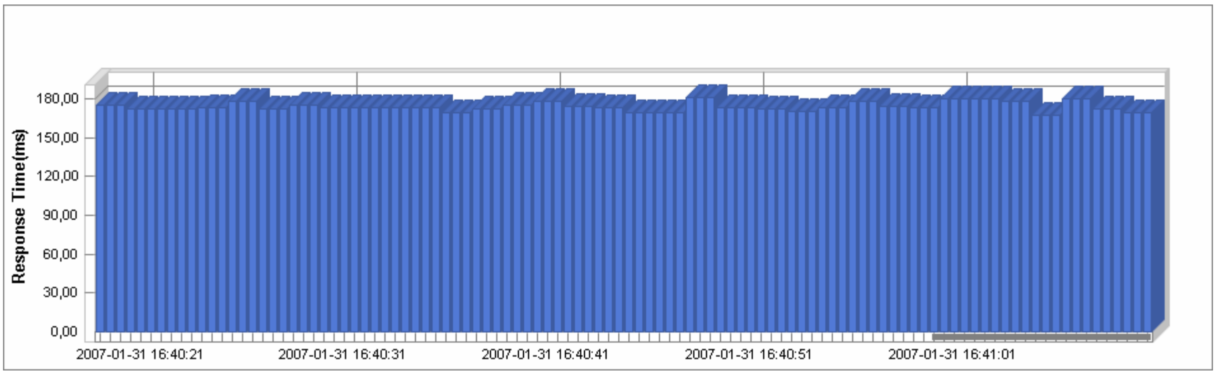

Figure 2: *Round trip time from a client in Sweden (at 147.214.27.17) to the server 66.94.234.13 (w2.rc.vip.scd.yahoo.com) In San Jose, California, USA.*

In general for players within the same country if both players are connected to wired networks they will experience much lower latency than when one or more is attached to a wireless mobile network. This is due to that the mobile station (MS) needs rather significant time to process the packets, that the transmission time across the channel is often slow and in the case of GPRS it also takes time to be assigned radio channel resources. Also cell switches are common in mobile networks, if the user is moving around with their terminal. These can frequently occur if the gamer for example is riding on a bus or a train and plays. When they occur the mobile terminal needs considerable time to setup a link to the new station [6]. Depending upon the underlying protocols which are used, there may even be some traffic which is lost – in some cases enough to make the game unsatisfactory.

Of course latency not only depends on the client/server geographic positions. Latency is also caused by the following other three factors.

- **Inadequate network performance**: Packets either gets dropped along the way, as the network can not handle the offered load of packets, or the packets due to congestion takes a longer parth from source to destination.
- **Inadequate server processing power**: If the terminal wants information from the server that is not cached, or that the server needs to use a lot of processing power to handle the request, there will be processing delay. This affects many multiplayer games during the busiest hours when the server has to deal many different client requests.
- **Inadequate terminal processing power**: A slow terminal may have difficulties processing all the received data and thereby create a delay. This can sometimes easily be solved by shutting down other programs running on the terminal or by upgrading to a more powerful terminal or if it is possible upgrading the hardware of the terminal.

Due to different loads on the network it gets (just as a road network) congested at different times during the day. Therefore you can not expect that the round trip time is time *independent*. Figure 3 shows the results of running a program called FREEping, ping the same two servers utilized in Figure 1 and Figure 2 over a period of 15.5 hours. As the reader can see in Figure 3 the response time of the London server varied greatly between 22 ms and 102 ms over this time period. The average response time was however 44 ms. This time dependent latency is something a game developer should be aware of, because gamers will always play, irregardless of the time, but more will be playing at the peak playing times. An assumption that peak playing times are in the evening and during weekends can easily be tested, by pinging a gaming server during different hours.

These time and geographically variances of latency in the wired network is, however, nothing that a mobile game developer has to worry much about. This because, as described in chapter 4, the greatest part of the latency in a cellular network today is over the wireless link. Nevertheless with newer technologies the wireless link will cause less and less delay of the overall latency, thus the latency variance of the wired network will just as in modern cable connections, influence the overall latency more.

Host	Description	Delay (ms)	Sent	% OK	Min.resp(ms)	Avg.resp(ms)	Max.resp(ms)
208.67.222.222	London	34	11262	99.51	33	44	102
66.94.234.13	USA	176	11380	99.82	169	175	207

Figure 3: *Round trip time from a client in Sweden (at 147.214.27.17) to the servers 66.94.234.13 in the USA.and 208.67.222.222 in the UK. Measurements were made over 15.5 hours, one new ping sent every 5 seconds, with a packet size of 72 bytes.*

Note however, that many routers treat ICMP packets (that are the particular type of packet sent by these ping programs) differently than other packets, hence these results may not be representative of the round trip time for a UDP packet (such as might be used in a game).

2.1.2. Bandwidth

Generally when we speak of bandwidth we talk about the useful frequency band which a channel can transmit. A high bandwidth leads to a high throughput. Usually the throughput is measured bytes (or bits) per second. However, we often talk about the bandwidth of a particular channel in terms of the maximum potential throughput. Necessary throughputs for some common mobile applications are given in Table 1.

Table 1: *Throughput for some common mobile applications*

Mobile applications	Typical throughput [29]
Microbrowsing (WAP)	8-32 kbps
MMS	8-64 kbps
Multi user gaming	2-100 kbps
Video telephony, e-mail and audio streaming	32-384 kbps

2.1.3. Jitter

Jitter, the delay variance, is almost as important for the gaming quality as latency [7]. It can be caused by routers queuing packets, due to congestion or prioritizing traffic. At high loads packets may take an alternative route. This causes packets to arrive at the destination with differing delays. The source can't avoid jitter - it is going to happen or not, but the source can enable the receiver to deal with it by timestamping the packets it sends, thus allowing the receiver to deal with them in a suitable fashion. Then when they arrive to the end node they can be stored in a “de-jitter” buffer until they are delivered in the right sequence and with appropriate interarrival spacing based on the timestamps. This will increase the end-to-end delay for some packets, and must be seen as a trade-off between delay and increased packet loss. The later packet loss occurs in the receiving terminal, when the packet arrives too late to be useful – it will be discarded. This is the worst form of packet loss, since it has actually been delivered and used resources all along the path.

2.1.4. Protocols

The internet is mainly based on two protocols TCP and UDP. These two protocols have both advantages and disadvantages for multiplayer game development purposes.

2.1.4.1. TCP

TCP is a reliable byte stream protocol. When a packet is lost, it will be retransmitted. To achieve this every packet has to be acknowledged, and this of course consumes bandwidth. If there is traffic flowing in the other direction, then this acknowledgement can be piggy-backed on the outgoing traffic so the over head of acknowledging a sequential stream of bytes is only 4 bytes. Due to the fact that TCP is connection orientated it has more overhead. The header size of TCP is at least 40 bytes while the header size of UDP 20 bytes [8]. However, header compression is generally very effective and can reduce this down to one or two bytes in most cases.

The game’s dependents on the performance of higher-layer transport protocols such as TCP, which affects the perception of the technology by the end user. This is due to TCP’s limitations in a wireless network, where the protocol influences the traffic strongly due to its slow-start and congestion-avoidance mechanism [9]. TCP assumes that a packet loss occurrs due to congestion, which usually is the cause in a wired network; where congestion is avoided by lowering the sending rate. In a wireless environment, however, non-congestion packet loss occurs at a much greater rate than in a wired network. Therefore TCP unnecessary decreases its throughput when implemented in a mobile network if packet losses occur. The slow-start mechanism of TCP influences the end user experience of web browsing to a much greater extent than the radio link [9]. This also affects gaming which produces similar traffic as web browsing.

2.1.4.2. UDP

UDP is a datagram protocol, and if packets are lost they are simply lost. This, however, consumes less bandwidth for the same amount of data compared with TCP. Of course this also reduces latency due to the fact that there is no waiting for retransmission. Due to the fact that the game state traffic is more time sensitive, but relative tolerant of losses (in this it is similar to voice over IP traffic), UDP is preferred by game developers. For example, if one update is missed due to packet loss there will be a new update slightly later that the terminal will be able to use. However, TCP can be used in parallel with UDP. For example, in a game application where packet delivery is important such as player-to-player chat; hence TCP may be used for the chat, while the game state is transmitted with UDP [7].

2.1.5. Packet loss

If packets never arrive at their destination they are considered lost. This loss of packets can be caused by many factors, including network saturation, degradation of the signal through the network medium, faulty network hardware, or faulty packets being rejected by nodes. Packet loss can significantly degrade the quality of a gaming experience. For example in the online multiplayer game Halo players already begin to feel a significant decrease in quality and their performance if more than 2% of the packets are dropped [10]. Packet drops due to network saturation can be caused by different factors, such as a bad route or lack of sufficient throughput on a bottleneck link or router [11]. Therefore it is crucial for a mobile game developer to know if players will be able to have sufficient throughput, when a when a game is being developed for mobile terminals.

2.2. Latency avoidance

Latency can be seen as a network parameter that is very crucial for online multiplayer gaming (see also chapter 3). Therefore, ideally for highly interactive games you use a low latency network. Unfortunately, current cellular networks have relative high latency, but luckily there are several methods to avoid or more speaking hide latency in order to achieve better gaming quality. Some of these techniques will be described below.

2.2.1 Dead reckoning

When latency occurs the client will suffer from missing game data. Dead reckoning (DR) is a way to estimate this missing data by taking into account recent positions, velocities, and accelerations of the game objects. This hides the negative effects produced by the network or server latencies.

A player sends his or her position, as well as velocity and acceleration to the other players in the game sessions. This is done through a DR vector, that contains the player's position and movement in the x, y, and z coordinates (in addition, it might also include pitch, roll, and yaw – along with their first and second derivatives in a game involving aircraft, space craft, or other vehicle moving in three dimensional space). When this vector is received by the other players, they can predict the sender's future movement assuming that the velocity and the acceleration are unchanged. This prediction is done until the next vector is received, which will give the receiver an updated state for this sender. If there is any difference in the predicted state and the actual state in the updated vector (threshold) the sender's real position will be updated at the receiver. Of course you want the threshold to be as low as possible so that the gaming experience is as smooth as possible without any glitches. This is achieved if the paths that the players take in the game are easy to predict or if we update the players frequently with new vectors. The whole idea of dead reckoning, is to predict the client's position rather than frequently update each peer with new information, thus reducing the necessary bandwidth [12].

To deal with the question of how often the other players should be updated you can implement an adaptive dead reckoning algorithm. Thus different threshold levels will be used in different game states, thus varying the rate at which updates are set – so as to maintain a good gaming experience while using only as much bandwidth as is actually needed to do so.

If two entities in the game are on a large distance from each other, they do not need to update each other as frequently as if they were at a close distance. The update rate can be predicted in the following way. Every entity in a game has an area of interest (AOI) and a sensitive region (SR). These are defined as areas inside a circle, whose radius is defined according to the entity [13]. By using these areas four acceptable threshold levels can be defined, where level 1 accepts the lowest and level 4 the largest. These are illustrated in Figure 4.

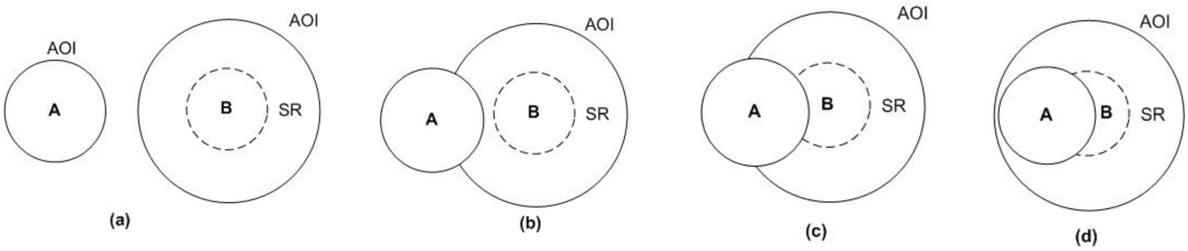

Figure 4: *(a) Level 4: no overlap of AOI, (b) Lavel 3: A overlap with another enties, (c) Level 2: A is in another entity's AOI, (d) Level 1: A is in another enity's SR*

The adaptation of the dead reckoning algorithm used can also be adapted to the entities' movements. As an example, if the simulated entity moves in a straight line its further movement should be easy to calculate, while if it moves in a zig zag pattern it will be hard to predict its position (after all this is generally why a player would not move in a straight line!) [13]. To keep things easy a number of different algorithms can be used to predict the player's movement and to decide when to emit updates.

Using dead reckoning the server can utilize a longer update period. Also latency which is caused by the network will be hidden from the gamer's perception. Thanks to dead reckoning and its ability to smooth over gaps caused by packet loss, game developers have, for game state information transfer, a tool to utilize the more insecure and unreliable UDP protocol to a greater extend than TCP [14]. The result is that you have lower delay together with smooth game play.

Unfortunately dead reckoning does not come for free. Its implementation means that all the clients have to run an algorithm calculating the vectors while running a game [14]. These calculations consume computational (and battery) resources and therefore in the case of mobile terminals, they can slow down the client significantly. Another disadvantage is if all clients' play is unpredictable, then dead reckoning will not help. Dead reckoning is only useful when it is possible to predict a probable path for the game objects, but if their predicted movement do not coincide with the client's actual movement, then the prediction wasted resources and perhaps leads to burst of updates being needed.

2.2.2. Heterogeneous network environments

To avoid the negative aspects of dead reckoning, especially in the case of mobile gaming, it is better to utilize a server solution that directly deals with the latency. In the case of mobile gaming there is a high probability that the latency is heterogeneous. This means that some clients have a very low latency when communicating with the server, while others do not. To address latency problems one effective method is to adapt the server's update time T according to this heterogeneous latency. Due to the limited hardware resources in handheld terminals, this method is a rather easy way to deal with latency, instead of applying very complex prediction algorithm which must run on the client.

There is literature that confirms that a periodically server update (T) is the best solution [15].

> "This information is periodically sent from all of the clients to the server. The server then takes this information and performs a periodic broadcast to each client, effectively distributing the global state of the game."

The protocol works as following.

- The game status in the server gets updated every period T.
- During that time clients send their update information to the server. Packets that arrives too late, will however be considered to be lost.
- The clients are updated by the server. Upon receiving information from the server, the clients send new information to the server.

- When these packets arrive at the server they have to wait until the next update. This waiting time is called the client's idle time.

As the reader easily can understand, we can minimize the unnecessary packet loss by increasing T. In that case the server would give all clients sufficient time to update. Nevertheless this will unfortunately give the clients with a low latency a huge idle time, thus an implementation like this is not a good solution in a heterogeneous network environment. To overcome this problem, the idea is to let the server update the clients every period T, yet accept packets arriving later than T. In other words, the server will accept all packets received either in the interval [0..T] or [0..2T]. This reduces packet loss, while the fast clients do not need to suffer from long idle periods.

To find out ideal protocol and update period T studies have been made where three clients with different latency participated in a game session. By considering loss probability, average idle time, and the tail probability of maximal idle time the study came to the conclusion that the game session performed better with the modified server protocol. In the experiment the clients respectively had 20 ms, 40 ms, and 60 ms latencies and with packet sizes of 100 byte, leads to an idle T of between 110-150 ms [16]. This result shows that using 40-50 ms server update, which is common today, is far from the ideal case.

Of course this method does not eliminate the latency problem itself. However, it shows that even if the network is only slightly heterogeneous (latency differing by 20-60ms), the game developer has a lot to gain with respect to network performance if he or she modifies the server update period correctly. This is especially crucial if the developer has scarce network resources such as may be the case in a mobile network.

2.3. Game models

There are two main ways that traffic can be routed in a gaming network, through a peer-to-peer model or through a client-server model. Below we describe the different principles and their advantages and disadvantages.

2.3.1. Peer-to-peer

A peer-to-peer connection is only based on clients, who all are connected to each other. The game status is updated individually at every client. This solution is perfect for the game provider because there is no need to invest in a server. Another advantage is that the network is very stable. If one client goes down, the remaining client still make up a network and continue to send information to each other.

The drawbacks are related to security. All the clients have all the game information, so it will be much easier for one gamer to cheat by hacking the game information in his terminal. Also the implementation is rather complicated in comparison to the other models. To avoid divergence in game state due to delays and other factors, synchronization has to occur between clients to avoid divergence. Another disadvantage is that the network traffic generated increases exponentially and a player can easily run out of bandwidth. This is a really important issue to have in mind when developing mobile network games, due to the limited throughput in today's mobile networks. To understand why the network traffic rises exponentially the reader should think of the numbers of open connection. With three clients, each connected to the other two and sharing their game states, then the network needs 3*2=6 connections. If we now instead have four clients, they all each connect with three others giving us 4*3=12 connections. With five clients 5*4=20 connections are needed, thus we have an exponential increase. Note however, the numbers of connections to and from each client only increase linearly. Peer-to-peer games used to be limited to 8 participants, however, thanks to clever data compression schemes some allows up to 32 clients simultaneously [17].

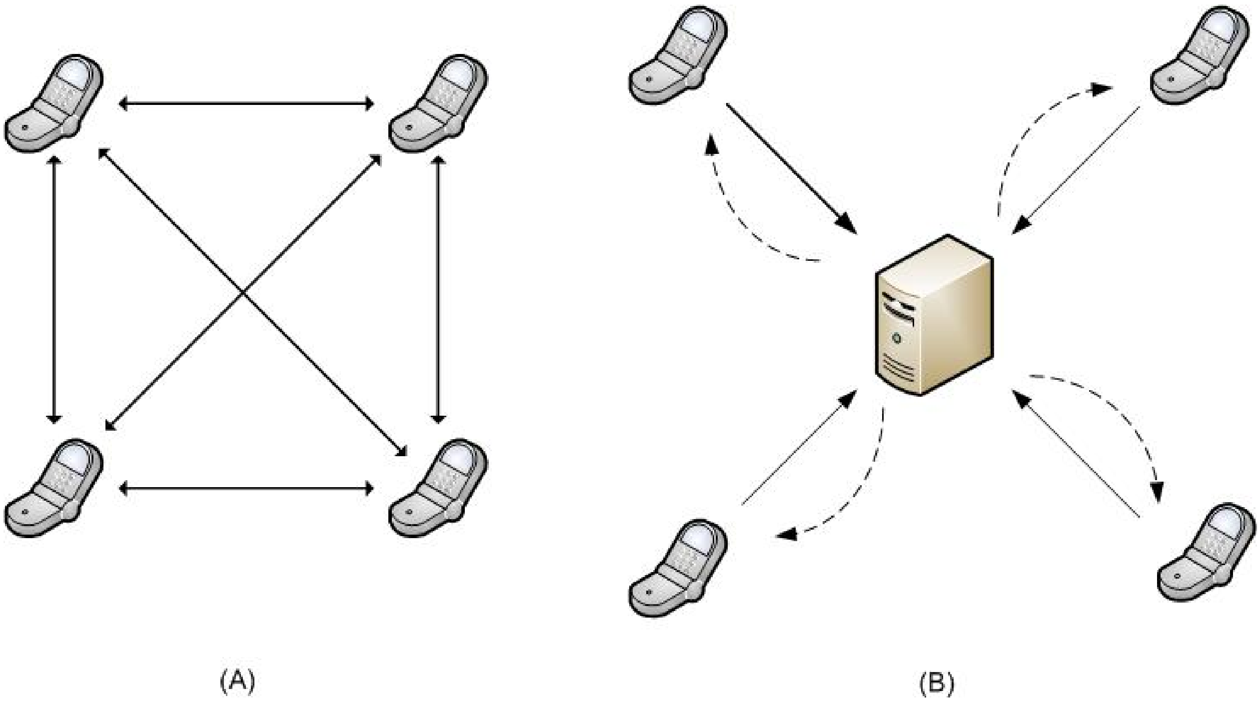

Figure 5: *(a) a peer-to-peer architecture (b) a client server architecture. Arrows indicate "Event message" while dotted arrows indicate "Game update Message"*

The advantage of peer-to-peer is of course that the player uses his own hardware and bandwidth to create the specific network, which leads to lower costs for the game publisher. The big advantage is that because all clients talk directly with all others, thus all of nodes have the complete game state. Therefore it is rather easy to extract secrete game state information (such as information outside the normal range of your observations), by simply hacking your own client in the same game [17].

2.3.2 Client-server

A client-server connection is based on a server. This server stores and processes all the game data it receives from all the connected clients. Then it only updates those clients with the data they need, thus every client receives a unique update. The limited information that the server sends to its clients is good from a traffic point of view and leads to lower latency. Also from a coding point of view this model is preferable, because little code needs to be added to support this sharing of state information, and it can easily be separated from the game code. These are the reasons why most modern multiplayer game implements this solution [7], and for mobile games it is essential – IFF there are many players or there is a lot of changed state on average per player. One drawback however, is that bottlenecks at the server still can occur, due to high load from many clients (who are sharing this machine because of the allocations of resources to each of the instances of the game). This architecture has another big disadvantage; someone has to pay for the server and this cost can be substantial if the game is to support many clients and to maintain the latency below some threshold.

The client server model is preferable from a (reduction of) cheating point of view [17]. Every client has a scope in the game world, stored in the server. Only the server knows the complete game state and the different client's scopes. The server individually updates each client only within their scope. Therefore the data received by each client contains only information about his or her scope, not the other clients, until the scopes from two different clients interfere with each other in the server's game world. If you want to cheat, you usually want to know information about your opponent *before* your opponent can learn about your state; however, as this information lies in other client's scope – until you reach the interaction distance you can not know the state of the opponent. However, because you do not receive that information, you would have to hack the game's server to get this information; this is likely to be detected. If you can't directly eves drop another client's traffic, this makes it difficult to have global knowledge of the game state. In a peer-to-peer architecture because all game states is distributed to all clients, this of course increases the potential for cheating, since you only need to hack your own client [17]. Therefore a server-client model is better, if the developer wishes to avoid players cheating in this game.

2.3.3. Network server

To avoid bottlenecks a network server based architecture is ideal for Massively multiplayer online role-playing game (MMORPG) games which is a game genre with many participants connecting to the same server. The clients connect to one or more servers, which are in turn inter-connected with each other through a local network. The local network enables the servers to exchange a huge amount of data very quickly [7]. This model allows many clients to be connected to a server without causing saturation of a single server.

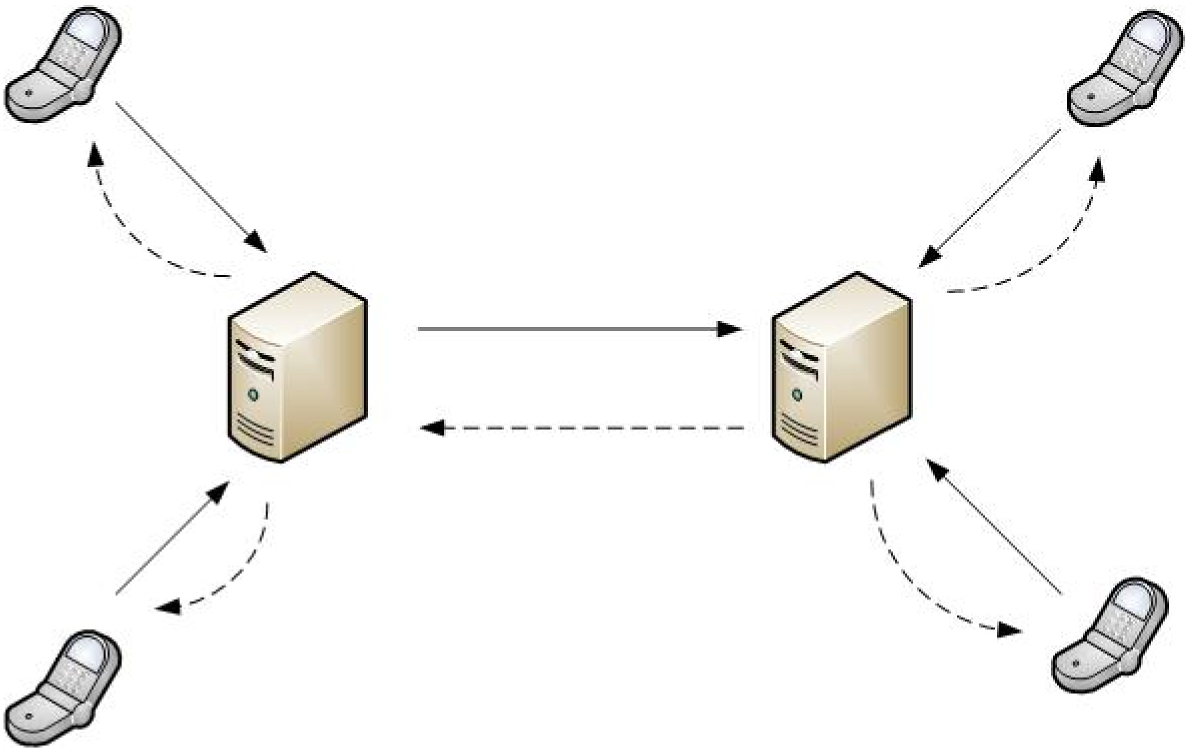

Figure 6: *A network server architecture. Arrows indicate "Event message" while dotted arrows indicate "Game update Message".*

2.3.4. Network architectures for mobile gaming

Ignoring local Bluetooth connections, most mobile multiplayer games are played using a client-server architecture. This due to the fact that today's mobile telephones do not have a public IP-address and therefore direct client-to-client communication is not supported. Therefore, if a game developer wants to create a mobile multiplayer game, he or she or the mobile operator has to host a server [(i)]. A server will have a cost for its hardware, software, bandwidth, and hosting. Additionally, from a developer's point of view, a multiplayer game increases the development cost. Not only must a client application be created, but also a server application. This usually involves two projects, hence it is two implementations with different requirements. The server side coding may need to be done in Java 2 Enterprise Edition (J2EE) which is beyond most single player game developer's knowledge [18]. Of course high revenue would be a solution to these cost problems. However, most of operators charge per kilobyte of data sent, without sharing any of these revenues to the game

[i] *(An alternative solution the reader is referred to the thesis by Gustav Söderström and his subsequent company Kenet works - for a solution to the problem of phones not having publig IP addresses.)*

developers. As a glimmer of hope, we see that in Sweden some mobile operators have started to introduce flatrate.

Bluetooth is a radio technology which was designed to connect clients within a distance of 10 meters and it posses the great advantage of having only 20-50ms of latency for devices which are part of a piconet [18]. Due to this, local networks can be set up to play really demanding multiplayer games, such as FPS. However, the actual architecture of a Bluetooth piconet is a client-server model. One device has to act as a “master”, while the others (a maximum of seven) called “slaves” connect to this device. The master can route data communication, hence communication between two slaves requires two transfers: one to master and one to slave – combined with additional processing and latency in the master the effective delay between two slaves is 40-100 ms. Thanks to this technology small local multiplayer games can be set up without any cost because the bandwidth is free, but the latency for slave communication may be high.

If you have your friend’s telephone number you can of course connect to him directly, then you can initiate a two person game session without any gaming server, just as with an old fashioned PC-modem [19]. However, games with more than two players need to have a “virtual lobby” where the gamers meet online before joining a game session [20]. This lobby introduces an extra cost to the gaming. Therefore companies have emerged specialized in producing and providing the middleware for gaming, including managing the virtual lobby as the reference [21] states.

“*Terraplay offers network solutions aimed at providing high quality, and commercial real-time gaming service, which targets both existing and future mobile and fixed networks These systems intend to enable new types of multi-player games and gaming experiences by focusing on network technology. These systems intend to enable the operator to launch and run high performance commercial on-line gaming services both resource and cost effectively. They try to enable a wide range of business models based on the systems’ features.*”

3. Technology

This chapter gives an overview of the most important mobile network technologies on the market today. Via these brief summaries, the reader should gain insight into the different mobile technologies, how they perform especially in terms of throughput and latency and their coverage area.

In the last decade mobile data transfer has undergone an enormous evolution in terms of network performance. Building on the great success of GSM/GPRS, which today is the most widespread wire area data transfer technology, today's UMTS/HSDPA and its evolution beyond 3G, cellular technologies will soon compete with other wireless IP technologies, such as WLAN in terms of throughput and latency. This together with the great advantage of greater mobility gives many mobile operators hope that in the near future most of their customers wanting an internet connection will chose their HSDPA services even for their home broadband access [22]. It seems that this evolution is held back primarily because of the large investments that are needed. This together with peoples reluctance to use advanced mobile services or to pay for them, are the main reasons why GPRS is still used to such a large extent.

3.1. GPRS/EDGE

GPRS, which added packet data services to the GSM network, is with its 2 billion subscribers world-wide without any doubt the most widespread wide area wireless data service available [29] With a throughput around 40kbps it can support applications like SMS, WAP, and email. GPRS offers packet-based IP communication and builds upon the existing GSM technology and networks.

EDGE is an enhancement to the GSM/GPRS network, where the architecture is unchanged. The change is limited to the radio interface. EDGE is considered a third generation (3G) cellular technology. By using a different modulation scheme during the timeslot allocated for GPRS the throughput can be increased. This enhances the throughput to more than twice that of GPRS. Also latency of EDGE is of a higher performance than the one of GPRS.

3.1.1. Throughput

The throughput of GPRS is determined by many factors. For example the distance to the base station and the coding scheme that is selected. There were initially 4 different coding schemes implemented in GPRS used to protect the data transfer from error. Today there exist newer coding schemes, as described below. Each of these coding schemes, other than scheme number 4 utilizes an error correcting code to reduce lost packets and thereby enhance the reliability at the cost of some throughput. The four schemes implemented in GPRS are numbered CS-1, CS-2, CS-3, and CS-4. CS-1 offers the most error correction and is therefore used to send information to the clients at the greatest distance from the bas station, while CS-4 is used for mobile terminal at close range (see Figure 7). This variation of error coding in a mobile cell is done because you can expect more transmission errors when you send a signal over long distance. Due to this change of coding schemes a mobile user can experience variation in throughput as a function of their position relative to their current service base station, which also will affect the user's gaming experience when throughput is crucial to this game.

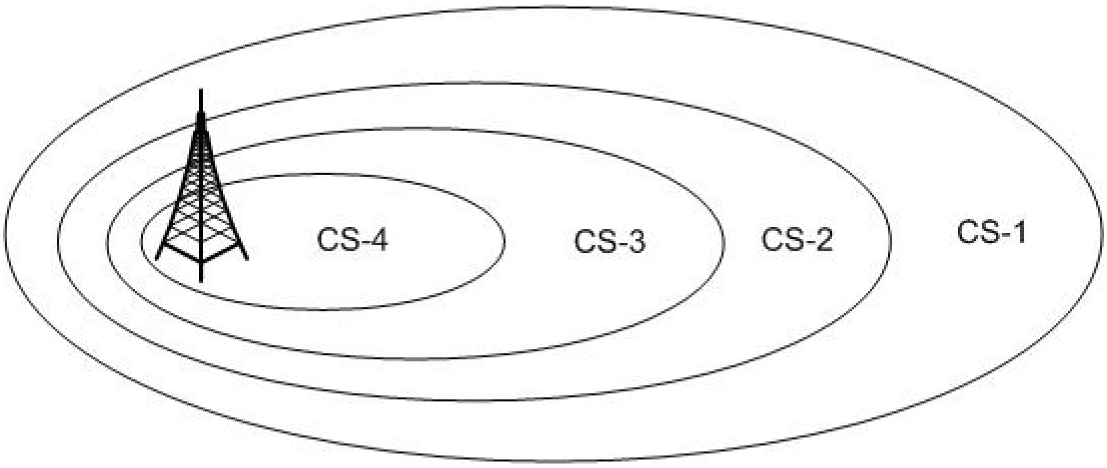

Figure 7: *Variation of coding schemes in a mobile cell*

With coding schemes 1 and 2 each data timeslot can deliver 10 kbps, and with 4 of these timeslots in one frame (i.e., 8 time slots constitute a frame) the network can achieve a theoretical download speed of 40 kbps. With coding schemes 3 and 4 the allocation between the data and the error correction code is better, and with coding scheme 4 every timeslot can achieve up to 20 kbps [23]. The higher throughput of coding scheme 4 is due to it not having any error correction at all, and therefore it adds least overhead to the information sent.

However, without error control a lot can go wrong with the information during its transfer through the air. Therefore the base stations only utilize this scheme for clients that are close to it.

3.1.2. Coding schemes

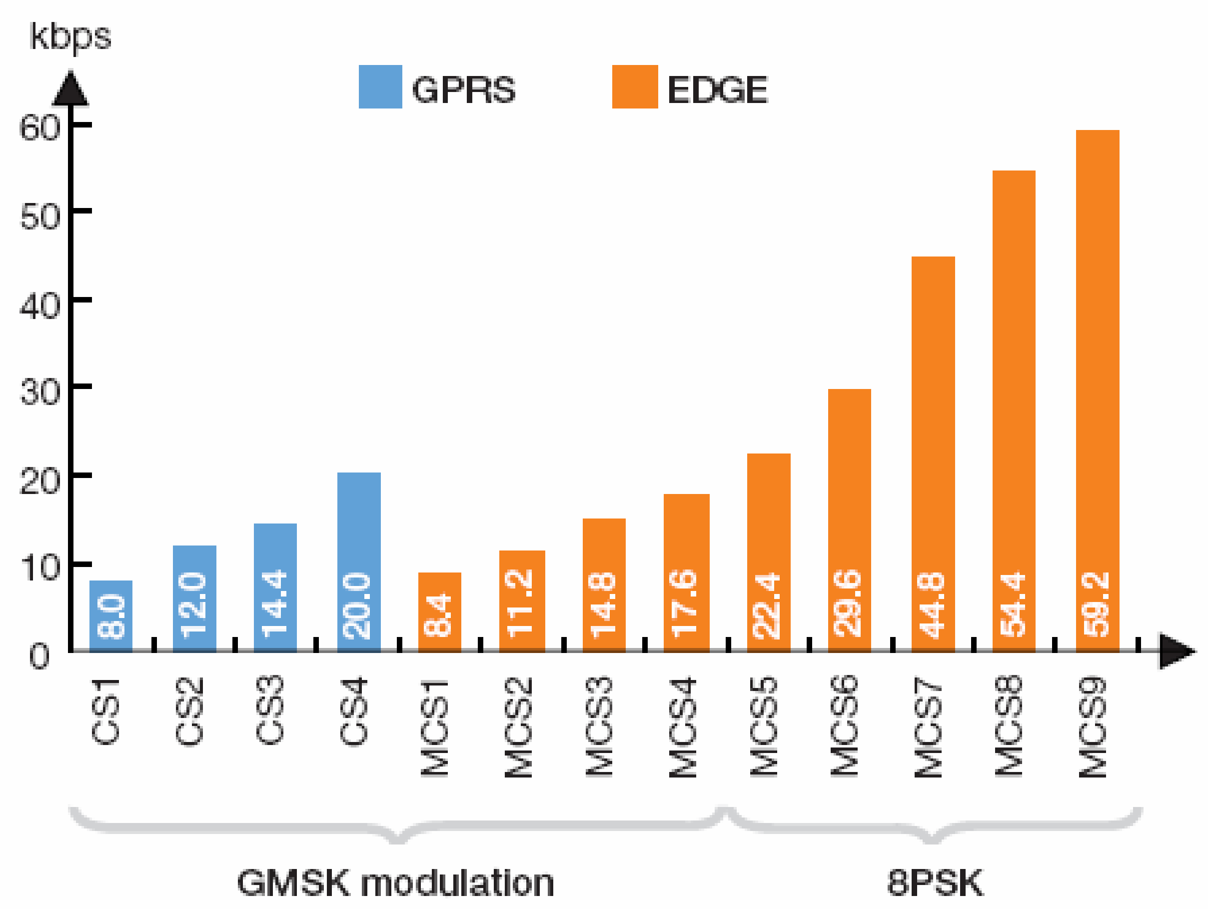

Figure 8: *Throughputs of different coding scheme. This figure appears with the permission of the copyright owner [24]*

Enhanced GPRS (EGPRS) is built on top of GPRS. This is done by introducing some more complex coding schemes. As mentioned earlier GPRS utilizes 4 different coding schemes while EGPRS has 9, labelled MCS-1 through MCS-9. Coding schemes MCS-5 to MCS-9 uses the more advanced octagonal phase shift keying (8-PSK) channel modulation, while MCS-1 to MCS-4 uses GMSK (the traditional GSM modulation). As mentioned earlier, greater coding gives greater error correction but lower throughput (see Figure 8). As the reader can see in the figure, the higher the order of the coding scheme is, the lesser error correcting capability it has and the greater throughput it delivers. The advantage of EDGE is that if a packet fails to be transmitted, GPRS would simply ignore it, while EDGE will try to send it again, this time using a coding scheme with more error correction so that hopefully the retransmission will be successful [24] hence also producing less packet loss which is good in a gaming point of view.

As some EDGE coding schemes give a huge gain compared with GPRS, the average throughput gain of up to 3,6 times is achieved with this more modern technology [29].

3.1.3. Latency

Latency in GPRS is not caused by a single source. The overall latency is caused by:

- The Mobile Station (MS)
- Radio resource procedures
- The effective data throughput
- The GPRS core network nodes.

The mobile station needs time to process the packets. This is a hardware issue and depends greatly on what architecture the platform developer utilizes for the MS. While, this isn't the greatest source of delay, it usually adds around 100 ms [25] to the overall delay. The greatest portion of delay is caused by the radio resource procedures, which mainly concerns the time it takes the MS to find a radio resource called a Temporary Block Flow (TBF). If there is a TBF ready to use for the MS, then there is only minor latency. However, if there is no TBF free, then the terminal first has to exchange information with the network, which may then grants the MS the radio resources it requests. The effective data throughput is the actual time it takes for the packets to be transferred from the MS to the base station. Most of this latency is caused by the over-air-transfer. This latency will linearly dependent on how many timeslots were required. The total one-way latency for GPRS is around 800-1000ms.

EDGE has lower latency than GPRS. This is done by reducing the transmission time interval (TTI), which is the time it takes for the radio packets to travel from the core network to the mobile terminal. TTI can be reduced by sending the radio blocks using more than one timeslot, or reducing the size of them. With EDGE you can theoretically achieve 150ms RTT, but a more realistic RTT is between 300-600ms [26].

3.1.4. Conclusions

Due to the high latency of GPRS networks it can barely support even none-demanding TBS games. This is despite the fact that the throughput is sufficient. Another drawback of GPRS is that it does not support soft handovers, a technology that will be described in section 3.2, and hence a lot of packet loss can occur. Packet loss in a network is never good from a gaming point of view. Also the throughput of GPRS is not sufficient for all kinds of games.

The latency of EDGE is clearly sufficient to play turned based games, such as Worms or Civilization, but may be a bit too high for the more demanding game genre real time strategy. The throughput also seams to tangent the demand of multiplayer gaming. However, the technology is also not so widespread, due to the fact that the carriers prefer to invest in “real” 3G technologies rather than upgrading their GPRS technology. Due to these reasons this thesis will neither consider GPRS nor EDGE technology any further.

3.2. UMTS/WCDMA

UMTS is a “real” 3G technology which, in comparison to EDGE, offers a significantly greater data transfer and allows simultaneous voice/data communication, which is great in a cellular online gaming point of view, as the gaming session will not be interrupted by an incoming call. The technology is based, like the internet, on packet switching. This gives the possibility to be connected to the network all the time, while you only need to pay for the packets transferred [27]. Because most UMTS terminals are also compatible with GSM, it’s rather easy for the 3G operators using UMTS technology to offer their subscribers extensive coverage (assuming that they have a roaming agreement with a GSM operator or operate their own GSM network). A drawback of these terminals is that they are rather big, heavy, and battery consuming, but the bigger screen gives an advantage towards gaming. Also the terminals have evolves a lot the last year, so now you can find rather slim non battery consuming terminals. Another disadvantage of the technology lies within the network technology that needs a rather wide bandwidth, which was expensive to license in many countries [28]. Never the less UMTS has become a widespread technology and today (beginning of 2007) is offered by well over 100 commercial operators with around 103 million subscribers [29]. The operators hope that their subscribers will find all the new services UMTS can deliver to be interesting enough to pay for them. All these new services can be delivered thanks to the throughput of 320 Kbps and the latency of 250 ms [29] that UMTS can offer. If inspecting Table 1 the reader understands that UMTS delivers more than sufficient throughput in a multiplayer gaming purpose.

3.2.1. WCDMA

UMTS uses a protocol called Wideband Code Division Multiple Access (WCDMA), which is built on the Code Division Multiple Access signaling method (CDMA) [30].

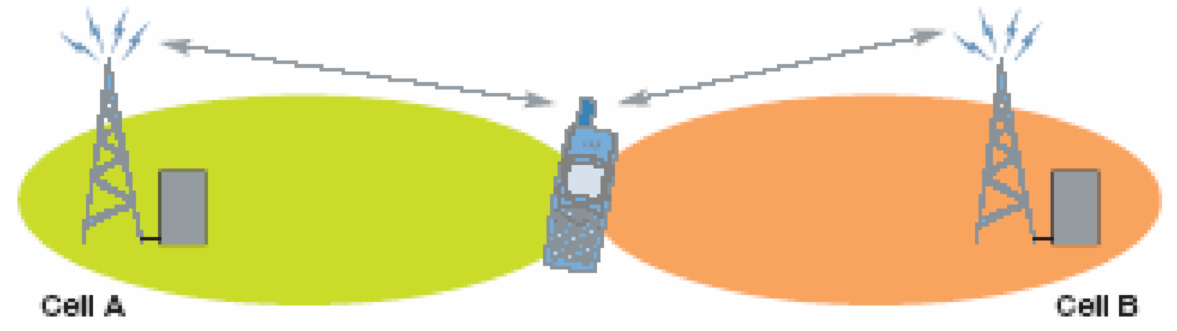

Figure 9: *Idea of soft handover. This figure appears with the permission of the copyright owner [30]*

An advantage of WCDMA technology is it can offer soft handover between cells. This means that close to a cell's border, the terminal starts to be in contact with the new base station while simultaneously maintaining a link with the old one (as illustrated in Figure 9). Then, gradually as the terminal gets closer and closer to the cell border more and more traffic is given to the new base station, until it handles all of the terminal's traffic. This leads to fewer dropped phone calls, and will also improve the handheld terminal's data transfer in terms of reducing packet losses as the terminal switches cells [19]. Also less bursty traffic can be expected, which is better in a gaming point of view. During hard handover there are short periods when the terminal does not have any contact with the base station, and hence can not transfer any data which lead to bursty traffic and perhaps losses.

3.3.2. Admission- and congestion control

WCDMA operators have the possibility to have admission control which is a powerful tool to attract gamers. When a new user tries to connect, the network calculates how many users are already connected and what their traffic demands are, then it estimates how much extra load a new user will put on the network. Depending on the operator's set quality level or the subscriber’s demand for services, the user will be accepted or declined permission to connect to the network. Also a bounded latency, packet drop, and throughput can be guaranteed [19].

Even though admission control is reliable it can fail sometimes. In case of congestion there are four steps taken to assure the quality level of the connected users:

1. The network reduces the traffic of non-real time applications
2. The network moves some subscribers to another, less loaded frequency.
3. The network moves some subscribers to the GSM network.
4. The network terminates some subscriber’s connections, to assure the connection quality of the remaining users.

With admission and congestion control operators have a means to attract gamers. They can sell more expensive gaming subscriptions and through congestion control assure that these users get the quality of service they demand - of course this may mean that voice calls have to be dropped, but the operator has to price the service accordingly.

3.3.3. UMTS real performance

Due to the good latency performance, the soft handover with the lower packet drop rate and the carrier’s ability to individualize their subscriptions to the clients it is interesting to further consider this technology for gaming. Before doing that, however, it would be interesting to see how UMTS/WCDMA really performs. Of course the performances mentioned in the white papers are ideal laboratory results. However, when using such a device in commercial networks the user has to share resources with others, thus the performance tends to decline. It is this actual performance that is interesting to know about, because this is what gamers will perceive. Therefore it would be interesting to conduct an experiment that measures the performance in a commercial UMTS network.

The carrier used in this experiment was the Swedish Tre, together with an Option WCDMA datacard that was connected to a laptop. The laptop could connect via the cellular network; by using different ping programs network statistics were collected. The first measurement was done stationary on Södermalm in Stockholm. To the battlenet server (213.248.66.170), which is a gaming server located in Sweden, the average ping, is around 123 ms, but peaks at almost a half second can be observed (Figure 10 (a)). Interesting to note is that the server's geographically distance does not contribute significantly to the overall UMTS latency. The average latency for the London server over UMTS/WCDMA is around 40 ms longer than the battlenet server in Sweden. That difference in percent was much less than measured over the fixed network (Figure 11). Because of these characteristics it is not as important for the game developers to consider where (geographically) to install the server, when develop a mobile game unlike when developing a game for fixed network connection. This is of course an advantage because the servers can be concentrated in fewer locations.

The second measurement was done on a bus. As the measurement equipment slowly moves in a local city bus travelling at a speed of approximately 30km/h, UMTS performance is shown in Figure 10 (b). To the battlenet server (213.248.66.170) the average ping is now around 182 ms, but peaks at more than a second can be observed. This latency is significant more, and though the measurements where done in the same geographical area during almost the same time period of the day, the results must bee seen as statistically secure and can not bee due to changes in the overall network performance. What is good to see from the bus measurements is that the packet loss is 0%. This is likely achieved thanks to the soft handover mechanism. Also due to the low rate at which the pings are made - thus the probability of a handoff at the time of a ping is very low!

The last measurement was done on a commuter train between T-centralen and Helenelund in Stockholm; a rather central area and suburbs. The train travels significantly faster than a city bus, but in no means in a speed of more than 100km/h. Now to the battlenet server (213.248.66.170) the average ping was around 229 ms, but peaks at more than a second can be observed (Figure 10 (c)). Now, also some packet loss occurred. A value of 4,9% to the battlenet server was measured as an average over the whole trip. This packet loss could be due to the cell switches where so rapid that the soft handover could not handle them fully. What was interesting to see is that the London server suffered from more packet losses. Another interesting thing to observe is that the min response time was less for both servers in comparison to the measurements done on a bus and stationary.

Host	Description	Delay (ms)	Sent	% OK	Min.resp(ms)	Avg.resp(ms)	Max.resp(ms)
208.67.222.222	London	163	128	100.00	150	164	471
213.248.66.170	Battlenet	123	125	100.00	110	123	454

(a)

Host	Description	Delay (ms)	Sent	% OK	Min.resp(ms)	Avg.resp(ms)	Max.resp(ms)
208.67.222.222	London	176	58	100.00	154	233	1339
213.248.66.170	Battlenet	135	58	100.00	116	182	1237

(b)

Host	Description	Delay (ms)	Sent	% OK	Min.resp(ms)	Avg.resp(ms)	Max.resp(ms)
208.67.222.222	London	349	204	92.16	139	298	1041
213.248.66.170	Battlenet	229	204	95.10	99	250	1208

(c)

Figure 10: *(a) Stationary measurement with an average ping of 123ms (b) An average ping of 182ms measured on a city bus traveling max 30 km/h (c) An average ping of 250ms measured on a commuter train (pendeltåg).*

Host	Description	Delay (ms)	Sent	% OK	Min.resp(ms)	Avg.resp(ms)	Max.resp(ms)
208.67.222.222	London	43	88	100.00	43	43	44
213.248.66.170	Battlenet	2	88	100.00	1	1	2

Figure 11: *Server ping with a fixed cable connection. It is easy to see that the geographical position of the server here contributes much more to the effect of latency if calculated in percent.*

As earlier mentioned RTS games are much less demanding than FPS, and only require a theoretically latency of less than 450ms to run perfectly well. Even if we experience a few peaks in the latency, we can consider this to be equivalent to packet loss. Additionally, because RTS games are not greatly affected by packet loss (see chapter 4), and the rather wide geographic coverage of UMTS/WCDMA, this network has been chosen to conduct further surveys on. TBS games have even more relaxed network requirements and therefore it should also be able to run such games on WCDMA/UMTS.

3.4. HSDPA

Using High-Speed Downlink Packet Access (HSDPA) the operators have the chance to give their subscribers an improved end-user experience by adding the latest multimedia services. This is possible because HSDPA offers higher bit rates (averaging 0.8-1.5 Mbps) and lower round trip delay (70 ms), enabling applications such as multi-user gaming [31]. It also gives users the option to surf the internet, where they can access bandwidth consuming sites which deliver video, music, and pictures. All this is achieved at a rather low upgrade cost, given that the operator has an existing UMTS network. The technology also uses the spectrum efficiently which lead to a low data transfer cost [29]. HSDPA is an improvement of UMTS, built upon the implementation of a new WCDMA or TD-CDMA channel. Hence it is relative easy to upgrade an existing UMTS network [32]. HSDPA today is a new technology with to few subscribers to be of any great interest of this thesis. Also the geographical coverage is limited to big cities. Therefore this thesis preferable considers the more widespread UMTS/WCDMA technology that is of less performance. If the games perform well over UMTS/WCDMA they also will run fine over HSDPA.

3.5. WLAN

WLANs are usually used to link two or more computers into a local wireless network. This technology has grown popular, because of the flexibility it gives a laptop or other devices equipped with a WLAN interface. Many cafés and other popular spots, give their customers free or very cheap WLAN access today. Google has even provided a whole suburb of California with free WLAN access. Due to this growing popularity devices such as IP phones and handheld gaming terminals have started to use this technology. The popularity of WLAN has grown so big that it even has become a competing technology to the cellular networks, not only for traditional data transfer, such as video and gaming, but also for voice services. Because WLAN offer much lower latency, greater throughput, and usually a flat rate price –

people don't need to worry how much they use the network nor do they need special applications.

WLAN is built on several IEEE 802.11 standards, where the most commonly used ones are IEE 802.11b and 802.11g. This due to the fact that they use the 2.4 GHz band and offer high throughputs, which is not restricted in most countries [33].

3.5.1. Throughput

IEEE 802.11b was introduced in 1999 and uses Complementary code keying (CCK) which is a variation of the earlier mentioned CDMA. CCK was implemented to achieve better throughput and offers a data rate of 11 Mbps at 30 m in a typical indoor environment and 1 Mbps at 90 m. However, much higher rates at 100m have been observed in voice over WLAN measurements of Juan Carlos Martin Severiano [34]. The 11 Mbps data rates must be considered as theoretical maximum and is in practice reduced to 5-6 Mbps due to protocol overheads [35].

IEEE 802.11g was introduced in 2003 and the hardware used is compatible with the older IEEE b standard. This newer technology uses a modulation scheme called Orthogonal frequency-division multiplexing (OFDM) for theoretical data rates up to 54 Mbps, but with a realistic throughput of 27-30 Mbps [35].

3.5.2. Latency

The latency of the IEEE 802.11 is between 60-400 ms [36]. The high latency occurs when you change cells. The technology does not support soft handover, and therefore the user has to expect a 252 ms gap during [36] an average handover. The cells are also much smaller than in a wide area mobile network, and therefore the handover delay occurs much more frequently – assuming that the user is moving at comparable rates. One conclusion is that cellular networks, with their soft handover, must offer some advantage in order to attract mobile gamers; for example gamers who are on the move by vehicle while playing, and want to play low latency demanding multiplayer games.

3.6 Chapter summary

With regard to throughput and latency, there are great differences among the technologies. In EDGE the throughput is twice that of GPRS; while UMTS is double again that of EDGE; while HSDPA throughput is approximately three times higher than of UMTS. However, network gaming not only demands a lot of throughput, but users also demand low latency. The comparison between all these technologies is summarized in the following chart.

Table 2: *Latency and throughput for different mobile technologies.*

	Peak download rate	Average download rate	Upload rate	Latency
GPRS	115 kbps	30-40 kbps	14,4 kbps	800-1000 ms
EDGE	473 kbps	100-130 kbps	80 kbps	300-600 ms
UMTS/WCDMA	2 Mbps	220-320 kbps	64 kbps	200-250 ms
HSDPA	14 Mbps	550-1100 kbps	384 kbps	100–150 ms
WLAN	54 Mbps	27-30 Mbps	27-30 Mbps	60-400 ms

Due to UMTS/WCDMA rather good performance and that the technology today is widely spread it will be the technology that this thesis further will consider.

4. Real time multiplayer games

Computer games have always been fascinating thing for many individuals. They can be played in a single player mode and sometimes in a multiplayer mode on different media platforms such as Nintendo, Playstation, PC. There are different ways that multiplayer version of game can be played. In old Nintendo games, as an example, two handheld joysticks were connected to the same machine and both players saw the action on the same TV-screen. Nowadays each player usually has their own terminal which is connected to other players through the internet. Although PC based computer games have existed for several decades it is only recently, with the high penetration of broadband connections that real multiplayer games have grown so popular.

This chapter introduces the reader to different multiplayer game genres, and then focuses on some real time multiplayer games from the different genres. The specific games will be the focus of the modelling in the rest of the thesis.

4.1. Genre

Real time multiplayer games can be divided into many genres. I will include the most popular ones, the basic ideas behind them, and their theoretic latency requirements. Thus the reader can gain better insight into group of real time multiplayer games and understand their different latency requirements with respect to the network.

4.1.1. TBS

Turned Based Strategy (TBS) games are best compared to regular board games, such as chess or monopoly. Each player takes his turn acts based upon this user's strategy, for example raise an army or explore a map, then the next player takes their turn. Usually the timeline of the game advances at each turn. While the player waits for his turn, he or she can't interact with the game, although this player may be computing their response to each of the other players' possible actions. Such a game requires only a modest bounded latency of the network. For example, 1000ms-4000ms per turn is acceptable [37].This makes it easy to believe that this game genre is ideal for the limited resources that today's cellular technology offers. However, mobile multiplayer games are preferable played in short sessions, due to interrupting calls or limited time, but many TBS games have the drawback that they are played over a long period of time. In a single player mode this is not an obstacle, due to the fact that the gamer can save his game state and return to it at a later time. Some popular games in TBS genre have been Civilization (1991) and different versions of Worms (1994-2007).

4.1.2. RTS

In Real Time Strategy (RTS) games the player has to gather resources, building a base, raising an army, upgrading weapons, and take control of specific units, then act. RTS games don't involve turns like the more classic TBS games. Each player has control of his army in real time, thus he or she can at anytime interact with the game, rather than needing to wait for his or her turn. This results in more action, but each player still has to plan their strategy and tactics (far) in advanced. Early in the game the player has to make a decision: what buildings he is going to construct and what kind of army he wants to raise. He also has to consider if he is going to rush the enemy, or wait until he has a really big army before attacking. As a result, the outcome of such games is more dependent on slow execution of strategies and tactics, rather than the player's reaction time hence latency isn't that crucial in such real time strategy based multiplayer session. An RTT of 450 ms is still acceptable from a gamer's point of view [37]. However, Nokia's study [38] of online gaming over cellular network came to the conclusion that a bit older RTS games, such as Age of king (1997) only needed less than 900 ms of latency for a good user experience. Additionally the outcome of the game is not influenced significant by the latency experienced by the players [39]. Popular games in RTS have been Command and conquer Red Alert (1996), Starcraft (1998), and Warcraft 3 (2002).

4.1.3. FPS

In First Person Shooter (FPS) games the gamer sees his character from a first person point of view. He is the character and sees the game world through this character's eyes. Usually the player has to move about in a virtual world, finding different weapons, which he then uses for killing alien, monsters, or in the case of a multiplayer game other players. In such games the gamer would like to have as close real time experience as possible. This is because the gamer's reactions are fundamental to their outcome of the game, if he experiences greater latency he may, in the worst case, already be dead before he even notices the shot. To avoid the gamer's performance in the FPS games being influenced by the latency in a negative way, the latency has to be less than 40 ms [37]. However, Nokia's study [38] of online gaming over cellular network came to the conclusion that a bit older FPS games, such as Quake 2 (1997) only needed less than 200-320 ms of latency for a good user experience. Popular games in this genre have been Duke Nukem 3d (1996), Quake 3 arena (1999), and Doom3 (2004).

4.1.4. MMORPG

Massively multiplayer online role-playing game (MMORPG) has become very popular during the last 5 years. The idea is that a large number of players interact with each other in a virtual world. The gamer creates a character and enters the virtual world with him or her and take control of his abilities. Usually the virtual world is a fantasy world where magic and strange creatures exist. The goal of the game is for the gamer to develop his character by gaining experience which gives the character stronger abilities and hence can fight harder monsters and accomplish more complex quests. The interaction among the players is usually by completing quests together or by changing, buying or selling items among each other. A interesting phenomenon is that these games have created a virtual economy. Money gained by killing monsters, can for example be used for buying items to your character. But this virtual economy has also impacted the real economy by gamers selling virtual money for real money. Also different items in the games can be bought for real money on auction sites such as E-bay.

4.1.5. Mobile multiplayer games

Mobile multiplayer gaming is a new phenomenon, and therefore its definition is not yet clearly delimited. The genre includes not only games running on cellular phones, but also games using a mobile handheld console as the Sony Playstation Portable (PSP). The PSP has a multiplayer option thorough WLAN which Mattias Åkervik mentioned in his thesis [37]. According to Nokia, the company who has spent the most money on mobile gaming and the creator of the first real attempt to implement a gaming device on mobile phones (N-GAGE) [40], mobile multiplayer games can be divided into the following subgenres [38]:

- Round-robin games: The form of games are easy to map to cell phones, as they are based on the same principle as TBS games. While these games are good from a latency point of view in that they do not require low latency, but the drawback is that players may get bored waiting for the other players moves.
- Simultaneous-movement games: Games that are also based on turns. The difference, however, is that in these games the participants play simultaneously and therefore waiting time is reduced.
- Act whenever games: These games allow the gamer to log into the game whenever he or she wants and play as long as he wants. When the player logs out the game state is saved, and can be continued at some later time. There are game solution where the players position can or can not be affected by other participants while the player is offline.
- Slow update games: These games are rather similar to the previous mentioned. The different is that the player's movements are only slowly updated in the game time. The player logs in to the game, moves, and then while the player is online this move is gradually executed until the player logs in next time and updates their orders. This produces participants who frequently, but in short sessions log into the game.
- Shared solitaire games: Are actual single player games, with the difference that you can upload your high score and compare it with others. All players who participate will due to fairness have to start at the same position. This can be achieved by network communication that only requires a small amount of network resources. Additionally at the end of the game the high score upload uses only a very small amount of network resources.

Regarding the cellular gaming, due to the fast development of cellular terminals, an assumption can be made that a similar kind of market will exist for these games as for traditional gaming, even if the mobility aspect can create new gaming phenomenons. There are visions already today how to capture the "non online casual gamers" who play with family, it is very uncertain if these visions will work in practice [41]. One thought is that a family member can see if another of his family is online and ready to play a game session, if so, then they can start a game session with them. However, even if you could send text and voice messages to the participants, it will not give the same social aspect as really being with your family playing. Also the probability to be available simultaneously with another family member is not so big. With more and more advanced cellular terminals the line between them and ordinary computers fades out, and in a not to far future they are merged together, creating the same kind of games. Then with the mobility there are also games that takes into account this in a better way then previously mentioned [42]. Cell phones enable users to achieve high mobility, due to the small compact size of the device. A game play that has taken into account this mobility is called Location Based Service (LBS) gaming. By implementing a GPS into the mobile phones or using another means of determing the device's location a new style game play can be created. Participants get through e-mail, chat etc updated quests that they should accomplish in the real world. Usually this involves travelling to a certain location. When they arrive at this location, then they have to "fight" evil monsters, recover treasures, etc. After accomplishment this, then they receive a new task that has to be accomplished within a certain amount of time.

4.2. Games

Choosing games for a mobile gaming study was not easy. First of all, games downloadable to mobile gaming devices today only have multiplayer functions over Bluetooth and not UMTS/WCDMA. Secondly if you apply traffic study of a PC game and assume that it is the same for a mobile version, then these results will probably not be very accurate. This is due to the fact that the packet sizes, throughputs, and inter arrival times the terminal generates may differ even if the game is the same. However, the networks parameters are rather fixed for the impact of the gaming experience for the different game genre, and Nokia's study uses the same assumption; that the game genres behave the same on a PC as on a cellular terminal although they get influenced by different network parameters [43].

However, an assumption that the traffic generated by a game on a PC does not differ from the traffic created by a cell phone, can be a bit weak. Claypool's [44] study shows that games played on the handheld terminals (specifically the Nintendo DS and Sony PSP) in general create the same packet size as on PC, but that the inter arrival time is shorter for the handheld terminal and hence this creates a noticeable higher throughput. However, the Nintendo DS and Sony PSP can not be compared to cell phones, even if they are mobile devices.

With today's WCDMA modems, easily connected to any laptop, another kind of mobile gamers will appear. Gamers who bring along their laptop together with one of those modems can play their favourite PC game wherever they are (as long as they have WCDMA connection). Due to the shortage of multiplayer games playable over a WCDMA link to cellular terminals today, this thesis will focus more on games playable on a laptop computer together with a WCDAM modem. There are many different games to laptop computers among different genres. Matias Åkerviks thesis focused mostly on PC FBS games. Therefore this thesis will focus more on RTS, TBS and MMORPG games.

Also picking which games should be included in the study had consider the fact that there are thousands of titles released, but only a limited number of games could be tested, hence the PC games chosen were those that are rather popular among gamers and are to some extend possible to implement on handheld devices, or already implemented on handheld devices. Also considerations on gaming sessions had to be taken into consideration. A mobile multiplayer gaming session should not be too time-consuming without an save option.

4.2.1. Warcraft III

Warcraft 3 is a very popular ("bestselling") RTS game which has won numerous awards, including Gamespot's game of the year 2002 [45]. In the game the player has to construct buildings, train a military, and with this army try to conquer his or hers enemies. The session is played on different maps which are provided by the game, or created by the player himself. You can play in a team, against another team manoeuvred by other players, or a computer. There is also an option where you can play "free for all" (FFA), where you have to be the last army alive to achieve victory. At the beginning of a game the player has to choose among four races (human, night elves, orcs, or undead). Blizzard, the game developer, has taken a lot of effort so that the choice of the race does not affect the outcome of the game. They are all equivalent in their power, creating a better gaming experience because the player can master the race of his choice, and still be successful against those who chose another race.

The basic idea of the game is that you have some workers that harvest gold and lumber. Then you use these resources for constructing new buildings, or upgrading existing ones. The gamer can also use his resources to train an army or upgrade his weapons or even learn new magic spells. All this is done to gain an advantage over the enemies' army. Using ones army, the gamer can either explore the map or engage in a battle. In a combat situation a player orders his army to fight a certain unit, then the computer automatically carries out the command. Also when the gamer's army is attacked, it automatically fights back. In contrast to a FPS game the outcome of the game is not dependent upon the reaction of the player; hence such games are largely uninfluenced by latency.

4.2.1.1. Latency

Because latency doesn't affect the outcome of the game, Warcraft 3 has not implemented latency compensation techniques such as dead reckoning [46]. Without dead reckoning the game state is fully determined by the client's position, and it is not used to predict any action, that would then need to be corrected if the prediction was wrong. Also the use of a latency handicap, which ensures that all players experience the same latency, is not needed or implemented. Therefore, a client will experience the game in real-time, regardless of other client's latency [46]. Because Warcraft 3 popularity and the fact that it is a RTS game, a genre that is latency forgiving, it seems as a natural choice to be played over wide area cellular networks.

4.2.2. Starcraft

Starcraft is a previous generation RTS game from Blizzard, who also developed Warcraft 3. It was released in 1998, and was awarded best RTS game of that year [47]. The game is built upon the same principles as Warcraft; the player has to gather resources for constructing a base and raising an army. The only difference is that in Starcraft the player can only choose among three races and gathers minerals and gas instead of gold and wood. The unit control is a bit easier, so it may be preferred when playing on a mobile terminal. Also the game was developed and released at a time when most gamers had modem connection, which just as today's mobile network had high latency. Thus it is worth studying Starcraft as a potential game for mobile gamers, especially if the latency requirements of Warcraft 3 turns out to be to strict for a WCDMA link.

4.2.3. World of Warcraft

World of Warcraft is a MMORPG game and can only be played online through subscription. These subscriptions cost about 12 euros a month, which has led to its great financial success. By January 2007 the game had 8 million subscribers world wide [48], and has won many awards, including Gamespots' game of the year 2004 [45]. When the gamer starts to play for the first time he has to create a character. He can choose among many different classes (Warrior, Druid, Hunter, Mage, Priest, Rogue, Warlock or Shaman) each having special abilities. He can also decide the race (Human, Night elf, Gnome, Dwarf, Troll, Taurent, Orc or Undead), the sex and other characteristics of the character's appearance. After that the character will enter a fantasy land where the player has to explore the landscape, find quests, fight monsters. If the gamer is successful he or she will be rewarded with experience points, items, or money. The experience points are gathered for achieving higher levels, where the characters abilities are upgraded. The gamer can also interact with others, by duels or alliances, to fulfill demanding quests containing monsters to kill. Also a chat facility exists enabling players to gather, trade goods, and so on. If the gamer does not want to sell or trade his items directly with another player, the player can go to a auction house. There goods can be auctioned, and other players can bid on these goods. This is very useful for a character that learns a profession. As he or she can create goods that can be put on auction, which can produce wealth.

4.2.3.1. Implementation on cellular terminals

The game comes on a hybrid CD that can both be run on a Microsoft Windows or a Apple Mac platform. The general system requirements 800 MHz main processor, 512 MB RAM are far beyond what today's handheld terminals can offer. Sadly all of these MMORPG games are hardware demanding. However, due to the high addiction rate of these games, its popularity, and high revenues there would be a huge potential market to develop a light version of the game, playable on a handheld terminal. The light version could contain simpler graphics and only a handful of the quests that the gamer could entertain him with while he is on the move. These quests should, just as in the full version, give his character experience points and items that also can be used in the full version that the gamer plays when they have access to a PC. Even a lighter version that only offers different trade options, looking for auctions, and carrying out the character's profession would still attract a lot of the addicted gamers. Therefore this thesis will further consider the MMORPG genre and particularly this game.

4.2.4. Worms

Worms is a 2D game easily run on clients. You can today download it to many mobile phones. However, sadly the mobile versions today do not offer a multiplayer mode, playable over WCDMA or any other mobile technology. The graphical icon for online gaming are there at startup just as in the PC-version, but you can not invoke it. So the mobile version is sadly just a light copy of the PC-version where they have taken away the multiplayer options. Because I would like to have game traffic to study I used a PC version making the light assumption that the traffic model would have been rather similar if it had used a mobile platform. I choose this game, because it is a rather popular turned based multiplayer game, which is a category that places only small requirements on the network. Also the fact that it already exist on the mobile market, but the existence of only the light version shows that the hardware of the handheld terminals is currently insufficient to implement a multiplayer option, but of course this will change with the continuing evolution of handsets. An advantage that Worms has in comparison to other TBS games in a mobile aspect is that its playing time is rather short. One game session ca be played within 10-30 minutes idle as an example for a commuter trip.

The system requirements for the PC version are 100MHz of processor power, 32Mb of RAM memory, and 300Mb of storage space. The downloaded file size for the mobile phone version (Phone SonyEricsson M600i) is 204 kb. The M600i has a 32bits 208MHz processor, 64Mb RAM (9-21Mb usable), and a 2D/3D graphic accelerator working at 80 MHz. This hardware is sufficient for the game and no quality difference is notice between the PC- and the mobile-version, other than screen size and input interface.

5. Gaming Quality

The gaming perception of the end-user is a very important for the game developers and carriers. Not only must the idea of the game attract the mobile gamers, but the quality that the mobile network can deliver has to be sufficient for a good gaming experience to the user. Today cellular carriers offer a solution for laptop users to use a UMTS link for surfing the internet, but also enabling multiplayer gaming. This will of course attract many gamers but packet loss and latency will always exist on these links. Real time multiplayer games are usually built upon a competitive background. Therefore it would also be interesting to know, what effect the network performance has on the user's performance when competing? But even if the gamers perform equivalent regardless of what network they play on, they may still think that the gaming experience is poor. Therefore it would be interesting to know what effect the network parameters have on the gaming experience? To attract and keep mobile laptop gamers, this is an important issue for the cellular carriers and network vendors.

Of course these are somewhat soft issues. A user's perception of a game can be very individual. A hardcore gamer, especially if he or she is used to playing online at home over a broadband access network, would be more likely to perceive a game session as unplayable in the face of impairments than a casual gamer. Therefore this thesis utilizes a survey of different types of gamers, as to how they perceive a particular gaming session. It also examines how latencies effect a user's performance when playing the game.

5.1. FPS

FPS games are very dependent on the rapid reaction of the user. Therefore we assume that the quality of the perception is greatly influence by the network's performance; with respect to latency and packet loss. Additionally the user's performance could be greatly affected. For example, high latency and packet loss can cause the player to think that there enemy is in front of them, when actually the enemy is not where the player expected them to be.

5.1.1. Quality of gaming experience

There has been some research done in this field, for example with the game Halo. Researchers made the following observations: Latency above 200 ms causes very noticeable delay, and if above 230 ms or if the packet loss is greater than 10% a connection will not be established [49]. After a game session was played, players graded the gaming experience on a scale from 1-5. As suspected the gaming experience gradually decreases with decreasing network performance. Between 0-70 ms of latency and 0-0.5% of packet loss the gaming experience was good (with grades of 4-5). The quality then rapidly reduced and with more than 225 ms of latency or 3% of packet loss the gamers express a desire to leave the game (this corresponds to grade 1-2) [49].

5.1.2. User performance

In the FPS game Unreal tournament the user performance is affected by latency [50]. When two players run an obstacle course competing to arrive first at the finish line, the performance was not influenced by packet loss. However, the latency affects the time needed to run the course slightly negatively. While a player tried to hit an enemy with a sniper riffle, the hit ratio was greatly decreased with increased latency. At a latency of 0ms the gamers hit ratio was 45% while at 200ms the hit ratio fell to 26% [50]. On the other hand, packet loss did not have any effect on the player's precision. The conclusion is that the gamer's performance in the FPS game Unreal tournament is influenced by a variation in latency (0-200ms), but that a packet loss rate of between 0-5% does not have any effect.

Even if Halo and Unreal Tournament are two different games, it seems likely that the user's perceived gaming experience and performance, especially when there is packet loss, may decrease.

5.2 RTS

As mentioned in a previous chapter, RTS games are not as latency dependent as FPS. This is due to the game structure, which in the case of RTS games does not involve many sessions where the gamer's reaction is put to test. From a performance point of view, these games also depend much more on a long term strategy in order to win, a strategy that is not affected by latency. There has been rather extensive research on the effect of latency on user performance in RTS games, this is described in detail below.

5.2.1. User performance

Research conducted by Mark Claypool at Worcester Polytechnic Institute examined the following RTS games: Warcraft 3, Age of Mythology, and Command and Conquer. The two major aspects to success in RTS games are unit control and structural control. Structural control means the gamer's control over which buildings should be constructed or upgraded, or what technology should be employed. Also what kind of army to rise is an important issue in the strategy of structural control. Unit control includes three subcategories; construct, explore the map, and combat. Construct is the control over the working units that harvest resources, make repairs, and construct new buildings. Thereby it overlaps slightly with the structural control. Exploration enables the gamer to know the battlefield where the game takes place. The gamer wants too know where he can find additional resources and where his enemy's base is. This he or she does by scouting the map with his or her units. Combat strategies involve simply grouping units so that artillery is behind the melee or letting individual units do some special focused damage. It is also possible to let the computer's artificial intelligence handle the combat, without the gamer needing to issue any further commands.

By breaking down the game session to building, exploring, and combating, then giving the gamers different latency handicaps (0-1000ms), the study [46] revealed how good players performed in the different categories. When building the complete technology tree (construct all buildings and researching all technology) there was only a very slightly difference among the players with no induced latency. In Warcraft 3 as an example, the whole technology tree takes 480 seconds to complete without any latency and only 14 seconds longer if the gamer has up to 3.5 seconds of additional induced latency. These 14 seconds is only 1% longer, so the latency has almost no affect on the gamer's constructing performance. In the games Command and Conquer and Age of Mythology the situation is the same. For the exploration session the Warcraft 3 map took 250 seconds to explore without any latency. The time needed to explore increased linearly by 6 seconds per 100 ms of induced latency. This is only a minor effect, so small that from a gaming performance point of view it can be neglected. The same was true for the two other RTS games. Also in the combat session the gamer without latency only gains slightly advantage in the three games. The conclusion occurs because every unit has an individual unit score depending on its strength and health. Comparing the non-lagging player's unit score and the lagged (delayed) players' unit score at the end of the battle and plot it, it is easy to come to the conclusion that latency does not have any significant effect.

Claypool study states that RTS games are also rather latency forgiving. In the interval between 0-500 ms RTS games run smoothly. With a latency of between 500-1000 ms the games still run smoothly with regard to updates, but commands are executed with a noticeable delay which negatively impacts the gaming experience. With latencies above 1000 ms game play is difficult even for an RTS game. This statement will be further investigated later in this chapter

5.2.2. Quality of gaming experience

To determine the perceived user experience of the popular RTS game Warcraft 3, an experiment was conducted in which both casual and hardcore gamers played Warcraft 3 under different network circumstances. To carry out the study 9 gamers, 4 beginners who never had a Warcraft 3 experience and 5 who play the game several times before, where invited to play different Warcraft 3 game sessions. The two crucial network parameters packet loss rate and latency were changed during 11 different sessions. The numbers of game sessions where chosen so that a good range of packet loss and latencies was carried out. The good range of the parameters and their interval was determined by before the study playing the game under certain controlled network parameters to see what impact the parameters had on the gaming experience. The intervals for the study had to be chosen so that there was a small perceived difference, but no so big that the precision got lost.

During each session the gamer had to determine the quality of that sessions gaming experience. This was done by giving each game session a grade (between 1-5). These grades had the following meaning;

5 => game runs perfectly well, no disturbance noticed
4 => slightly disturbance, acceptable game play

3 => disturbance but still playable

2 => so much disturbance that the gamer had the desire to leave

1 => impossible to play

To see if there was any difference in perceived gaming experience between casual and hardcore gamer the gamers were divided into two categories. The gamers where able to choose a race by own interest, but had to play with the same race during all 10 sessions. Every session also contained different game play; such as exploring, building, and fighting. The first game session was played without any disturbance so that the gamer could get a reference to how a perfect game session should be perceived. Also the 6th game session was played without any disturbance, so that the gamers could refresh his memory how the gaming experience should be without any disturbance.

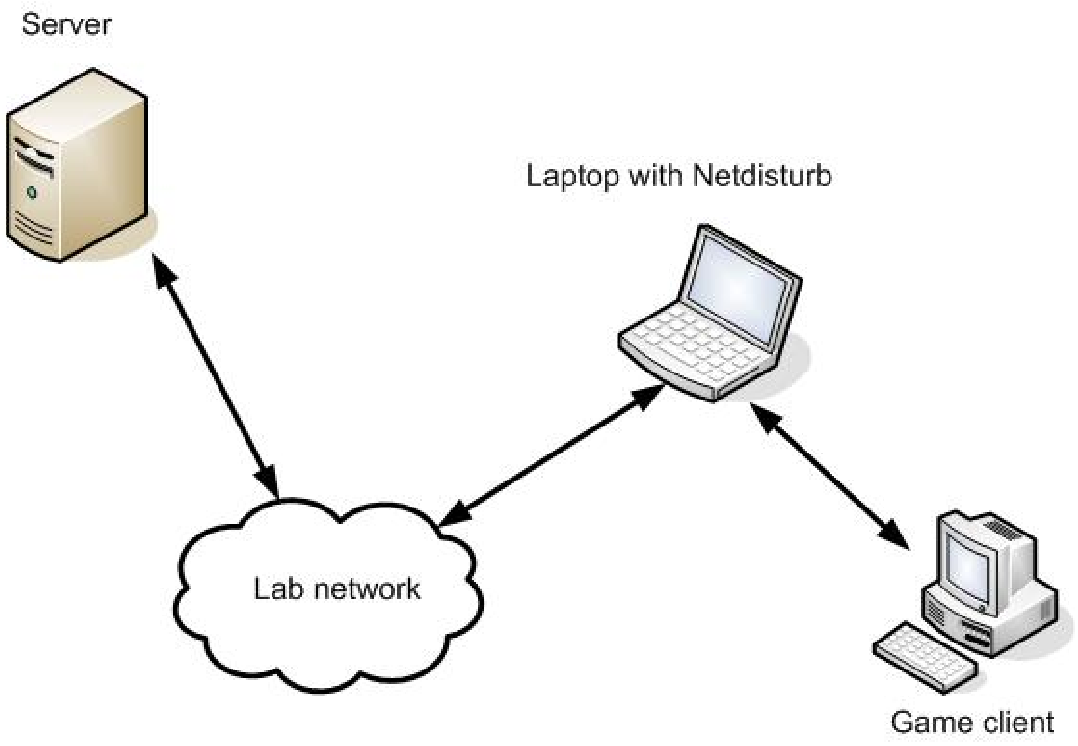

Figure 12: *The lab setup for the study of gaming quality.*

The gaming client was a PC running on Windows XP, connected to a laptop that had a bridged connection to Ericsson's lab network (see Figure 12). The laptop computer run an application called NetDisturb [51] that could emulate different controlled network parameters for the link. The gaming client was then connected to battlenet, where a 3v3 game was joined.

Before the gamers where allowed to play, ping run on the gaming client to assure that NetDisturb had the desired effect. Each gamer received introduction to the survey, and it was explained that the quality of the game sessions would be disturbed, but in random order; the first session may not be the best and the last may not be the worst. The gamers were told to try to focus on grading the disturbance, rather than whether they enjoyed the actual game or not. Because hardcore gamers worried about their characters scoring, they were told grade the game regardless on how they think they perform; the game session was only played for fun. The randomly chosen network settings for the 12 game sessions where chosen among the following range of values;

Delay [ms]: 0, 500, 750, 1000, 1250
Packet loss [%]: 0, 4, 8, 12

Please refer to appendix A to see how the parameters where distributed among the game sessions. For each game session the gamers then had to record the grade given, and if he/she had the desire to terminate the game prematurely.

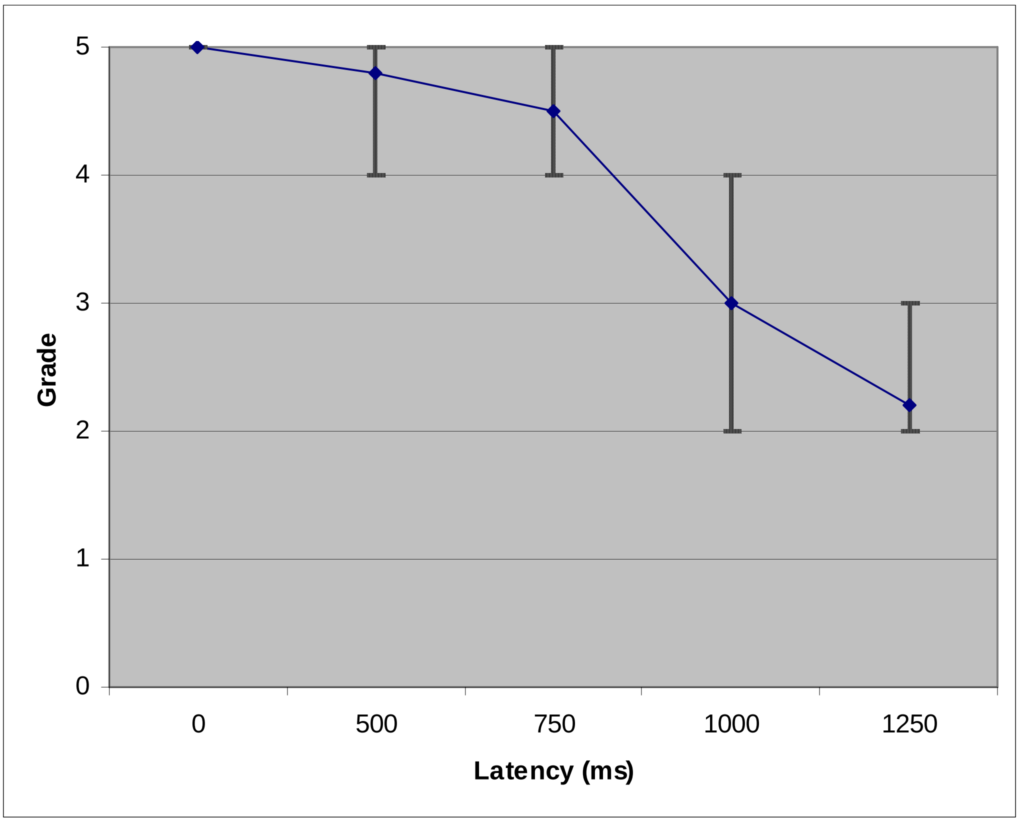

Figure 13: *Perceived gaming experience at different latencies with error bars*

Figure 13 shows the quality of perception with which the gamers rated the game sessions, while packet loss was set continuously to 0%. The reader can clearly see how the grade falls with increased latency. The first noticeable result of the studies was that the answers where rather homogeneous. At its maximum only one grade was different than that by another gamer for the same gaming session, except at 1000 ms of latency. So the perceived gaming experience was rather equal among the gamers, this was regardless of if they considered themselves to be hardcore or casual gamers. The user perception at high latency is that the commands in the game are lagging. For example if the gamer commands his army to go to a certain spot on the map, then the response to this command is not immediate. In these cases the gamer may not know, for a second, if the command was actually given, and thus give a new command, which can be a bit annoying. What is interesting to see is that even with 1 second of roundtrip latency all the gamers graded the quality in average as a 3 and did not desire to leave the game prematurely. However one hardcore gamer graded the experience as a 2 and one casual gamer as a 4. At 1250ms of latency only 2 out of the 8 gamers thought the session still was playable.

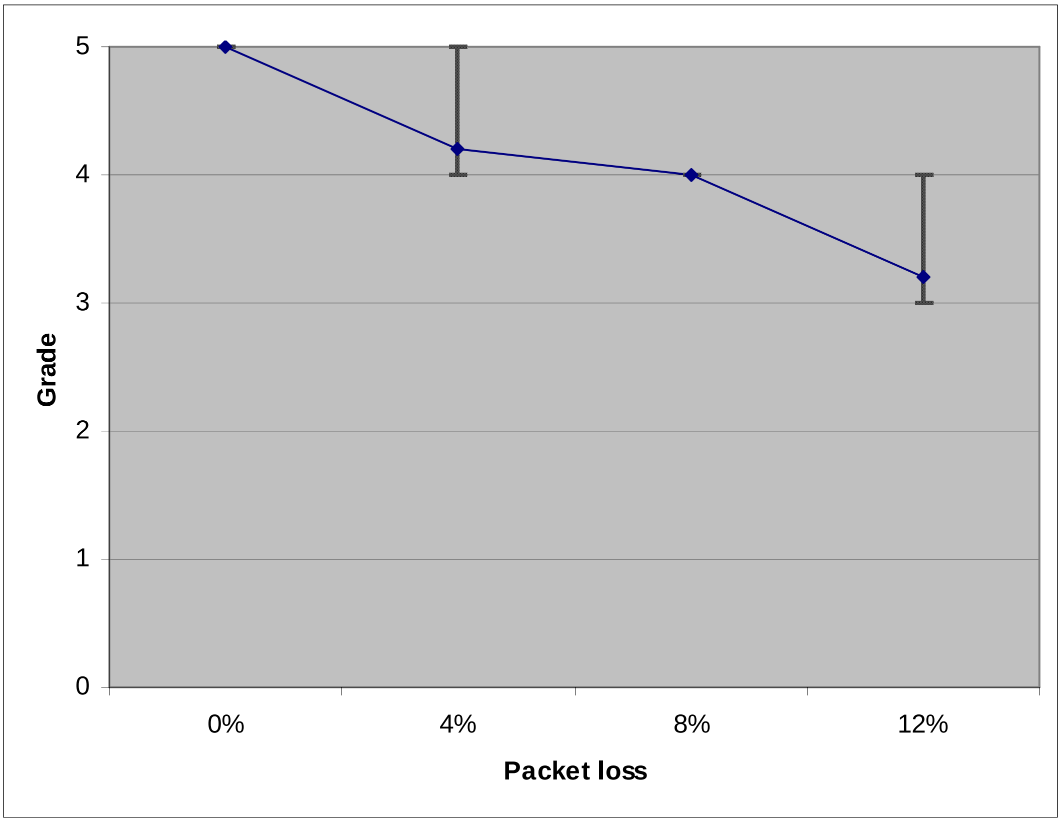

Figure 14: *Perceived gaming experience at different packet loss .with error bars*

The effect of packet loss is different from the delay. As figure 14 shows, the grades do not significantly change as packet loss varies from 4-12% while the latency was set continuously to 0ms. However, at 12% packet loss the gaming quality starts to suffer, but still perceived as playable by all the gamers. At 8% of packet loss all the 8 gamers in the survey graded the experience as a 4.

When a game session suffers from both latency and packet loss then the requirements needed to satisfying the gamers is more strict. Already at 500 ms of latency and 8% packet drop the gamers tend to think that the game is unplayable. At 750 ms and 8% packet drop most gamers graded the session as a 2 if not a 1, and all had the desire to leave the game. At 750 ms and 12% packet loss all test subjects perceived the session as a horrible gaming experience.

5.3. MMORPG

Nowadays the MMORPG genre has become very popular, but at the same time it has generated a huge demand on the networks. Many laptop gamers might soon play MMORPG games over WCDMA links. Hence is interesting to study gaming quality on this particular genre. When a WCDMA link is used, the carriers have to offer their MMORPG clients adequate network performance in order to prevent having unsatisfied gamers. But what happens if the performance of the network starts to suffer?

As it was previously mentioned, MMORPG games are demanding, so it seems natural to make the assumption that a change in the network parameters has a rather big impact on the gaming experience. To verify this assumption 8 gamers were invited to rate how good their gaming experience was when playing World Of Warcraft (WOW). Games such as WOW – which this study will be based on, may suffer if the characteristics of the network connection changes, we based the study in this report on emulation using different network parameters. The interval at which we change the network parameters for the study were chosen in advance, so when changing them their effect on the gaming experience only had a small perceived difference. If the intervals had been longer the precision of the user evaluation would have been reduced as the user would not so easily be able to compare their current performance with earlier performance. Seventeen game sessions constituted the complete study. During some earlier game sessions, it was noticed that WOW is rather sensitive changes in network performances and therefore the study needed many game sessions.

The chosen gaming sessions used a combination of the following values;

Delay [ms]: 0, 400, 500, 600, 700, 800
Packet loss [%]: 0, 4, 6, 8, 10, 12

During each session the gamer had to determine the quality for that session by assigning each game session a score in the range from 1 to 5. These scores were to be as follows:

5 => game runs perfectly well, no disturbance noticed
4 => slightly disturbance, good game play

3 => disturbance but still acceptable game play

2 => so much disturbance that the gamer had the desire to leave

1 => impossible to play

In this study the participants had some WOW gaming experience, this was necessary due to this game's high complexity. Some of the participants were allowed to use their own accounts where they had their own characters stored; while others used an account created specifically for conducting this study. Each gaming session was played over at least 15 minutes, so that the gamers could have sufficient time to carry out a number of different actions.

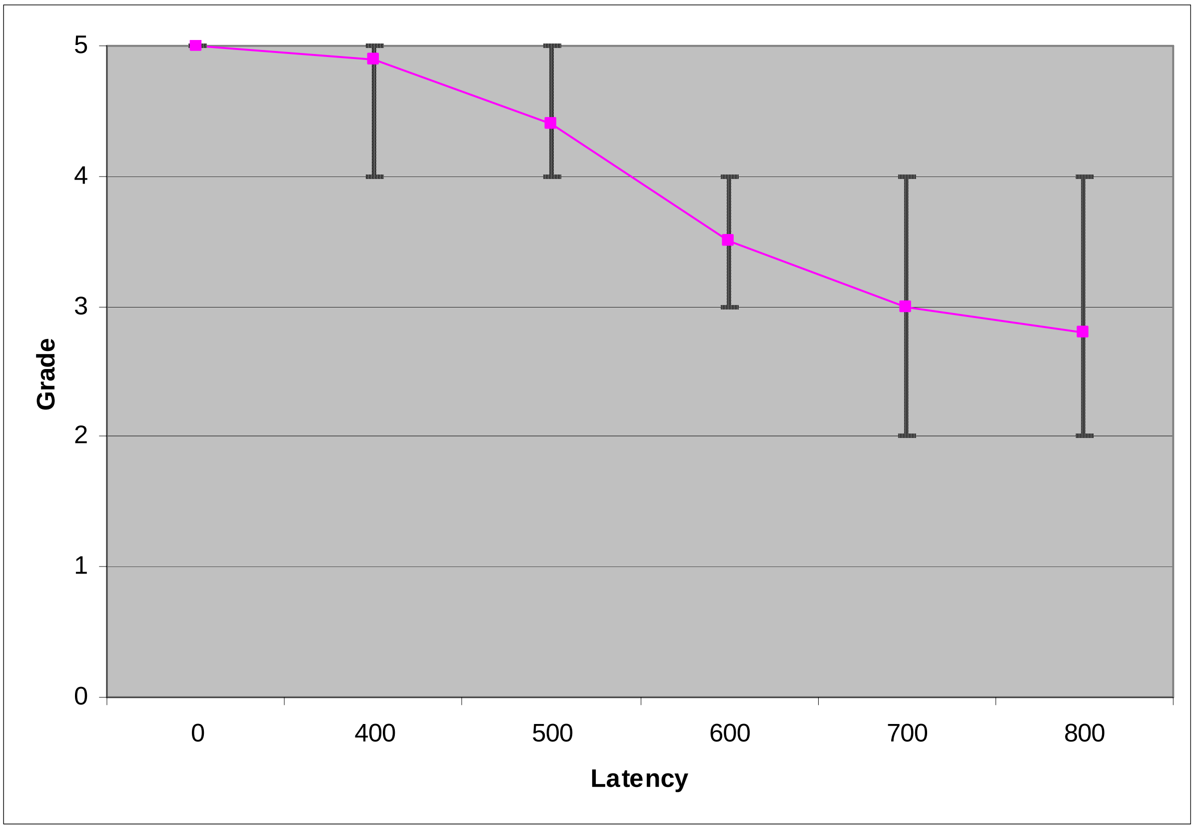

Figure 15: *Perceived gaming experience at different latencies with error bars*

The gaming client was executed on a PC running Windows XP, connected to a laptop that had a bridged connection to Ericsson's lab network (see Figure 12). The laptop computer ran an application called NetDisturb [51] that could emulate different controlled network parameters for the link. The gaming client was connected to a WOW realm, where the gamer entered the virtual world with his character. Before the gamers where allowed to play the session, the latency was checked in the gaming window (WOW has a built in latency indicator) to assure that NetDisturb had the desired effect on the game. Sadly the latency fluctuates highly on the link between the laptop and Battlenet. Unfortunately, this fluctuation is hard to avoid, but it easy to monitor. In this survey the fluctuation was accepted as long as the latency different by less than 10% of the desired value.

In Figure 15 the reader can see the impact latency has on the gaming experience, while packet loss was set continuously to 0%. Note that the answers where rather homogeneous among the participant up until the greatest amounts of delay. At 700ms of latency two gamers graded the experience as a 4, two other as a 2, while the rest scored it as a 3. At 800ms of latency the same two gamers as before perceived the session as 4. However, two gamers who gave the 700ms session a 3, now graded the experience as 2 resulting; thus, four gamers thought it was unplayable. The last two gamers graded the experience as a 3 just as at they did at 700ms of latency. This rather heterogeneous grading at high latency could be due to two things. Either the fluctuation of the latency had a rather big impact on the quality of the game sessions, or the gamers played with different styles in their game sessions. Some actions in WOW are not as latency demanding as other. However, as the different game styles played during each session were monitor, it is more likely that the heterogonous grading was due to the fluctuation of the latency.

The reader can clearly see in Figure 15 how the grade falls with increased latency. From latencies between 0-600 ms, only one grade was different from that by another gamer for the same gaming session. At higher latencies the answers differed a bit more as can be seen in the Figure 15. At a latency of 700ms the gamers overall still perceived the game playable; however, one out of the 8 gamers graded the experience as unplayable (i.e., a score of 2). At 800ms of latency three gamers gave the perceived gaming experience a score of 2 and only one gamer graded it as a 4. The conclusion is that the game can still be considered playable at 700ms, but at 800ms a significant percentage of the gamers desire to quit playing. The user perception at high latency is that the executions of commands in the game are lagging. For example, if the gamer commands his character to attack an enemy or talk with another character, then the response to this command is not immediate. In these cases the gamer may not know, for nearly a second, if the command was actually performed, and thus they may give a new command, which can be a bit annoying.

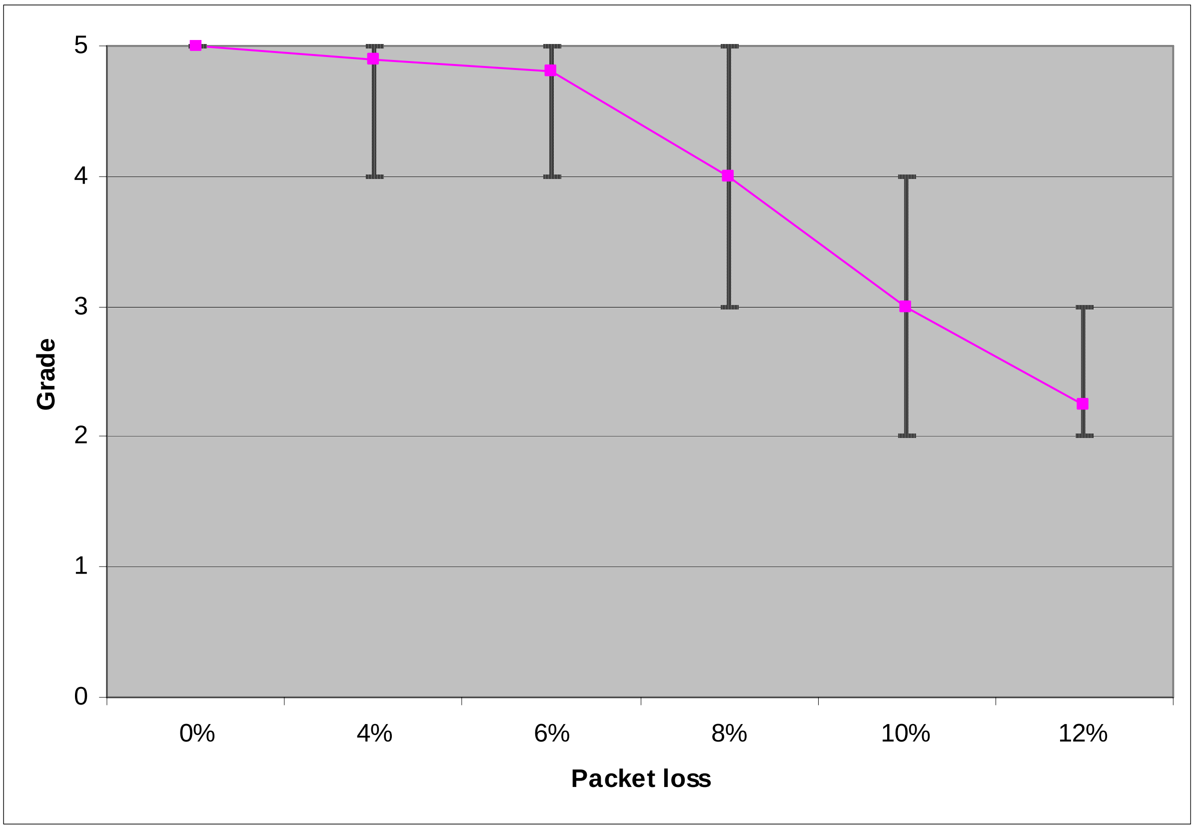

Figure 16: *Perceived gaming experience at different packet loss with error bars*

In Figure 16 the latency was continuously set to 0 while the packet loss fluctuated. As the reader can see the packet loss has also a significant impact on the gaming experience. However, it is only above 6% packet loss that the gaming experience starts to decline rapidly. At 10% of packet loss the gaming experience starts to become unplayable for some gamers. Two gamers grade it as unplayable, but two others still think the gaming experience deserves a grade 4. At 12% packet loss the answers were more homogenous; half of the test subjects perceived it as unplayable, while the other half thought it was still playable. The greatest degradation at high packet loss is that occasionally some commands are never carried out (as they have been lost). A gamer can for example command his character to speak with another character, but even if he waits, nothing happens. This is very frustrating for the gamers, especially in battle, when command spells/attacks are crucial. The gamers who graded the gaming experience as a 2 at 10% packet loss probably experienced a situation like that, while the ones grading it a 4 where lucky and did not suffer from this during that particular gaming session. When the link both suffered from latency and packet loss at the same time the game was perceived as unplayable at 400ms of latency, and 6% of packet loss, with one gamer grading the experience as a 2. With the same latency but a packet loss of 8% the gamers perceived the session as unplayable, and one gamer even giving the experience a score of 1. At 600ms of latency and 6% of packet loss the grades did not differ so much from when the latency was 400ms with the same packet loss; only one gamer graded it as a 2 while the rest gave graded it as a 3. At 800ms of latency and 4% it was interesting to see that the gamers did not think the session was worst than at 800ms of latency and no packet loss at all.

5.4. Conclusion

What is interesting to see is when a game is playable or if it is even playable with a good quality with respect to a given network performs. At the moment we have only considered the network to suffer from latency or packet loss, but both could occur at the same time. To get a better visual understanding what happens with the gaming quality when the network connection simultaneously both suffers from packet loss and latency we plotted the result of our measurements. In Figure 17 the reader can see the gaming experience of Warcraft 3 as a function of packet loss and latency. By good quality (grade above 3.5) it is meant that the game runs smooth, without any disturbances or only minor disturbances in terms of freezing screens for example With an acceptable game (grade between 2.5-3.5) play the gamer perceives disturbances rather frequently, but not at a level where he or she wants to quit the game session. Even if the conclusions are presented as the game play changes at a rather finite bound, the reader has to understand that this is more of a theoretical bound. The game play changes gradually with the network parameters. The reader can see the bound as the last point where the game play is still acceptable/good quality.

It is easy to see that for a good quality game play that low delay is much more crucial than packet loss rate. The reader can see a packet loss at 14 % still can give an acceptable game play if the latency is 0ms. But only with a minor increase in latency the threshold of the packet loss for an acceptable game play drop rapidly, but then flattens out between 250-750ms. When Warcraft 3 is played with latency above 750ms the delay becomes critical.

Despite the packet loss being 0% the game only performs acceptable up to 1000ms of latency.

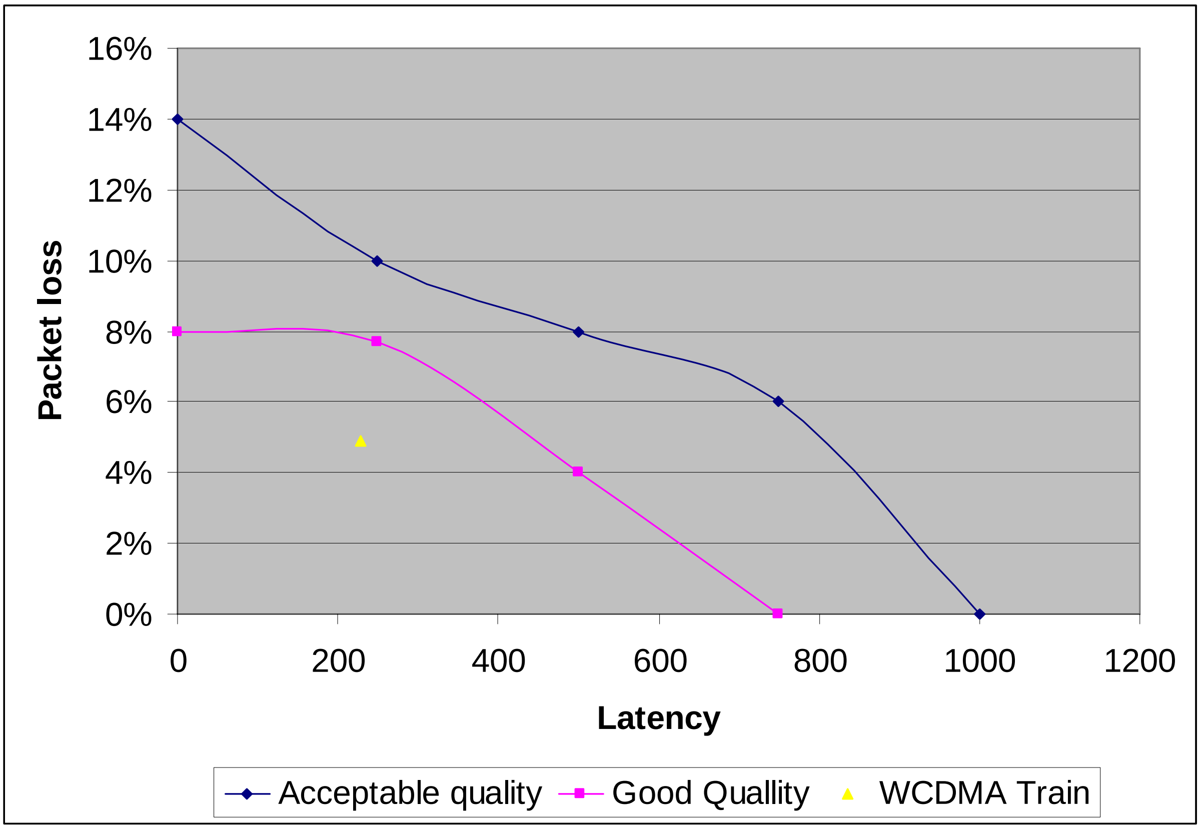

Figure 17: The performance of Warcraft 3 as a function of packet loss and latency.

Latency above causes a bad gaming experience. At 250ms of latency the packet loss has a critical impact on the gaming experience. With a change of only 2-3% of packet loss the gaming quality shifts from acceptable to good quality. Above a latency of 250ms the slope of accepted packet loss is linearly down towards the point of 750ms of latency and 0% of packet loss, which is a spot where the game still performs very well.

As we know since earlier (section 4.3.3) both a stationary WCDMA link and a link on a bus have between 120-190ms of latency together with 0% of packet drop. This network performance will give the gamer a good Warcraft 3 gaming experience. On the commuter train the latency was 250ms together with a packet loss of 4,9%. Also this network performance will give the gamer a good RTS gaming experience.

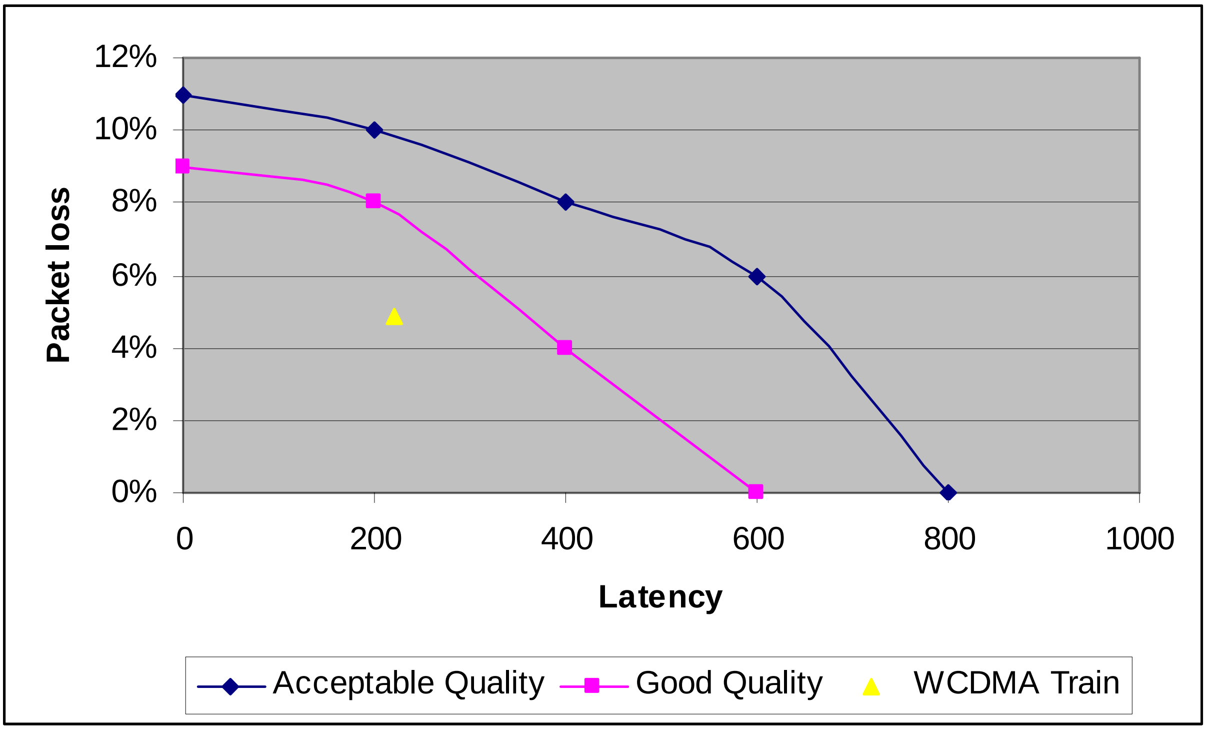

Figure 18: The performance of WOW as a function of packet loss and latency.

In Figure 18 we can see the gaming experience of WOW with different network parameters. In the case of no latency the gamer will have an acceptable gaming experience at packet losses as high as between 8-12%. Also at latency between 600-800ms, in the case of no packet loss, the gamer perceives the game session as acceptable. If the network performs better the gaming experience tends to be good with only minor disturbances. What is interesting to see is that at 200 ms and 400 ms of latency a packet loss of 4% divides the good quality from the acceptable. However, at 600 ms of latency the span is 7% of packet loss. This is a sign that at high latency the network is not impacted by packet loss to such a great extent. In other words, high latency always degrades the perceived gaming quality regardless of great decreases in packet losses.

As we know since earlier (section 3.3.3) both a stationary WCDMA link and a link on a bus have between 120-190ms of latency together with 0% of packet drop. This network performance will give the gamer a good WOW gaming experience. On the commuter train the latency was 250ms together with a packet loss of 4,9%, resulting in a network performance that still gives the gamer a good WOW gaming experience.

However, the reader has to have in mind that the measurements done in the previous chapter were collected over a period of many minutes and the results given are the average. Also the line of good quality game play in Figure 18 is as previously discussed not a finite limit. Hence, a gamer playing WOW on a commuter train may occasionally experience worst gaming quality.

6. Traffic modeling

Real time gaming over mobile phones is gaining popularity. Not only among the users of cellular terminals, but for users of laptops with WCDMA cards – thus mobile gaming is occurring today. For example the Swedish carrier TRE is offering the requisite connectivity through an external WCDMA/HSDPA modem. TRE is offering high speed connectivity as a flat rate in order to (not only) attract mobile business users, but also to attract other customers. Among this wide audience there will be many gamers, who will use their computers for playing different online games over UMTS/WCDMA. If this application increases in popularity, it could fill up the 3G networks with gaming traffic from regular computer users.

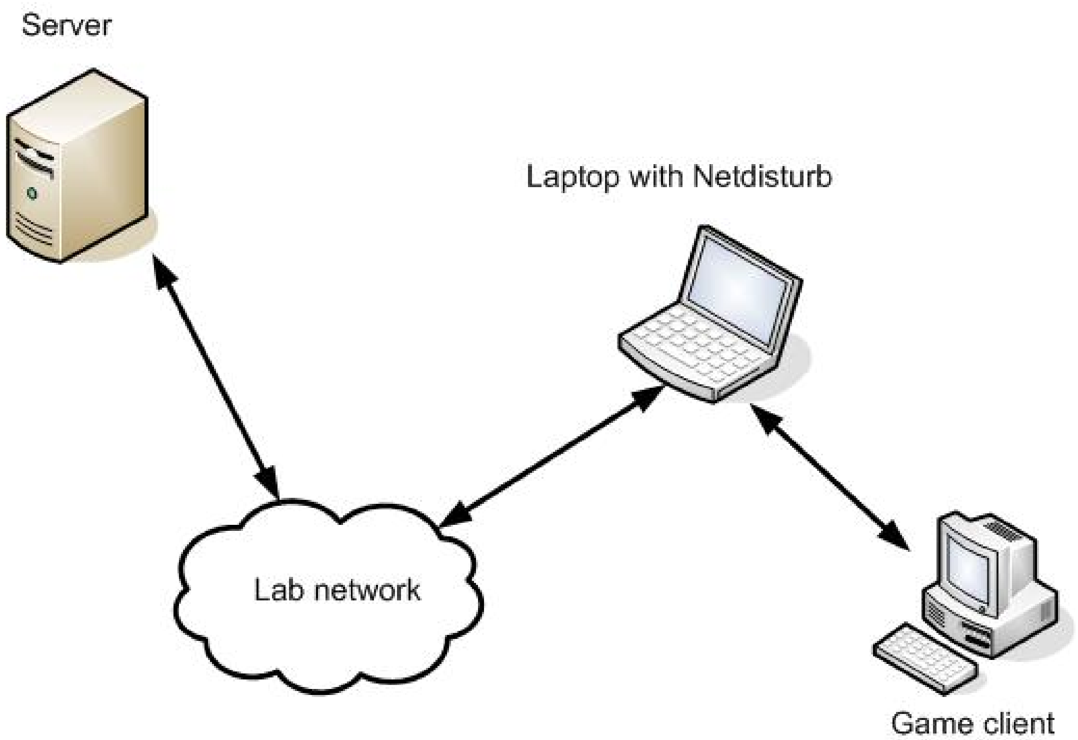

Figure 19: *Lab arrangement while measuring game traffic while emulating WCDMA link.*

In order to better adapt the cellular networks to gaming it is crucial to gain a better understanding of the gaming traffic online games generate. Not only how much bandwidth (with respect to input/output traffic) they consume is interesting, but also what size and inter arrival time the packets have can be of great importance in order to better optimize the network.

For measuring the game traffic it would be interesting to see the results both for a fixed and a WCDMA link to see if the game traffic differs. The rest of the parameters, such as the network location of server, should of course be the same. To emulate a controlled WCDMA link a program called NetDisturb was used. The program was installed in a laptop Figure 19 which with was connected between the gaming client and the lab network. Net disturb could then induce latency and packet loss for the link, and it could also decrease the effective bandwidth. On the laptop two measuring tools was also installed: Comview and Etherreal. Both are packet sniffers that also can filter traffic and generate a number of different types of reports. To adjust NetDisturb to emulate a UMTS/WCDMA link the values from the bus experiment in chapter 3.3.3 (latency 182 ms, packet drops 0%). The throughput for the downlink was set to 300 kbps and the uplink to 64 kbps based upon the theoretical values given in chapter 3. However, as mentioned before the throughput of a WCDMA link is more than sufficient for multiplayer gaming, so this throughput limitation has probably no effect on the game. The games where also played over a real WCDMA link (Figure 20), measuring the traffic with a packet sniffer running on the same computer as the gaming client, while the connected laptop standing still. This was done to see if the result from the emulated WCDMA link differed from the real WCDMA link. The results of this test indicated that the emulated link created the same traffic as a real WCDMA link, which was expected. Therefore in this chapter as the mobile link is referred to Net Disturb set to emulate 182 ms of latency or a real UMTS/WCDMA link that is stationed; the results presented are the same for both.

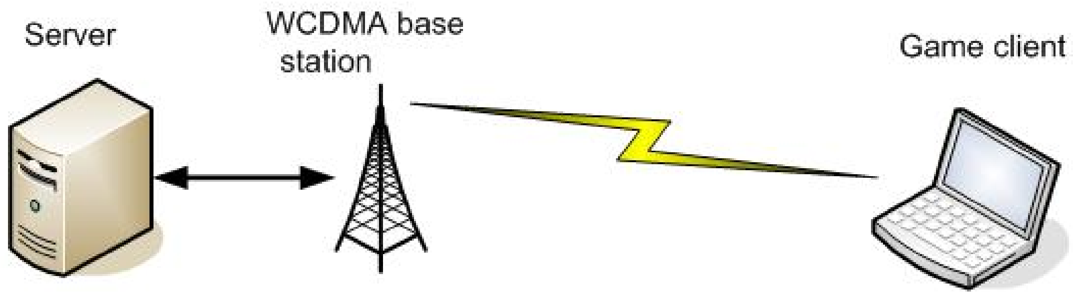

Figure 20: *Lab arrangement when the game was played over a UMTS/WCDMA link*

6.1 Bandwidth

Figure 21 shows the throughput experienced by a game client in a multiplayer game involving 6 participants, note that WOW is a MMORPG and thus could have thousands of participants. The throughput varies significant by the different games; especially Starcraft with its 3.5 kbytes/sec. Also can bee seen is that the throughput is slightly lesser when you have an emulated latency. This has to do with the variation of packet size as will be described in section 6.2. As can be clearly seen in Figure 21 all throughputs are rather low. Actually they are much lover than Peter Rysavy in Table 1 suggest as the throughput for a multiplayer games. With this low throughput requirement a UMTS/WCDMA link should not have problems delivering this data – in fact it will fit in a single 64kbps channel.

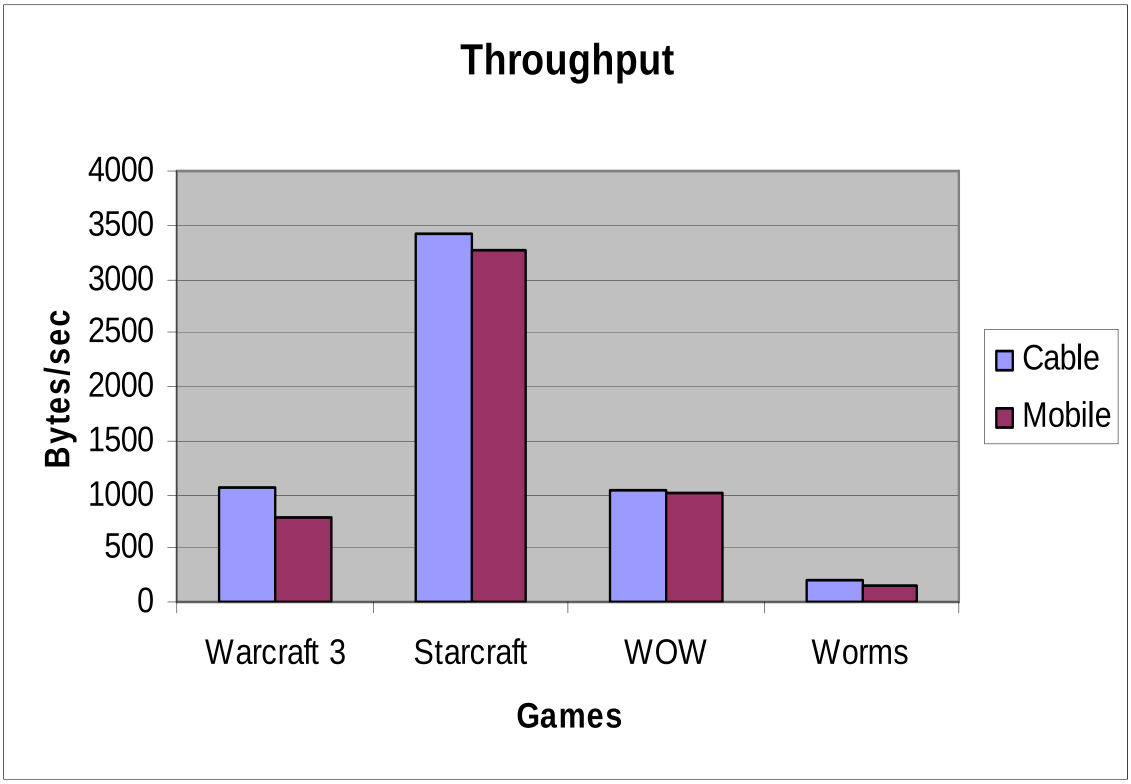

Figure 21: *Throughputs of the games studied*

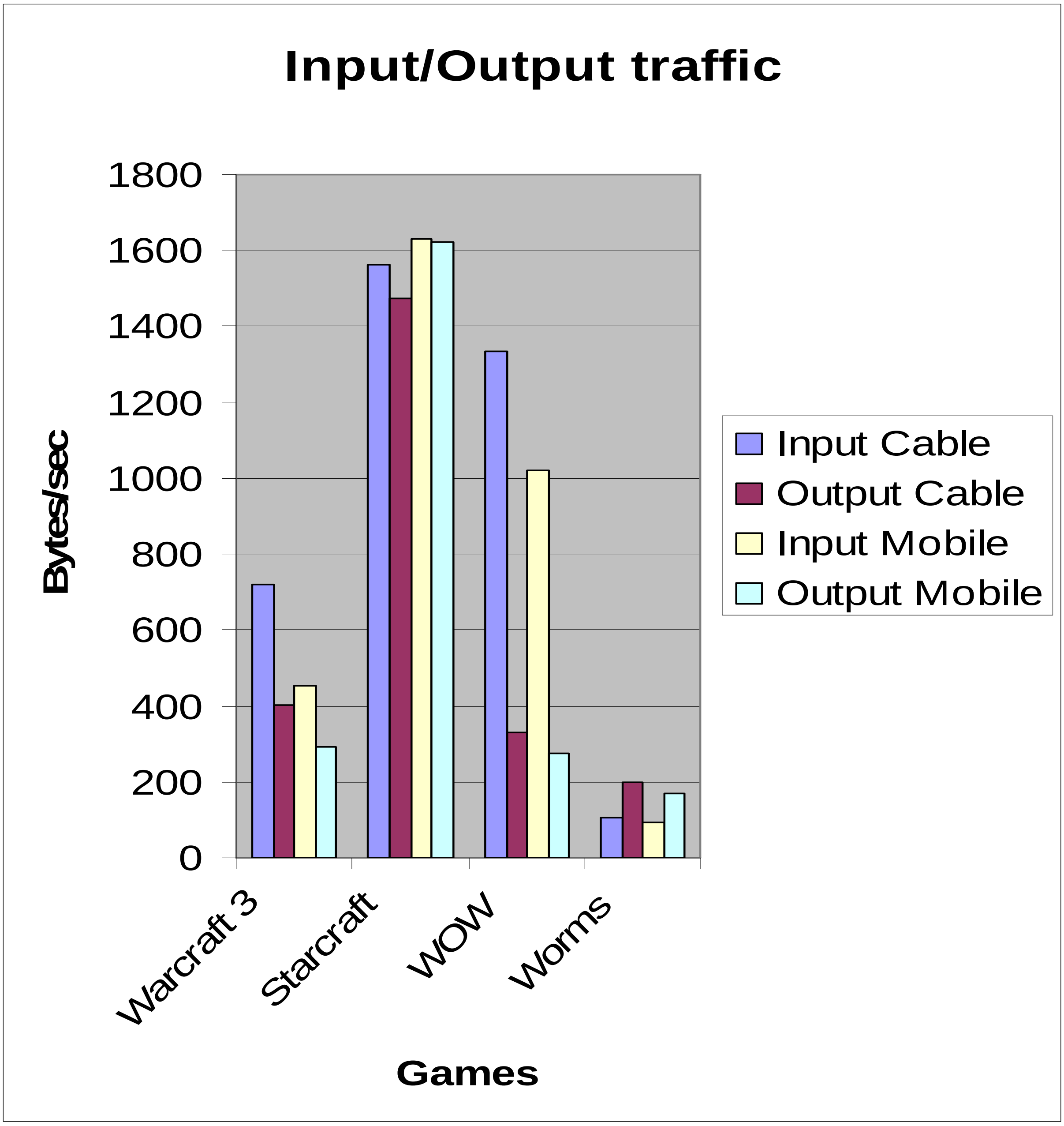

Figure 22: The input and output traffic bandwidth for the different games.

Figure 22 shows the input and output bandwidth observed for the emulated and actual mobile gaming traffic. Just as in the previous case Starcraft is an exception to the behavior of the other games. Its input traffic does not differ significantly from its output traffic. Also worms is special, it sends more information than it receives. The other two games both have a much bigger portion of input traffic than output traffic.

Of course the throughput is fluctuating over each gaming session and the values above are only average. This fluctuating could be big or small, but still create the same average. Therefore a throughput capture was made over time and presented in. As you can see, most of the games fluctuate rather much. Only Starcraft has a rather steady flow with fairly regular small peaks while the other graphs are rather unsymmetrical.

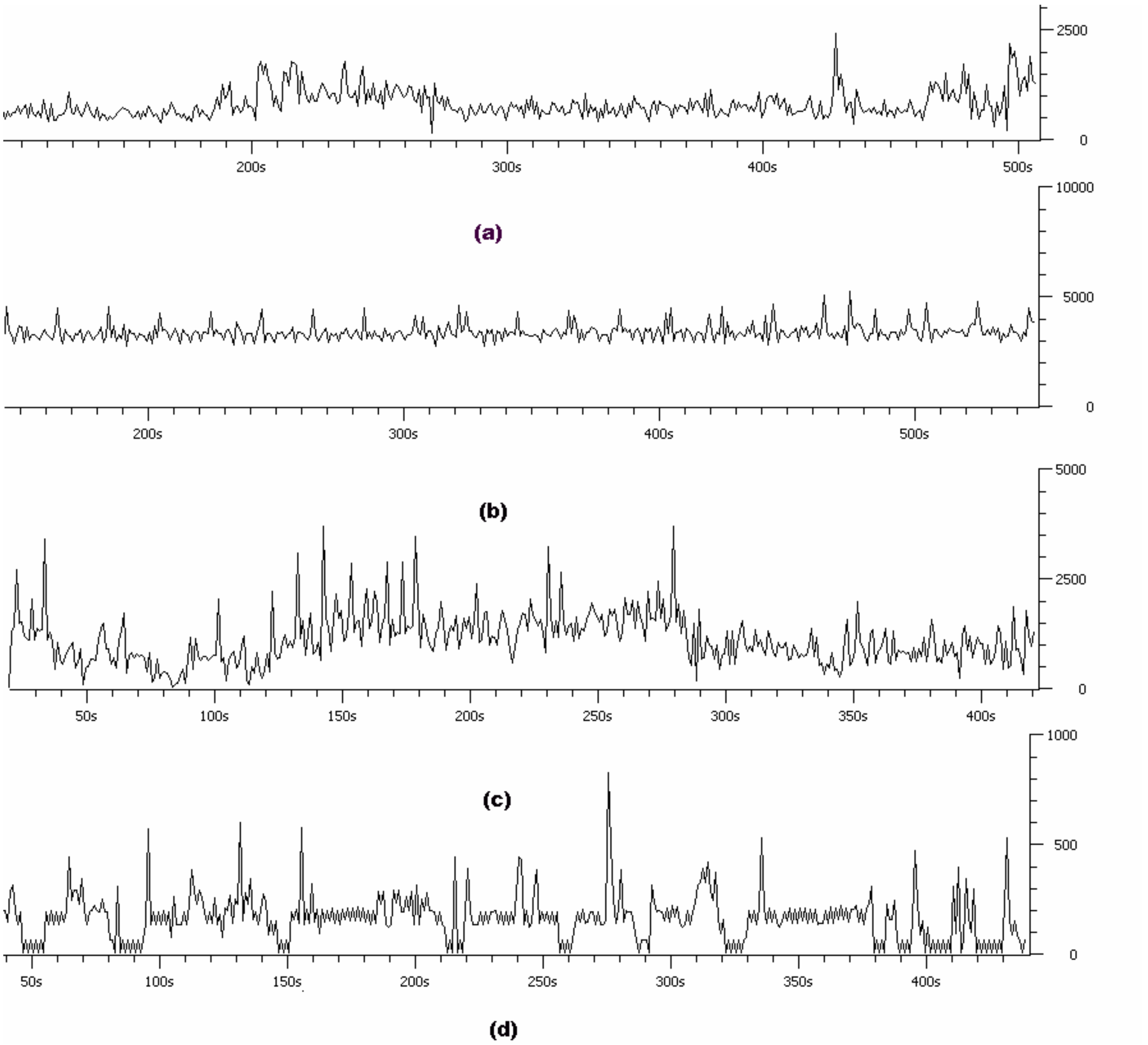

Figure 23: *Throughput variation for different games. Y-axes throughput in bytes/sec X-axis time in sec (a) Warcraft 3 (b) Starcraft (c) WOW (d) Worms.*

6.2 Packet size

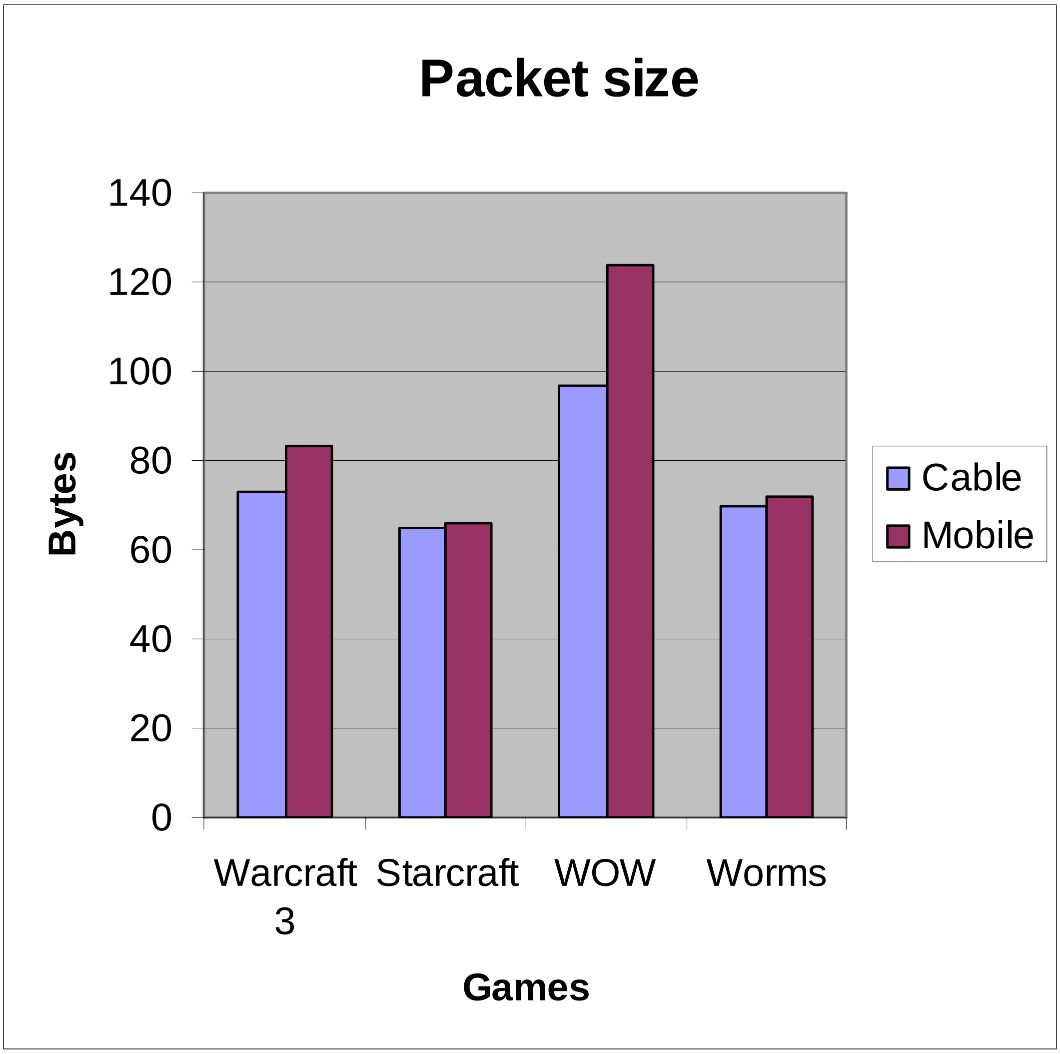

Figure 24: *Packet size sent and received by the different games.*

Figure 24 shows the packet sizes (both inbound and outbound) for the games in the study. All games produce rather small packets, averaging between 60-125 bytes, compared with the packet size of typical Internet traffic that is over 500 bytes [52]. What is interesting to see is that the packet size is affected by latency. When played over a UMTS/WCDMA link the packet sizes increase. Some games packet sizes are more affected by this than others.

As can be seen in Figure 25 the interarrival time also gets influenced by latency. Latency cause the interarrival time to be greater. In other words, the game sends and receives packets in less frequency, so the game has to pack in the information in less packets resulting in that every packet is bigger. Notice however that the overall throughput is less with higher latency, but it is not that low that it affects the size of the packets.

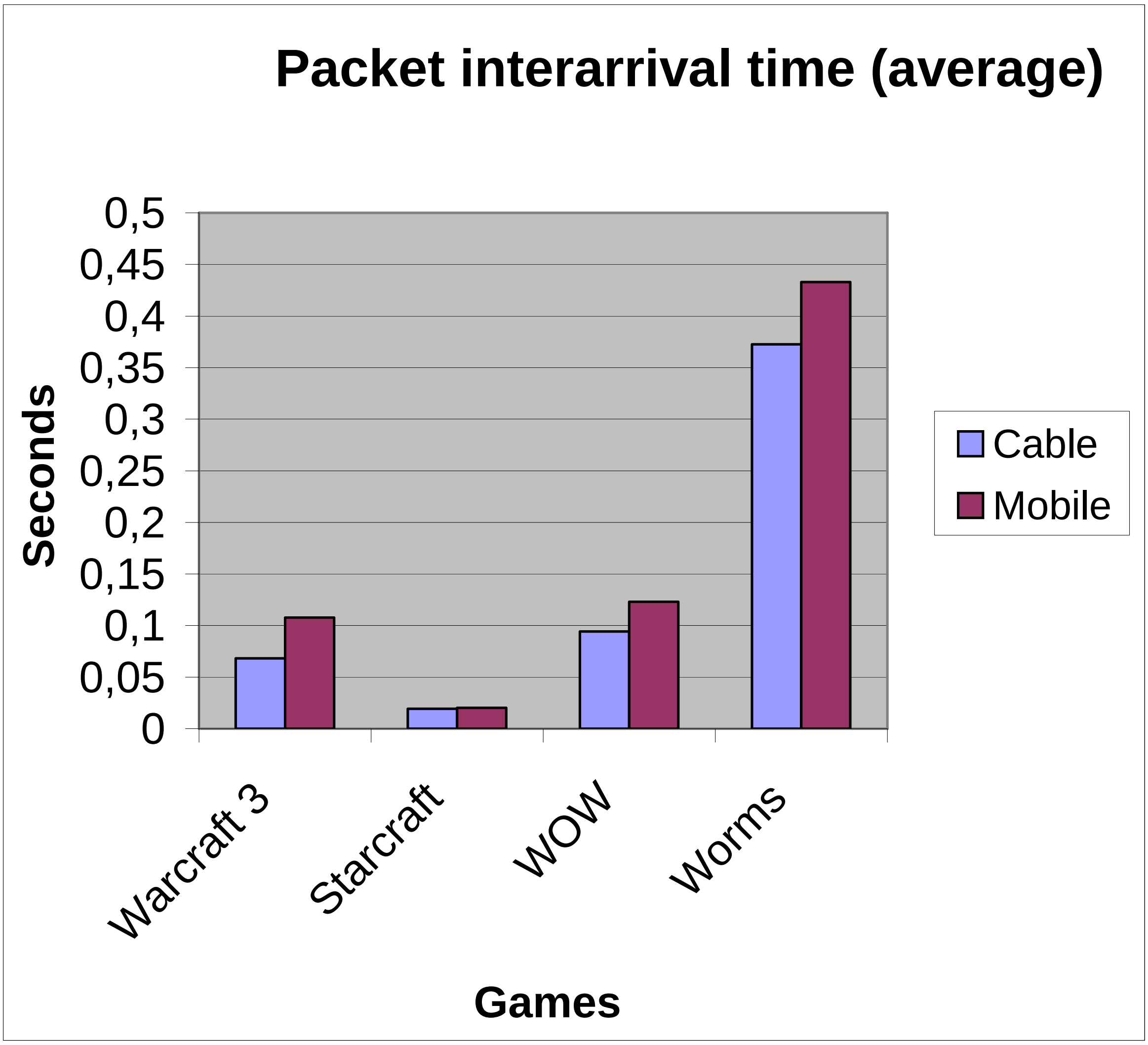

Figure 25: *Packets interarrival time for the games in the study*

Another interesting aspect is to see how the packet sizes are distributed. Some games may generate a fixed set of packet sizes while another one may be very heterogeneous. As seen in Figure 26 Starcraft has a rather homogeneous packet size where most of the packets, over 90%, are between 64-127 bytes. As previously discussed the reader can notice in Figure 26 (b), the effect of a WCDMA link with its higher latency does not cause Starcraft to produce any significant differences in packet sizes. In Figure 27 de reader can see a packet distribution function, which gives a exact understanding how the packets are distributed. As can bee seen Starcraft produces many packets of the same size. In fact 75% of all packets that the game produces are 65 bytes..

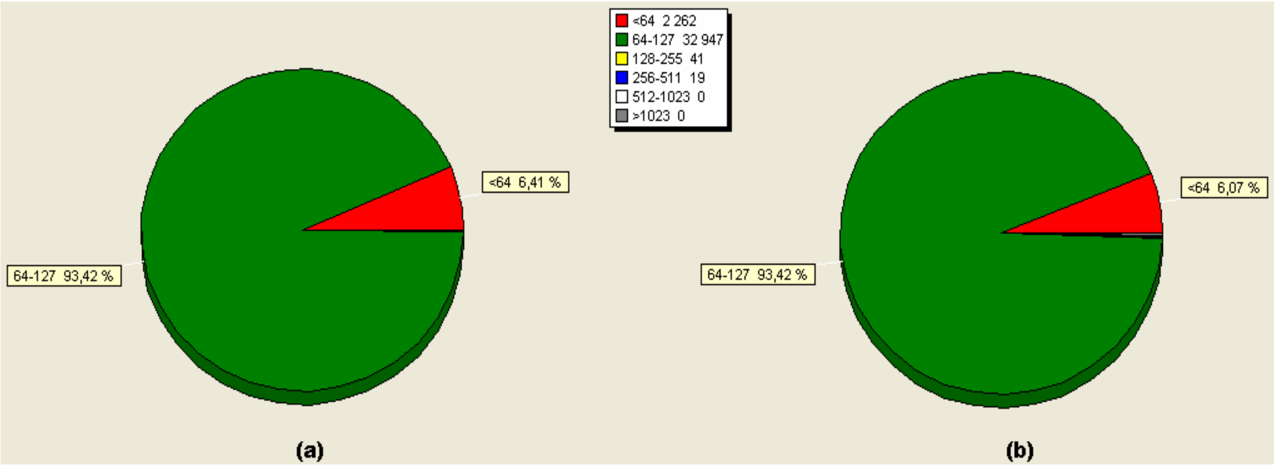

Figure 26: *Packet sizes for input output traffic of a Starcraft game session played by 6 persons over (a) cable, (b) UMTS/WCDMA.*

As can bee seen in Figure 28 (b) Warcraft 3 produces different packet sizes depending on the link. An increase in latency tends to create bigger packets. Figure 29 and Figure 30 shows in detail how the packet sizes changes. When Warcraft 3 is played over cable 12% of all packets are 54 bytes, 43% are 60 bytes and 23% are 63 bytes. When played over a WCDMA link most of the packets are either 54 bytes (35%) or 63 bytes (27%).

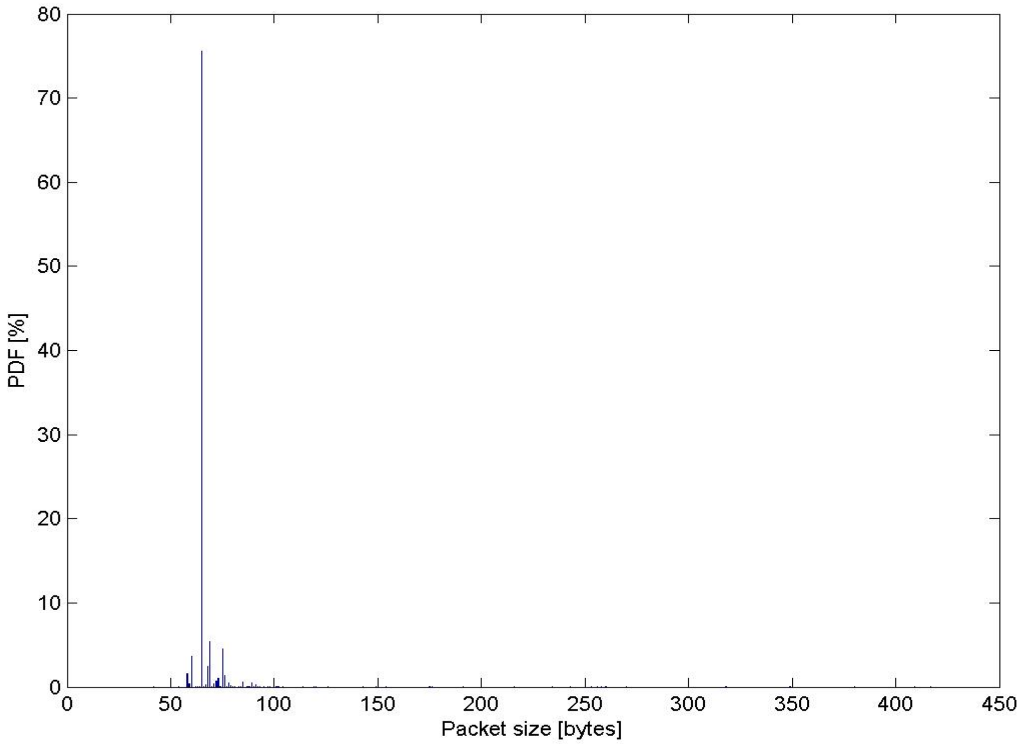

Figure 27: *Packet sizes created by a 3vs3 Starcraft game. Packet size max=417 byte, min=42 bytes, mean value =66.5 bytes, median =65 bytes.*

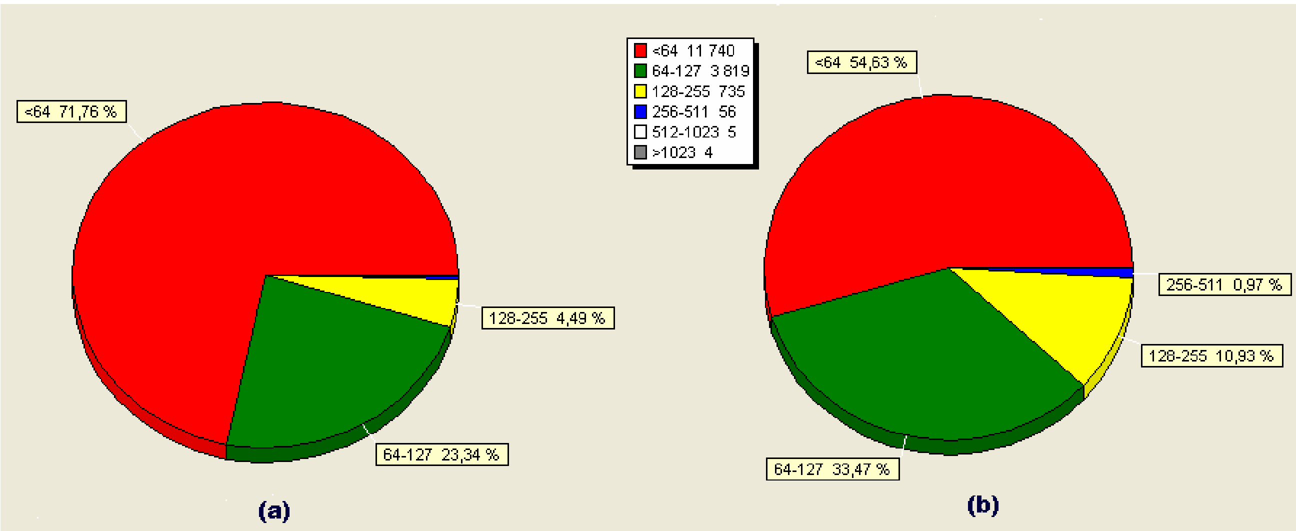

Figure 28: *Packet sizes for input output traffic of a Warcraft 3 game session played by 6 persons over (a) cable, (b) UMTS/WCDMA.*

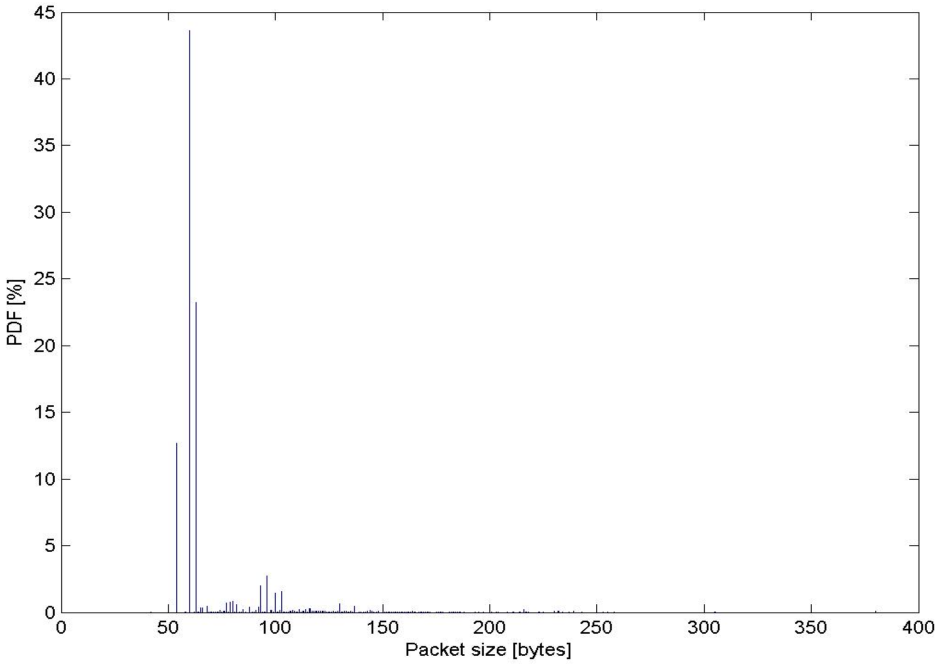

Figure 29: *PDF function of the packets sizes produced by a 3vs3 Warcraft 3 game played over cable. Packet size max=380 bytes, min=42 bytes, mean=69.8 bytes, median=60bytes.*

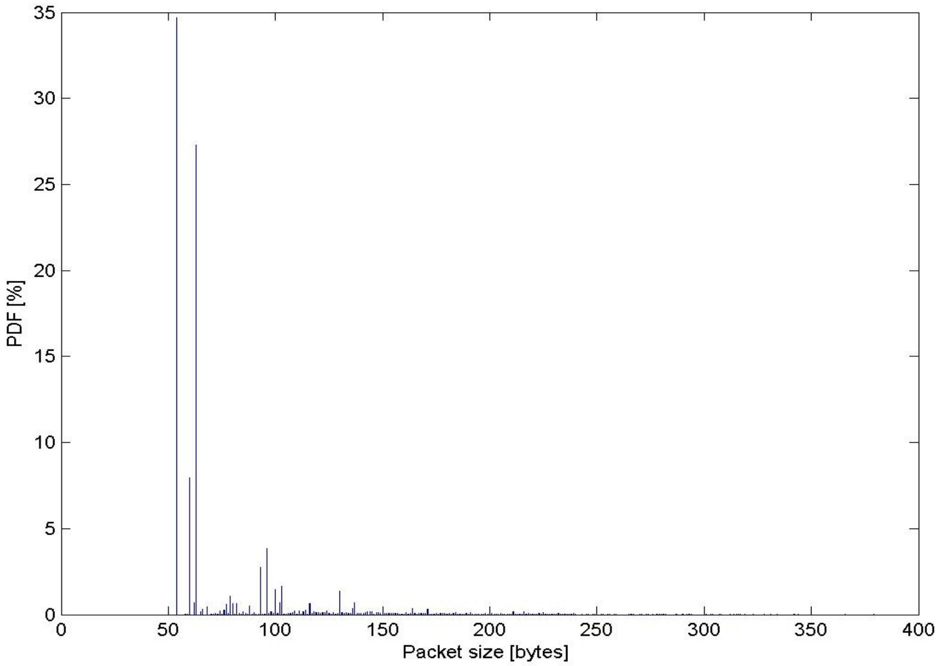

Figure 30: *PDF function of the packets sizes produced by a 3vs3 Warcraft 3 game played over a WCDMA link. Packet size max=518 bytes, min=54 bytes, mean=76.0 bytes, median=63bytes.*

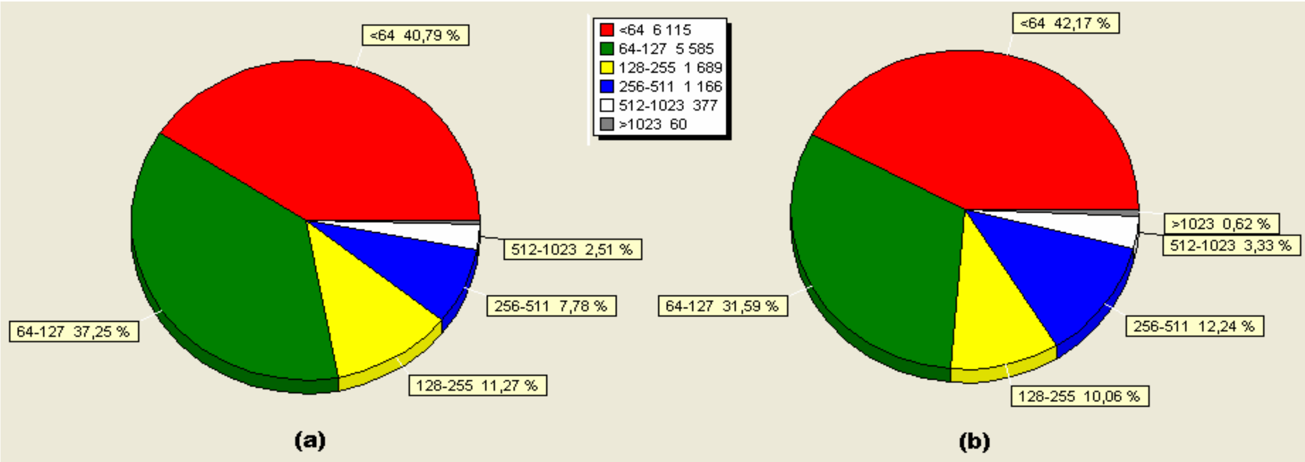

Figure 31: *Packet sizes for input output traffic of a WOW game session played over (a) cable, (b) UMTS/WCDMA.*

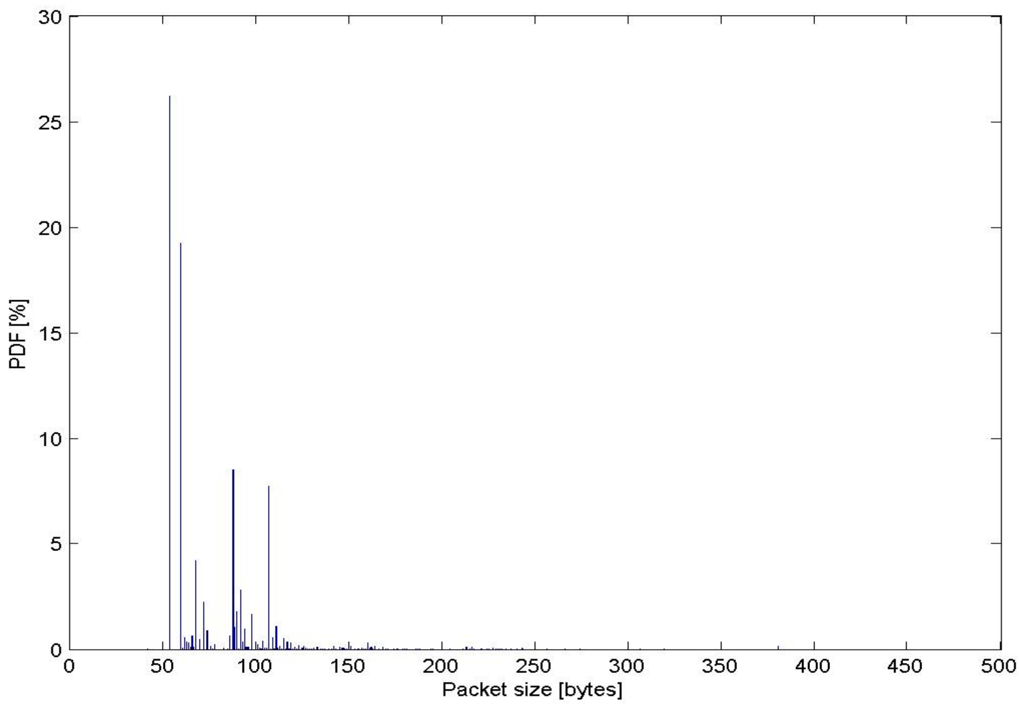

Figure 32: *PDF function of the packets sizes produced by a WOW game session played over a cable link. Packet size max=1514 bytes, min=42 bytes, mean=99.8 bytes, median=68bytes.*

WOW differs from the other games by producing a very heterogeneous palette of packets as can be seen in pie chart (Figure 31). Also here the WCDMA link has an impact on the packet sizes and can create as big packages as over 1023 bytes. Most of the packets produced while playing over a fixed connection (Figure 32) are either 54 bytes (27%) or 60 bytes (19%).

When played over a WCDMA connection 40% of the packets are 54 bytes and 28% of them are 88 bytes.

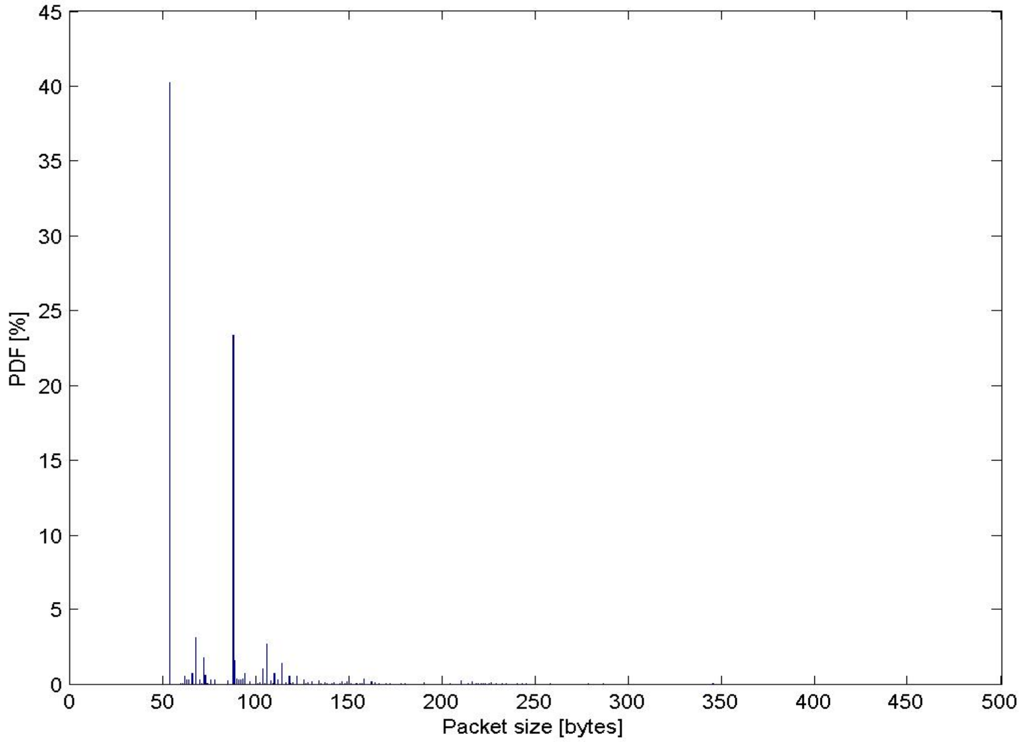

Figure 33: *PDF function of the packets sizes produced by a WOW game session played over a WCDMA link. Packet size max=1514 bytes, min=54 bytes, mean=101.0 bytes, median=88 bytes.*

6.3. Scaling

When only 4 players participate in a multiplayer session we can observe how the throughput decreases around 20% for the Starcraft game session Figure 34a. However, for the game Warcraft 3 Figure 34b there is no significant change. What happens with the Starcraft game is that it will send its packets more frequently. The interarrival time with 6 players in the gaming session is 0.019 seconds while with 4 players in the arrival time is 0.03 seconds. Due to that the packets keep the same size, this change in interarrival time has an impact on the throughput.

Traffic	Captured
Between first and last packet	949,030 sec
Packets	28722
Avg. packets/sec	30,265
Avg. packet size	66,000 bytes
Bytes	1899419
Avg. bytes/sec	2001,431
Avg. MBit/sec	0,016

(a)

Traffic	Captured
Between first and last packet	649,143 sec
Packets	8898
Avg. packets/sec	13,707
Avg. packet size	74,000 bytes
Bytes	661260
Avg. bytes/sec	1018,666
Avg. MBit/sec	0,008

(b)

Figure 34: *(a) Starcraft session 2vs2 played over fixed network - Summary by ethereal; (b) Warcraft 3session 2vs2 played over fixed network – Summary by ethereal*

6.4. Time variation

Of course the traffic produced by gaming is not constant over the day, and either over the week. Most of the gamers work or go to school during weekdays, so an assumption could be made that the traffic is the highest during the afternoons and the weekends. Video gamers survey [53] has made an investigation in this issue. They conclude that the numbers of gamers increase with almost the double during afternoon (17:00 GMT) compared with the mornings (10:00 GMT). Peak time is between 20:00 GMT and 24:00 GMT when Europe’s and North Americas evening gaming overlap. Another peak in gaming traffik is found at 5:00 to 8:00 GMT when the afternoon gaming strikes Asia, especially Korea and Japan. During the weekends the traffic increases with around 20%.

Also the traffic volume could be intresting to know about. At 17:00 GMT on a thursday 92 000 gamers plays Halflife/Counterstrike and 38 000 play Halflife 2 all popular FPS games. At the same time there are about 360 000 gamers connected to battlenet, which is the server that host Warcraft 3 and Starcraft games.

6.5. Startup traffic

Before a gaming session is started the gamer has to connect to a gaming server, where he or she meats other gamers to compete against. Usually there is a chat, and a window where you can spot which games you can join. Also there is some information about the type of gaming sessions available. Of course all this also creates traffic. The result when measured this traffic can be seen in Figure 35. It can easily be observed that also here the traffic differs among the different games. Also the characteristic of this traffic differs from the one created by the actual gaming session.

Traffic	Captured	Traffic	Captured
Between first and last packet	96,132 sec	Between first and last packet	61,183 sec
Packets	220	Packets	2096
Avg. packets/sec	2,289	Avg. packets/sec	34,258
Avg. packet size	123,000 bytes	Avg. packet size	90,000 bytes
Bytes	27110	Bytes	190219
Avg. bytes/sec	282,009	Avg. bytes/sec	3109,029
Avg. MBit/sec	0,002	Avg. MBit/sec	0,025

(a) (b)

Traffic	Captured	Traffic	Captured
Between first and last packet	44,060 sec	Between first and last packet	170,793 sec
Packets	220	Packets	562
Avg. packets/sec	4,993	Avg. packets/sec	3,291
Avg. packet size	251,000 bytes	Avg. packet size	139,000 byte
Bytes	55404	Bytes	78386
Avg. bytes/sec	1257,155	Avg. bytes/sec	458,953
Avg. MBit/sec	0,010	Avg. MBit/sec	0,004

(c) (d)

Figure 35: *Traffic for (a) Warcraft 3, (b) Starcraft, (c) WOW and (d) Worms*

6.5. Protocolls

As previous discussed multiplayer games use different protocols for communication. When taking a closer look Warcraft 3 tends to send mostly TCP protocols (94%) and also WOW prefers to send its traffic with that protocol (92%). As we previously discussed TCP needs a bigger header than for example UDP to send the same amount of data. However, the game creators understand that the throughput is no restriction when developing these games, hence using the more reliable TCP protocol.

7. Terminal

7.1. Limitations

Mobile terminals, and specifically cellular ones, put a lot of limitation to the game development. Due to that the terminals should keep their mobility, the hardware has to be limited. With limited hardware there is of course a limit in the complexity of the software that can be ran on the terminal. But there is light for the game developers. Of course there are also attempts to maximizing the hardware performance and to evolve the terminal also at a tremendous tempo, resulting in handsets with the same performance as today's laptops. This chapter will give the reader an introduction to what limits the cellular terminals put on the mobile games today.

7.1.1. Screen

Mobile phones should be, as the name indicates, mobile. Therefore they can not offer the same physical screen size as a PC or a gaming console. A typical modern mobile phone, such as the Sony Ericsson M600i provides the user with 240x320 resolution with 262,144 colors. Today's modern PCs generally deliver 1024x768 (or more) with millions of colors, so the difference is quite significant. Sadly this affects the games that you can develop for a mobile terminal. The game content has to fit the handset's screen, hence the display must be rather compact, and games that demand a lot of fine details are hard to implement. Another solution is that you have to have a projected screen - so that the device is physically small but the virtual size of the screen is large. Examples of such systems include heads-up displays, projection displays, etc.

Detailed RTS games are an example for this. They need rather high resolution, so that the gamer can easily and quickly change the view of the game area through a minimap. With poor resolution the game developer will have difficulties producing a detailed minimap. Of course the resolution also affects how detailed you major characters can be presented. A major character should not be too blocky on the screen and therefore can only takes up between 10-15% of the whole screen, which is not many pixels when you have low resolution [54].

Another drawback with mobile telephone screens are that they are square or higher than wide i.e., portrait format, which is not preferable from a gaming point of view. Consoles and PCs have a wider screen i.e., landscape format. This screen orientation is particularly important for a side-scrolling game, such as FPS, where the character has to avoid obstacles from the side. The character is usually centered in the screen and with a thinner screen the gamer will have less time to react to things that appear on the side, thus providing the landscape mode which is desirable for gaming.

Of course not everything is negative. We easily forget that not long ago we had only black and white screens on cellular handsets. Since 2002, when color displays started to appear on cell phones, we have had a significant increase in display sizes and resolution. The added colors and resolution gave the mobile games new life and gave the developer the possibility to add 3D effects, which made the console more enjoyable, rather than simply archaic time killers. The colors also boosted the mobile evolution in terms of other hardware. With more advanced screens, the phones needed more advanced hardware to cope with the extra calculations. In addition it is possible to use some color techniques to increase the perceived resolution of the screen [55].

7.1.2. Memory

Usually a mobile game needs more memory to run, than when it is stored [55]. This due to the graphic buffers and objects created at run time. Usually the available workspace is called “the heap” and its maximum size is limited in mobile devices. Lately, however, the memories in mobile phones underwent a great evolution, largely due to the fact that these phones needed to support better graphics. In 2003 mobile phones had between 200 kb-1Mb available for running application; today (2007) available memory for applications is between 9-21 Mb for the M600i while the total memory for that phone is 80 Mb. This can be compared with the memory needed to store a game, which is typically around 250 kb today [56].

Of course all this leads to richer game content, and thereby attracts more gamers. Additionally, larger memories allow the user to store more games, providing a broader choice to the consumer market. This occurs because downloaded commercial (and non-commercial) games do not have to be deleted in order to download a new game. In 2002, devices could only store two games simultaneously; today that number might be hundreds.

7.1.3. Processor

Processing power is mostly needed “pushing the pixels”, calculating changes in the game view and updating the graphical display [54]. So with more advanced screens, the phones also need more powerful processors to drive them. This limitation creates a rapid evolution of processing power, but excludes many potential game buyers. This is because older terminals may not be able to support modern games. This has not only been a problem for handheld terminals, but in the PC market the demand for more processing power has created ever more advanced game algorithms. Fortunately the mobile cellular handset processors have somewhat reduced demands, due to the mobile handset’s smaller screen and lower resolution. The M600i has a main 32 bit ARM9 processor that runs at in 203 MHz. Additionally, it has a separate graphics processor running at a 80 MHz clock frequency, which offloads the main processor. This technology enhancement is a sign that more specialized hardware solutions will come to mobile terminals, just a graphics processor and physics processor have been used in PCs.

7.1.4. Batteries

The drawback with greater processor power is that they will consume more battery power, which is an issue. A home PC or console simply plugs into an electrical outlet and the user has nearly unlimited power (perhaps being limited to a couple kilowatts - for practical purposes). Mobile devices of course do not have this option continuously because they should be mobile, hence they depend on a battery resulting in limited energy resources.

The solution to this problem is rather obvious; either you utilize batteries with greater capacity or hardware that consumes less power per function. Today we already have devices that in a very smart way dynamically adapt their power consumption and use various power reduction techniques [57]. For example a voice call does not need to consume the same amount of power as a streaming video. Sadly the electronic devices also waste a lot of power through power leakage, which is caused by transistor switching. Unfortunately, physically smaller processors, which are common in today's market, also have a negative influence due to power leakage, as with a decrease in processor size, you have an exponential increase in power leakage [57].

7.1.5. Game control

A cell phone has long been seen as just a phone and therefore only offered a 12 button display for dialing (using mostly numbers). This is a severe limitation from a gaming point of view, because the games can be hard to control. Preferable is some kind of joystick or a mouse. Lately this has come true. Many mobile phones today have a kind of mini joystick that can bee moved in 8 different directions. Also pointing devices can be utilized, such as the one found on the M600i. This is a good mobile substitute for a mouse. You use the pen, to point or drag on the screen giving input just as with a mouse. From a gaming point of view this direct coordinate entry option is good because there are many games operated by mice. Of course FPS games will not benefit from this hardware, because they are also very dependent on the mouse buttons. They However, they can utilize the mini joystick, or button improvements in general. RTS games are the opposite; generally do not need a joystick or any other buttons, but they will be very hard to play without a mouse pointer or pen.

7.1.6. Software

Current devices are dominated by two major software platforms: Qualcomm's BREW (Binary Runtime Environment for Wireless) and Java's J2ME (Java 2 Mobile Edition). Unfortunately this generate added complexity for the game developers, because they have to develop their game twice to support the different standards. Not only this, but if a game is developed in J2ME it doesn't automatically support all J2ME devices, but rather it has to be tailored for each specific hardware device. In addition, the game supports different native APIs, different graphic- and audio formats, screen sizes, and processors. All of these differences have to be taken into account for each device.

8 Market

Today Nokia is the company that has spent the most resources on the development of mobile gaming. The result of such investment was the Nokia N-gage, which it was, unfortunately, a flop. N-gage was built on a pre-existing business mobile phone platform that was enhanced for gaming purposes. Games where bought on a separate chip to facilitate game purchase and implementation. However, this product did not attract large numbers of gamers, mainly because of the high price of the games in comparison with the quality the player experienced. Nevertheless the gaming revenues are rather high and also the revenues for mobile gaming start to rise.

8.1. Gaming revenue

Despite this flop by Nokia, many research companies predict better times and a huge increase in the gaming market. However, they all have different views of size of this future market. ARC Research [58] predicts that the global gaming revenue is worth US$ 8,400M in 2008 up from US$ 1,100M of 2003, while Screen Digest [59] together with Instat predict [60] that the gaming market in 2008 alone will be worth US$ 4,200M.

If we consider only mobile gaming, Instat predicts that mobile gaming revenue will be up from US$ 204M in 2004 to US$ 1,790M in 2009 just for the US market. Telephia [61] states that 17 million North Americans downloaded a mobile game during the last three month of 2006, a huge increase by 45% from the 12 million recorded during the same period 2005. According to Screen Digest and iSuppli [62] the US market is about the same size as the European market; each making up 1/3 of the global market. ISuppli predict that the gaming market will be worth US$ 6,100M worldwide in 2010 significantly up from US$ 1,800M in 2005.

8.1.1. Billing Models

On-line purchase of mobile games has the following characteristics:

- Pay by phone bill: The mobile carriers put their customer's purchase as an item on the subscriber's phone bill, which is a form of payment highly trusted by the end users, because they do not need to give out their credit card or personal information.
- Low prices: Prices of mobile games are relative low, US$ 3-4 each, which generates impulsive consumption.
- Availability: Gamers can access the store 24/7 purchasing games wherever they are.

The way the end user is charged can vary greatly. Some of these alternatives are:

- One Time Purchase: The gamer buys the game, and can then use it as long as he or she wants.
- Time defined: The gamer pays to play the game for a certain amount of time
- Usage defined: The gamer pays to use the game a certain number of times.
- Subscription: The gamer agree to pay a recurring monthly payment to play.
- Multi pack: The gamer pays for an authorization to play a number of different games.
- Free trail: The gamer get to play a game for free, during a certain period of time or number of uses/games.
- Pay to Play: The gamer pays to play the game once.
- Micro payment: The gamer pays for broadening the game. For example unlocking new weapons by paying an additional fee.

Today in Europe the most common model is one time purchase, but the subscription based model is rather common, for example, in the US. Even though all of these different charging models are not used today, the carriers have a broad set of choices available to them in order to tailor purchases and extract the highest possible revenue out of the market.

8.2. Mobile subscribers and coverage

A major advantage of implementing a game for a cellular device is that the game developer has a huge potential market. Every cell phone is a gaming console owned by a potential customer. These customers do not need to buy a gaming console to buy a game, but rather he or she is a potential game consumer via this cellular phone. This makes the market very broad. Let us see what the statistic tell us about how many potential cellular gaming customers there are out there.

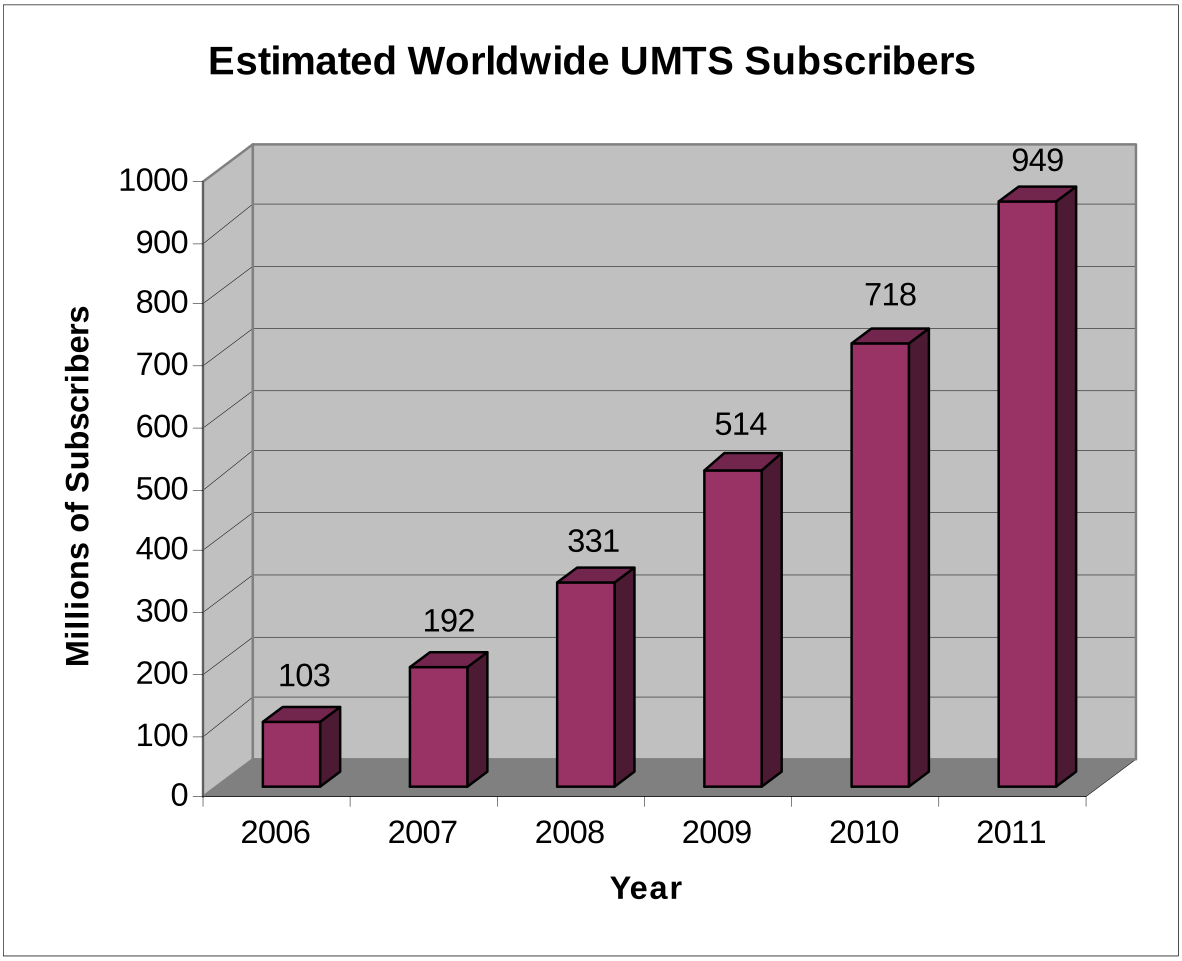

Figure 36: *Estimated millions of UMTS subscribers. Values taken from 3G America's report*

According to 3G America [29] at the end of June 2006 the number of GSM/UMTS subscribers hit 2 billion. They predict that in the year 2009 that number will be up to 3 billion and out of them 514 million will use UMTS, which as previously discussed is the target technology for mobile gaming in this thesis.

UMTS technology coverage is mostly in the western world. Japan uses only 3G and in Europe the coverage is 80% of the population [19]. Even though it has extensive coverage in Europe, UMTS only has 103 million subscribers worldwide as of 2006. However, 3G America forecasts that by 2011 UMTS will have 949 million subscribers (see Figure 36). These one billion customers will then make up a big market in terms of potential customers running around with mobile consoles ready to get loaded with games. This can be compared to the popular Microsoft Xbox 360™, which has sold a bit over 10.4 million consoles, which is a factor of 10. The difference is rather big, thus considering game developer's interest in the Xbox market, it is easy to understand that gaming for cellular phones could also represent a large market. Note however, that costumers who buy a Xbox 360 are potential better consumers for game developers than a random cellular owners; since a Xbox costumer likely will buy more games than a cell phone costumer and they have already bought a machine explicitly for gaming – unlike the purchaser of a cellular phone who is only a potential gaming customer.

8.3. Mobile gaming value chain

Gaming revenue is simply to a single party, but potentially there are a number of parties, each of which is looking for some portion of this revenue. Of course there have been a number of different clever business strategies to get rid of some of the participants in the chain. These tactics are carried out by one player to increase theire own revenue. For example, traditionally the game distribution was made via a physical medium; this was due to the large size of the games and the desire to exploit an existing sale and distribute infrastructure. Nowadays with the high penetration of broadband connections many game developers distribute their games through the internet, thereby eliminating the publisher, distributors, and retail sales entities from the value chain. The value chain for mobile gaming doesn't differ too much from the traditional game business. Never the less even if the players are similar, they are competing in a very different business. The business is different because the development is carried out on different hardware platforms, but also the gamers to attract are not similar. As I discussed earlier mobile telephones are excellent devices to attract casual gamers who wants to kill some time while waiting for a bus etc. the games should be adapted after that. Also the programming differs, needing different program knowledge, so it is rather safe to say that there is a difference in the business.

8.3.1. Game developers and porting

The game developer produces the games. He or she creates a game concept and then develops a game. As the reader should understand the game developer for mobile games does not differ significantly from a traditional game developer. However, the nature of the challenge is different. Mobile game developers not only have to have in mind creating a fun game, but they also have to adapt it to the different cellular platforms that are found on the market today, but as noted earlier this is not an easy task. Many developers are hired by a game publisher who pays them a royalty for their work, this might generate between US$ 10,000-100,000 per game [55].

Due to the complexity of the porting of the games to the cellular devices,game developers sometimes utilize a porting service. This entity ports the game code so that it works on perhaps hundreds of different cell phone handsets in the market today. Another task is to localize the game into the five primary EFIGS languages (English, French, Italian, German and Spanish). Depending on the complexity of the game, the cost for this step can be higher than initial development cost. Therefore the trend today is to utilize technologies that automate this process. The cost of the porting is usually between US$ 500 and 1,000 per device [55].

8.3.2. Game Publisher

Game publishers for the mobile market do not differ significantly from the traditional game publishers. They acquire the rights to different games, that they then try to distribute, by themselves or though a carrier. Sometimes they start with the intellectual property that they have and then broaden it by acquire new IP through though a game developer. Usually they also fund the porting process so that the game is compatible with as many cell phones as possible, so that they can achieve a greater potential sales volume. Carriers also favour games that are compatible with as many devices as possible. Usually newer devices are not a target for game publisher because they have a too small an installed base. Never the less publishers still make the game compatible with these devices, so that they can capture this market as it grows. All of these issues are factors when the carrier chooses a game publisher.

The game publisher is paid by the carrier, based on how many end users buy the game. The percentage of the retail price which the publisher gets varies a lot geographically. In Japan the publisher get 90% of the revenue, in Europe around 50%, and in the US 65%-80%. The publisher gets paid between 30-90 days after the download. The publisher has to pay the game developers royalties. This royalty is usually paid 30-45 days after the publisher is paid, so the developer has to wait 6-7 month after the first game is sold to receive any royalties.

8.3.2. Handset Manufacture

The manufacturer of the cellular terminals is comparable to the console manufacturer of the traditional gaming value chain. The manufacture adds different runtime environments to the terminal, including virtual machines. The handset manufacturer sets the standard of the mobile gaming (based upon their chaoice of screen, graphics, processor, main processor…). This manufacturer might also introduce new technology which enables the game developers to produce more advanced games. The manufacturer also sometimes plays a role as a game distributor. In some cases a preloaded demo comes with the terminal, if the end user wants the full version they have to buy it. Many manufactures such as SonyEricsson also distribute games through their internet page.

8.3.2. Mobile carriers

Mobile carriers such as Vodaphone, Orange, and others are the entities who provide the mobile network. This gives them significant power in the value chain, especially as compared to retailers in the traditional gaming value chain. Carriers often have a small monopoly over their subscribers (at least for the duration of the subscription), but their range of mobile games is usually not the main reason for their subscribers to select them as their provider. Carriers also drive pricing, technology specifications, and influence the various business models, based upon such factors as the different forms of subscriptions, prepaid, flat rate, free data usage, etc. Another powerful factor is their influence upon distribution of games to the terminals. Today a game is usually purchased by requesting a game through an SMS, then the program/game is sent to the customer through the carrier's network. After the purchase the consumer is charged by the carrier through phone bill. This shopping occurs through the carrier, thus the game publisher and developers have to understand the carriers culture to succeed with this method of distribution. Of course the game publisher can establish their own web site and sell directly to the customers for download via the customer's PC, WLAN, or via the mobile carriers network (each with potentially different costs and different prices).

Of course this dominance in the value chain causes the other players to seek ways to make the carrier less important. Such attempts include independent channel distribution, for example, through terminal manufacturer's homepages, for example, via a portal. In this method the end user downloads the mobile phone content to his or her home computer, then install it in the handheld device. These channels, however, have very limited traffic in comparison to the carriers, but there are regional differences. The European market for instance is more open to new distribution than the US market [55].

9. Conclusions

This thesis has dealt with two main what kind of traffic mobile games creates and the question of if these games are playable over today's cellular networks. We first concluded that UMTS/WCDMA is an ideal cellular technology for gaming. Nor only does it have low latency and packet loss, but also because of its rather broad coverage of the population (80% of all EU citizens). With WCDMA technology carriers have the possibility to adjust the network performance for particular clients, hence they may be able to sell more expensive subscriptions to demanding hardcore gamers. Due to this and the fact that today's WCDMA networks are empty on traffic, cellular carriers have started to offer WCDMA modems and data cards to connect to laptops, enabling mobile gamers to utilize their laptop computers for gaming. They hope then in the future when they have the market, that they can offer a special pricier gaming subscription.

However, the performance is always more limited in a mobile network, especially noticeable if the games where created for a fixed cable connection. This difference in network performance could lead to a less satisfactory gaming experience. Therefore this thesis conducted a survey, where different gamers had to grade the gaming experience limited in a controlled environment by different network performance. These results where then compared to the performance of a WCDMA link. A conclusion could be made that a WCDMA link will give a satisfied gaming experience, both for RTS and MMORPG games and hence also for the less demanding TBS genre; this even when played on a moving commuter train. This result gives hope that mobile gamers will utilize this technology for gaming and that they willl have a positive experience.

The thesis examined what traffic is produced by games from different genres. The conclusion was that the traffic differs significantly from game to game and thereby also from genre to genre. While all the games needed a throughput of less than 3.4 kbps, (which a WCDMA link easily can deliver), but that the latency produced by a WCDMA link, had an impact on the packet sizes, inter arrival times, and throughputs. Scaling of numbers of players had only an effect on Starcraft, while the traffic produced by Warcraft 3 and Worms where the same regarding the numbers of players.

9.1. Mobile Gaming evolution

Cellular phones are actually perfect for multiplayer gaming. They are the user’s private property, and they allow good communication between the gamers. But due to many factors today’s mobile gaming market only consists of downloadable single player games. The following obstacles for the mobile gaming market have to be overcome:

- Develop games with handset and network constraints in mind
- Multilanguage support and porting produces an huge extra cost (50 handsets * 5 languages = 250 portings)
- Choose the right middleware and investigating carrier restrictions
- A significant extra cost for gamers will occur if the carrier does not offer a flat rate subscription.

Luckily evolution is going the right way. Today we see smarter vector graphics that does not negatively impact hardware performance (unlike the earlier graphics). Another breakthrough for the terminals is that porting programs exist, making porting much cheaper. Carriers contribute to the evolution by offering flat rate subscriptions. Carriers initially offered this because they over dimensioned their 3G networks, thus there was little or no traffic in the network. The thought is that such abundance of high bandwidth at low latency could help gaming evolve. If mobile games become widely adopted, then perhaps the carriers will start to charge gamers a special gaming subscription price due to the extra traffic they produce; such a price would be instituted to avoid saturated their networks. The carriers also hope to attract mobile laptop gamers their WCDMA modems together with a flat rate subscription. Also the lower latency that HSDPA will offer in the future will make even the most demanding FPS games playable over a cellular network.

10. Further work

The field of this thesis has been rather broad. An examination of the performance of different cellular technologies has been presented and how these performances affect mobile gaming. Also the traffic patterns that mobile gaming produces has been presented, to give the cellular operators a better understanding of how they can adapt their systems towards gaming. However, these studies have been based on the idea that the mobile gamer plays the games on a laptop over a WCDMA link. Ericsson also has an interest in what traffic games running cellular terminals will produce. However, it is a bit trickier to access this traffic as it is embedded in the cellular terminal. The intent of this thesis was to access even this data, but the time was too short to do so. With a bit of C and Symbian knowledge access to this traffic could be achieved within 2-4 months depending on previous knowledge. To gain access to the terminal's traffic the Symbian IPHook() function can be used; however information about this application programming interface is restricted Symbian Platinum level partners. Using the IPHook function enables an application to get a time stamped copy of the packets in the phone. These packets can then be analyzed using Wireshark [63] or another packet filtering or analysis package.

References

[1] ESA [www] http://www.theesa.com/facts/gamer_data.php Last access on 8 Feb 2007

[2] CNN Money [www] http://money.cnn.com/2004/06/04/commentary/game_over/column_gaming/index.htm Last access on 8 Feb 2007

[3] GLU [www] http://www.glu.com/about/press_room.php?pryear=2005. Last access on 19 March 2007

[4] Ziff Davis Media http://gamegroup.ziffdavis.com/. Last access on 19 March 2007

[5] ESA [www] http://www.theesa.com/facts/sales_genre_data.php Last access on 8 Feb 2007

[6] Cellular online [www] http://www.cellular.co.za/gprs.htm. Last access on 12 Feb 2007

[7] Alex Spurling, *QOS Issues for Multiplayer Gaming, Case study at Cardiff university 2004* [www] http://users.cs.cf.ac.uk/O.F.Rana/data-comms/gaming.pdf Last access 3 Mai 2007

[8] Behrouz A. Forouzan, TCP/IP Protocol Suite, 3rd edition, McGraw-Hill, publication date January 2005, (Copyright 2006) 896 pages, ISBN 0072967722 (hardbound) or 0071115838 (softbound).

[9] Ericsson white paper, *WCDMA Evolved* [www] http://www.ericsson.com/technology/whitepapers/wcdma_evolved.pdf. Last access on 29 Jan 2007

[10] Tanja Lang, *User Experience while playing Halo with network delay or loss, [www] http://caia.swin.edu.au/reports/031205A/CAIA-TR-031205A.pdf. Last access 21 Feb 2007*

[11] W. J. Dally and B. Towles: *Principles and Practices of Interconnection Networks*, Morgan Kaufmann

[12] Sudhir Aggarwal et al., *Accuracy in Dead-Reckoning Based Distributed Multi-Player Games, [www] http://www.sigcomm.org/sigcomm2004/workshop_papers/net610-aggarwal.pdf. Last access 20 Feb 2007*

[13] Bu-Sung Lee et al., *Adaptive Dead Reckoning Algorithms for Distribute Interactive Simulation*, [www] http://ducati.doc.ntu.ac.uk/uksim/journal/issue-1/Lee-Turner/B-S%20Lee.pdf. Last access 20 Feb 2007

[14] Wentong Cai et al., *An Auto-adaptive Dead Reckoning Algorithm for Distrubuted Interactive Simulation,* Proceedings of the thirteenth workshop on Parallel and distributed simulation, Atlanta, Georgia, United States, 1999 pp. 82 - 89, ISBN:0-7695-0155-9. [www] http://portal.acm.org/citation.cfm?id=301429.301461 Last access 2 Mai 2007

[15] Wu-chang F Feng et.al., *Provisioning On-line Games: A Traffic Analysis of a Busy Counter-Strike Server* [www] http://www.imconf.net/imw-2002/imw2002-papers/168.pdf. Last access 11 April 2007

[16]Zsolt Kenesi, Gábor Kiss, János Levendovszky, and Sándor Molnár, *Optimizing multiplayer gaming protocols for heterogeneous network environment*, International Conference on Communications 2007 - ICC-2007, Glasgow, Scotland, UK, 24-28 June 2007. [www] See the list of Molnár's publications at: http://hsnlab.tmit.bme.hu/~molnar/abstracts.html Last access 2 Mai 2007

[17] Nathaniel E. Baughman and Brian Neil Levine, *Cheat-proof Playout for centralized and Distributed Online Games* [www] http://www.cs.ubc.ca/~krasic/cpsc538a/papers/baughman01cheatproof.pdf. Last access 21 Feb 2007

[18] Nokia white paper, *Overview of Multiplayer Mobile Game Design* [www] http://www.forum.nokia.com. Last access on 23 Feb 2007

[19] Interview with Erik Westerbarg, Ericsson AB, March 2007

[20] Sun developers [www] http://developers.sun.com/techtopics/mobility/midp/articles/gamepart1/index.html. Last access 21 Mar 2007

[21] Xu Chen at Terraplay, *Performance Analysis of Wireless Multiplayer Games on Terraplay Systems*, Masters thesis, Royal Institute of Technology (KTH), Department of Communication systems, 10 October 2005, [www] ftp://ftp.it.kth.se/Reports/DEGREE-PROJECT-REPORTS/051111-Xu_Chen-with-cover.pdf

[22] Mobile carrier Tre [www] www.tre.se. Last access on 12 Feb 2007

[23] Vesala Sami and Koivu Katja, *POSITIONING EDGE IN THE MOBILE NETWORK EVOLUTION*, Master thesis, Helsinki University of technology. [www] http://www.nectec.or.th/pressnews/bcp/Positioning_EDGE.doc Last access on 12 Feb 2007

[24] 2007Ericsson white paper, *EDGE Introduction of high-speed data in GSM/GPRS networks* [www] http://www.ericsson.com/technology/whitepapers/edge_wp_technical.pdf. Last access on 27 Jan 2007

[25] Cellular online [www] http://www.cellular.co.za/gprs.htm. Last access on 12 Feb 2007

[26] Nokia white paper, *EDGE Evolution The Revolution in end user experience* [www] http://www.nokia.com/A4126047. Last access on 27 Feb 2007

[27] Wikipedia. [www] http://sv.wikipedia.org/wiki/UMTS. Last access on 27 Jan 2007

[28] Wikipedia. [www] http://en.wikipedia.org/wiki/UMTS. Last access on 27 Jan 2007

[29] Peter Rysavy, *EDGE, HSDPA & LTE*, Rysavy Research developed for 3G America, September 2006, [www] http://www.3gamericas.com/English/Technology_Center/WhitePapers/Index.cfm. Last access 25 Feb 2007

[30] Ericsson white paper, *Basic Concepts of WCDMA Radio Access Network* [www] http://www.ericsson.com/technology/whitepapers/e207_whitepaper_ny_k1.pdf. Last access on 27 Jan 2007

[31] Nortel whitepaper, *HSDPA and beyond*, [www] http://www.nortel.com/solutions/wireless/collateral/nn_110820.01-28-05.pdf Last access on 29 Jan 2007

[32] Wikipedia. [www] http://en.wikipedia.org/wiki/Code_Division_Multiple_Access. Last access on 27 Jan 2007

[33] Wikipedia [www] http://en.wikipedia.org/wiki/IEEE_802.11. Last access on 19 Mar 2007

[34] Mail contact with Gerald Maguire, Professor KTH, March 2007

[35] 3com whitepapers, [www] http://www.3com.com/corpinfo/en_US/technology/index.jsp. Last access 19 Mar 2007

[36] Minho Shin et al, *Improving the Latency of 802.11 hand-offs using Neighbor Graphs*, [www] http://www.cs.umd.edu/~mhshin/paper/ngprobe-mobisys.pdf. Last access 19 Mar 2007

[37] Mattias Åkervik, *Network Gaming: Performance and Traffic Modeling*, Masters thesis, Royal Institute of Technology (KTH), Department of Communication systems, November 2006.

[38] Nokia white paper, *Overview of Multiplayer Mobile Game Design* [www] http://www.forum.nokia.com. Last access on 23 Feb 2007

[39] Mark Claypool, *The Effect of Latency on User Performance in Real-Time Strategy Games* (no date), [www] http://citeseer.ist.psu.edu/cache/papers/cs/31277/ftp:zSzzSzftp.cs.wpi.eduzSzpubzSztechreportszSz03-32.pdf/claypool03effect.pdf. Last access 19 Mar 2007

[40] Nokia N-gage, [www] www.n-gage.com. Last access 18 March 2007

[41] Interview with Marcus Nilsson, Ericsson AB, January 2007

[42] LBS insight [www] http://www.lbsinsight.com/?id=548. Last access on 16 Feb 2007

[43] Nokia white paper, *Multiplayer Game Performance over Cellular Networks* [www] http://www.forum.nokia.com. Last access on 23 Feb 2007

[44] Mark Claypool, *On the 802.11 turbulence of Nintendo DS and Sony PSP hand-held network games*, Proceedings of 4th ACM SIGCOMM workshop on Network and system support for games, ACM Press New York, NY, USA, 2005, pp 1 - 9, ISBN:1-59593-156-2 [www] http://web.cs.wpi.edu/~claypool/papers/hand-held/paper.pdf Last access on 16 Feb 2007

[45] Gamespot. [www] http://www.gamespot.com/. Last access on 19 Feb 2007

[46] Nathan Sheldon et al., *The effect of Latency on User Performance in Warcraft III* (no date). [www] http://web.cs.wpi.edu/~claypool/papers/war3/war3.pdf. Last access on 4 Feb 2007

[47] Wikipedia. [www] http://en.wikipedia.org/wiki/Starcraf. Last access on 4 Feb 2007

[48] Wikipedia. [www] http://en.wikipedia.org/wiki/World_of_warcraft. Last access on 16 Feb 2007

[49] Tanja Lang, *User Experience while playing Halo with network delay or loss* [www] http://caia.swin.edu.au/reports/031205A/CAIA-TR-031205A.pdf. Last access 19 Mar 2007

[50] Thomas Beigbeder et al, *The effect of Packet and Latency on Player Performance in Unreal Tournament 2003* [www] http://web.cs.wpi.edu/~claypool/mqp/ut2003/final.pdf. Last access 19 Mar 2007

[51] NetDistrub

[52] Measurement done in Ethereal

[53] Video games survey, [www] http://www.video-games-survey.com/online_gamers.htm Last access 29 Mai 2007

[54] Nokia white paper, *Designing Single-Player Mobile Games* [www] http://www.forum.nokia.com. Last access on 27 Feb 2007

[55] IDGA, *2005 Mobile Games White Paper*, [www] http://www.igda.org/. Last access 23 Mar 2007

[56] Cellular terminal vendor, SonyEricsson, {www} www.sonyericsson.com

[57] Texas Instruments White paper, *SmartReflex power and performance management technologies*, [www] http://focus.ti.com/pdfs/wtbu/smartreflex_whitepaper.pdf. Last access 23 Mar 2007

[58] ARC Research, [www] www.arcresearch.com Date of last modification 22 Feb 2007

[59] Screen digest, *Wireless gaming Business model*, Nov 2005 [www] www.screendigest.com. Last access 29 Mai 2007

[60] Instat, *Mobile Gaming Services in the US 2004-2009*, August 2004 [www] www.instat.com. Last access 29 Mai 2007

[61] Telephia, [www] www.telphia.com Last access April 2007

[62] iSuppli, [www] http://www.isuppli.com/

[62] Wireshark, [www] http://www.wireshark.org

Apendix A: Gaming Quality Survey forms

The following two survey forms where given to the participants of the gaming quality survey. The first survey form was for Warcraft 3 while the second survey form was for World Of Warcraft. For more information about the actual survey se chapter 6.

Warcraft 3 Survey Form

How often do you play online games?				
[ ] Every day	[ ]several times a week	[ ] once a week	[ ] once a month	[ ] never

How often do you play mobile games?				
[ ] Every day	[ ]several times a week	[ ] once a week	[ ] once a month	[ ] never

What kind of gamer do you think you are?	
[ ] Casual gamer	[ ] Hardcore gamer
Have you ever played Warcraft 3 before?	
[ ] Yes	[ ] No

Session 1 – 500 ms latency and 0% packetdrop		
Between 1-5 how did you perceive the gaming experience?		
Do you have the desire to leave?	[ ] yes	[ ] no

Session 2 – 750 ms latency and 10% packetdrop		
Between 1-5 how did you perceive the gaming experience?		
Do you have the desire to leave?	[ ] yes	[ ] no

Session 3 – 1000 ms latency and 0% packetdrop	
Between 1-5 how did you perceive the gaming experience?	
Do you have the desire to leave?	[ ] yes [ ] no

Session 4 – 1250 ms latency and 0% packetdrop	
Between 1-5 how did you perceive the gaming experience?	
Do you have the desire to leave?	[ ] yes [ ] no

Session 5 – 0 ms latency and 4% packetdrop	
Between 1-5 how did you perceive the gaming experience?	
Do you have the desire to leave?	[ ] yes [ ] no

Session 6 – 0 ms latency and 8% packetdrop	
Between 1-5 how did you perceive the gaming experience?	
Do you have the desire to leave?	[ ] yes [ ] no

Session 7 – 0 ms latency and 12% packetdrop	
Between 1-5 how did you perceive the gaming experience?	
Do you have the desire to leave?	[ ] yes [ ] no

Session 8 – 500 ms latency and 4% packetdrop	
Between 1-5 how did you perceive the gaming experience?	
Do you have the desire to leave?	[ ] yes [ ] no

Session 9 – 500 ms latency and 8% packetdrop	
Between 1-5 how did you perceive the gaming experience?	
Do you have the desire to leave?	[ ] yes [ ] no

Session 10 – 750 ms latency and 6% packetdrop	
Between 1-5 how did you perceive the gaming experience?	
Do you have the desire to leave?	[ ] yes [ ] no

Session 11 – 250 ms latency and 10% packetdrop		
Between 1-5 how did you perceive the gaming experience?		
Do you have the desire to leave?	[ ] yes	[ ] no

Session 12 – 250 ms latency and 8% packetdrop		
Between 1-5 how did you perceive the gaming experience?		
Do you have the desire to leave?	[ ] yes	[ ] no

WOW Survey Form

How often do you play online games?				
[ ] Every day	[ ]several times a week	[ ] once a week	[ ] once a month	[ ] never

How often do you play mobile games?				
[ ] Every day	[ ]several times a week	[ ] once a week	[ ] once a month	[ ] never

What kind of gamer do you think you are?	
[ ] Casual gamer	[ ] Hardcore gamer

Session 1 – 400 ms latency and 0% packetdrop		
Between 1-5 how did you perceive the gaming experience?		
Do you have the desire to leave?	[ ] yes	[ ] no

Session 2 – 0 ms latency and 12% packetdrop	
Between 1-5 how did you perceive the gaming experience?	
Do you have the desire to leave?	[ ] yes [ ] no

Session 3 – 800 ms latency and 0% packetdrop	
Between 1-5 how did you perceive the gaming experience?	
Do you have the desire to leave?	[ ] yes [ ] no

Session 4 – 0 ms latency and 6% packetdrop	
Between 1-5 how did you perceive the gaming experience?	
Do you have the desire to leave?	[ ] yes [ ] no

Session 5 – 0 ms latency and 4% packetdrop	
Between 1-5 how did you perceive the gaming experience?	
Do you have the desire to leave?	[ ] yes [ ] no

Session 6 – 400 ms latency and 8% packetdrop	
Between 1-5 how did you perceive the gaming experience?	
Do you have the desire to leave?	[ ] yes [ ] no

Session 7 – 0 ms latency and 10% packetdrop	
Between 1-5 how did you perceive the gaming experience?	
Do you have the desire to leave?	[ ] yes [ ] no

Session 8 – 600 ms latency and 4% packetdrop	
Between 1-5 how did you perceive the gaming experience?	
Do you have the desire to leave?	[ ] yes [ ] no

Session 9 – 600 ms latency and 8% packetdrop	
Between 1-5 how did you perceive the gaming experience?	
Do you have the desire to leave?	[ ] yes [ ] no

Session 10 – 600 ms latency and 6% packetdrop	
Between 1-5 how did you perceive the gaming experience?	
Do you have the desire to leave?	[ ] yes [ ] no

Session 11 – 800 ms latency and 6% packetdrop	
Between 1-5 how did you perceive the gaming experience?	
Do you have the desire to leave?	[ ] yes [ ] no

Session 12 – 800 ms latency and 4% packetdrop	
Between 1-5 how did you perceive the gaming experience?	
Do you have the desire to leave?	[ ] yes [ ] no

Session 13 – 0 ms latency and 8% packetdrop	
Between 1-5 how did you perceive the gaming experience?	
Do you have the desire to leave?	[ ] yes [ ] no

Session 14 – 600 ms latency and 0% packetdrop	
Between 1-5 how did you perceive the gaming experience?	
Do you have the desire to leave?	[ ] yes [ ] no

Session 15 – 400 ms latency and 6% packetdrop	
Between 1-5 how did you perceive the gaming experience?	
Do you have the desire to leave?	[ ] yes [ ] no

Session 16 – 500 ms latency and 0% packetdrop	
Between 1-5 how did you perceive the gaming experience?	
Do you have the desire to leave?	[ ] yes [ ] no

Session 17 – 400 ms latency and 4% packetdrop	
Between 1-5 how did you perceive the gaming experience?	
Do you have the desire to leave?	[ ] yes [ ] no

Apendix B: Score given by gamers

Below you can find the raw data that the figures for the gaming quality are built on.

Warcraft 3 scores

Latency	Packet loss	Gamer 1	Gamer 2	Gamer 3	Gamer 4	Gamer 5	Gamer 6	Gamer 7	Gamer 8	Gamer 9	Average Score	Comment
500	0%	5	5	4	4	5	5	5	5	5	4,777778	Good Quality
750	0%	4	5	5	4	4	5	5	4	5	4,555556	Good Quality
1000	0%	4	3	2	3	3	3	3	4	2	3	Acceptable Quality
1250	0%	3	3	2	2	2	2	2	2	2	2,222222	Not Acceptable
0	4%	4	4	5	5	4	4	4	4	4	4,222222	Good Quality
0	8%	4	4	4	4	4	4	4	4	4	4	Good Quality
0	12%	4	4	3	3	3	3	3	3	3	3,222222	Acceptable Quality
250	8%	3	4	4	4	3	4	4	4	4	3,777778	Good Quality
250	10%	4	4	3	2	4	4	3	2	2	3,111111	Acceptable Quality
500	4%	4	4	4	3	4	4	4	4	4	3,888889	Good Quality
500	8%	3	3	3	4	3	3	4	3	3	3,222222	Acceptable Quality
750	6%	3	3	2	4	4	3	3	3	3	3,111111	Acceptable Quality
750	10%	1	2	2	1	2	2	2	2	2	1,777778	Not Acceptable

WOW scores

Latency	Packet loss	Gamer 1	Gamer 2	Gamer 3	Gamer 4	Gamer 5	Gamer 6	Gamer 7	Gamer 8	Average Score	Comment
400	0%	5	5	5	5	4	5	5	5	4,875	Good Quality
500	0%	4	4	5	5	5	4	4	4	4,375	Good Quality
600	0%	4	3	3	3	4	4	4	3	3,5	Good Quality
700	0%	3	4	3	3	2	3	3	3	3	Acceptable Quality
800	0%	4	2	3	3	3	3	2	2	2,75	Acceptable Quality
0	4%	5	5	5	5	4	5	5	5	4,875	Good Quality
0	6%	5	5	4	5	5	5	5	4	4,75	Good Quality
0	8%	4	3	5	4	4	4	4	4	4	Good Quality
0	10%	4	4	2	3	3	3	2	3	3	Acceptable Quality
0	12%	3	2	3	2	2	2	2	2	2,25	Not Acceptable
200	8%	4	3	3	4	5	4	4	3	3,75	Good Quality
200	10%	4	3	3	3	2	2	2	2	2,625	Acceptable Quality
400	4%	3	4	4	4	3	3	4	4	3,625	Good Quality
400	6%	3	3	3	3	2	3	3	3	2,875	Acceptable Quality
400	8%	2	2	3	2	2	2	3	3	2,375	Acceptable Quality
600	4%	4	3	3	3	3	3	4	3	3,25	Acceptable Quality
600	6%	3	3	2	3	3	2	3	3	2,75	Acceptable Quality
600	8%	2	2	2	1	2	1	2	1	1,625	Not Acceptable
800	4%	3	3	3	3	2	2	3	2	2,625	Acceptable Quality

Apendix C: Traffic measurements

The traffic was captured in Etherreal and then exported into an excel document where the information could be sorted. To present the data in a packet distribution function (PDF) or a cumulative distribution function (CDF) Matlab was used. The sorted packet sizes where then copied into a matlab file that could plot the desired graphs. Beneth the reader can see the CDF plots for the packets sizes and the M-files that where used.

PDF M-file

```
clear

x=[
]'

last=x(length(x));

y=[1:1:last];

for i=1:last
xp(i)=0;
end
k=1;
f=0;

for j=1:last
   if x(k)-j==0
      while x(k)-j==0
      xp(j)=xp(j)+1/length(x);
      k=k+1;
      f=f+1;
      if k==length(x)+1
      k=k-1;
      x(k)=j+1;
   end
   end
   end
end

xp=xp*100;
bar(xp,0.1)
```

CDF M-file

```
clear

x=[

]';

xtemp=[1:1:length(x)];
  xs=xtemp*100/length(x);
  j=0;

for i=1:length(xs)-1
  j=j+1;

if (x(j)-x(j+1)==0)
  xs=[xs(1:j-1),xs(j+1:length(xs))];
  x=[x(1:j-1),x(j+1:length(x))];
  xs;
  j=j-1;
end
end
xs
plot(x,xs)
```

COS/CCS 2007-20

www.kth.se