Empirically Measuring the QoS Sensitivity of Interactive Online Game Players

Sebastian Zander, Grenville Armitage

Centre for Advanced Internet Architectures (CAIA)

Swinburne University of Technology

Melbourne, Australia

{szander, garmitage}@swin.edu.au

Abstract—**It is widely assumed that highly interactive online games require a minimum quality of service (QoS) from the network, which suggests that offering premium IP service quality for game players may be a new source of revenue for Internet service providers (ISPs). In order to offer such services, ISPs must know the upper bounds of the performance metrics players are willing to tolerate. This paper reports on our attempts to experimentally establish performance bounds for different types of online games. First we placed a group of players in a controlled network environment, where artificial network delay and loss is introduced during their games. We logged objective measures (performance indicators from the game server) and used a questionnaire to establish a subjective measure of user perceived quality as a function of different network conditions. Our paper concludes with an analysis of our results and a comparison with previous work based on indirect measurements.**

Keywords—Network Games, User QoS Sensitivity

I. INTRODUCTION

Recent years have seen substantial growth in the popularity of interactive network games, growth in the prevalence of game traffic on the Internet, and the emergence of network games as an important consideration from a business viewpoint [1]. Interactive game traffic has stricter quality of service (QoS) requirements than current web or email traffic. Providing premium Internet services to the growing on-line game community promises to be a potential new source of revenue for Internet Service Providers (ISPs). To adequately engineer their infrastructure for such premium services ISPs must have knowledge of the network load caused by game traffic as well as the upper bounds on performance metrics (e.g. network delay and packet loss) that game players can tolerate.

Research has already started in both areas, with most interest focused on fast paced, highly interactive 'first person shooter' (FPS) games. Traffic from a number of different popular FPS games has been characterized to provide suitable traffic models for testing existing or planned network designs (e.g. in [2], [3], [4], [5], [6] and [7]). The effect of network delay on game play has been studied in [8], [9], [10] and [11]. Previous work on network delay sensitivity has been based on indirect measurement – researchers set up public servers and then attempted to correlate observed user affinity for a particular server with observed network conditions over time. The methodology is attractive for its simplicity, but

challenging in that it is difficult to ensure a wide variety of network conditions are explored [12].

Our work addresses the empirical measurement of the QoS sensitivity of FPS players using direct measurement. A group of players is placed in a controlled network environment, where artificial network delay and packet loss is introduced during each game. We log objective measures (game performance indicators from the game server, such as 'number of kills') and use a questionnaire to establish subjective measures of user perceived quality as a function of different network conditions.

We built our first study around two different games - Quake 3 and Halo. Quake 3 is a well known and played [13] PC game designed for IP networks, with a particular goal to play well over the Internet [14]. Halo is a popular 'System Link' (LAN) game for Microsoft's Xbox console. Even though the System Link feature has been designed to work only over LANs, several Ethernet-over-IP tunneling solutions are available to connect Xboxes over the Internet [15]-[17]. In addition the launch of Xbox Live allows Xboxes to be directly connected to the Internet without the use of tunneling [18]. This variety of different solutions and the reported success of Xbox Live [19], suggest that ISPs will see a growing level of Xbox traffic.

The rest of the paper is organized as follows. Section II gives an overview on related work. Section III describes our experimental approach. Section IV presents and analyzes the results. Section V discusses how to improve the experimental methodology. Section VI concludes and outlines future work.

II. RELATED WORK

A number of papers exist on the modeling of game traffic. Early work in [2] presented a traffic model for Quake 2. A traffic model for the newer Quake 3 is proposed in [6]. The network traffic and server workload of the game Half-Life is characterized in [3] and [4], and [5] presents a traffic model for the Xbox game Halo.

In [8] the latency tolerance of Quake 3 players was empirically established to be between 150ms and 180ms. In [11] an empirical study of the user latency sensitivity for Half-Life showed players would not play when latencies are above 225-250ms. In [10] the same authors explored player tolerance to latency being added for

short periods of time during games, concluding that player tolerance rises once a player has become engrossed in a game. In [8], [10] and [11] the user sensitivity is inferred by observing the behavior (e.g. average time on the server, average kill rate) of a large number of users playing FPSs on public servers. While [8] and [11] passively analyse the user behavior in the face of uncontrolled (normal) network delay, [10] explores the affect of adding variable levels of artificial delay at the server. In [9] the effects of latency on user performance have been investigated for the Real Time Strategy (RTS) game Warcraft III. The authors find that the performance is not significantly affected by delays ranging from hundreds of milliseconds to several seconds because the nature of RTS emphasizes strategy more than highly interactive aspects.

The use of public game servers limits a researcher's ability to assess player perceived quality in the face of delay and packet loss because the players cannot be asked about their opinion. Especially it is quite challenging to correlate user perceived quality with passively inferred packet loss rates along the network paths to each client, and to the best of our knowledge this has not been attempted.

III. EXPERIMENTAL APPROACH

This section describes the methodology we used to collect the data and our testbed setup.

A. *Data Collection*

We organized a number of game sessions for both Quake 3 and Halo. For Quake 3 we had six players while for Halo we had eight players playing in four different trials. In the trials we let the users play a number of games on the same map. In each game we simulated either a constant loss or a constant delay (including zero loss and delay games). At any time the players did not know the actual loss or delay value and to avoid a possible bias the sequence was randomly chosen from a set of loss and delay values. Furthermore we asked the players to play as usual and not change their strategies between different games.

Table 1: Delay and loss values simulated

	Delays [ms]	Loss [%]
Halo	0, 50, 75, 100, 150, 200, 250	0, 1, 2, 3, 4
Quake 3	0, 50, 100, 150, 200, 250, 300	0, 5, 10, 15, 20, 25, 30, 35

Table 1 shows the different emulated delay and loss values. We chose these values after preliminary trials provided rough expectations about the players' sensitivity, and taking into account the results of previous work outlined above. We used quite different packet loss rate ranges for Quake 3 and Halo because our preliminary tests showed that Halo would simply fail to function at loss rates over 4% while Quake 3 shows no signs of quality degradation for small loss rates. Each

game was finished when a player reached the target number of 15 kills. After the game each player had to note the following statistics:

- Perceived quality from 1 to 5, where 1 means bad and 5 excellent (for calculating a mean opinion score)
- Opinion whether to continue playing under that conditions or rather choose to leave the game
- Number of kills
- Number of deaths

The first two statistics are expected to provide the user's opinion on the network quality while the last two provide information on how well they actually performed. We explicitly asked players base their perceived scores solely on their perception of their gaming experience - how it 'felt', rather than their objective kill and death scores at the end of each game. For Quake 3 the number of kills and deaths was obtained from the server log file from which we also computed the duration of the games.

In the Halo trials we also noted whether a player was playing on a client or the server (since players on the server might have a different experience of network path degradations). In fact we assumed that they even might perform better by exploiting the QoS degradation of the clients. The players were randomly rotated between games so that each player was playing some games on a client and some on the server. In our Quake 3 trials the players were all clients to a standalone server.

Our participating players were all volunteers, with no particular strategy behind their selection. We had no 'professional' players - all participants self-identified as occasional or regular players. Our method does not remove all the inter-dependencies that may influence a player's perception of game quality. Ideally we would isolate the players and have them play separately under exactly the same conditions, something that is infeasible with the existing game software. For Quake 3 the setup could be improved using computer enemies called 'bots'. However, Halo does not provide this option.

B. *Testbed Setup*

In all trials a FreeBSD PC is used as an Ethernet bridge, with FreeBSD's kernel-resident "dummynet" functionality [20] providing controllable network delay and packet loss (see Figure 1). The server is connected to one interface of the bridge with a cross-connect cable. A hub is connected to the other bridge interface and all clients are connected to the hub. In the case of Quake 3 six client machines were connected to the hub while for Halo three Xboxes were connected used by two players each. For Quake 3 we used a dedicated server (no player on the server) while in the Halo trials two players were actually playing on the server Xbox.

Although dummynet allows us to establish asymmetric delays through the bridge we configured the bridge for constant, symmetric delay in each direction. In the rest of the paper we use the actual round trip times between a client and the server, which are twice the simulated delays quoted before. Loss is introduced by

configuring dummynet to drop packets according to a uniform packet loss probability. As with delay, we used a constant, fixed loss rate in each direction.

Fixed delays and uniform packet drop probabilities are clearly not realistic models for the behavior of the Internet. However, they are appropriate for studying the impact of different levels of delay and loss on user's perceived quality. We did not emulate jitter, as other research initially suggested that public Internet jitter is typically less than one fifth the observed latencies [21].
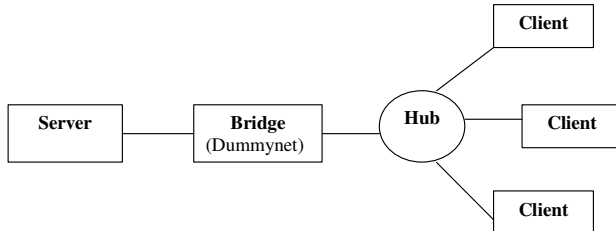
Figure 1: Testbed setup

IV. EXPERIMENTAL RESULTS

First we analyze the perceived quality for both games and different delay and loss. Then we analyze whether the loss or delay actually influenced the performance of the players. We also analyze to what extent different network qualities lead to unfairness and try to identify if different players (depending on their skills) react differently to quality degradation.

A. *Player Perceived Quality*

We compute the mean user perceived quality (mean opinion score) for the different loss and delay values (boxplots of the distributions are provided in the appendix). To verify whether there is a correlation between perceived quality and network quality we compute Pearson's product moment correlation r. The correlation can vary between +1 (perfect positive correlation) and -1 (perfect negative correlation). Values close to 0 indicate no correlation. We also compute confidence intervals and test whether a correlation is significant using a 95% confidence level.

Figure 2 shows the mean perceived quality over the different delays. As expected players on the Xbox server always experience perfect quality independent of network conditions (no significant correlation). On the other hand players on the Xbox clients notice the delay and their mean decreases with increasing delay (significant correlation r=-0.72). It can be seen that the quality is still perceived as good up to a delay of 200ms. Then the utility curve goes straight down. Interestingly the values are quite similar for Quake 3 until 300ms delay but then the slope of the utility function is less steep (signifciant correlation r=-0.56).

The smoother drop-off in the utility function for Quake 3 is likely a consequence of Quake 3 being explicitly designed to 'hide' latency – a command executed at a Quake 3 client immediately shows a result on screen using local (client-side) prediction. By contrast when a Halo client executes a command the action is shown on the screen only after a full round trip time to

the server. For small delays (typically found on LANs) the Halo algorithm works fine but makes a big difference for high delays. Quake 3 players perceive the game as still being responsive at far higher delays (although the trade-off is that some events are 'rolled back' in time if the server tells the client software that its predicted action was incorrect).
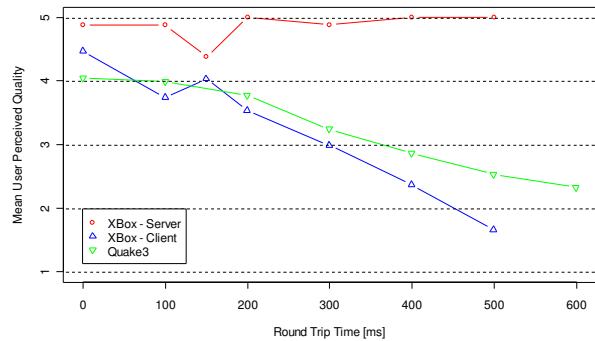

Figure 2: Mean perceived quality for different round trip times

In preliminary tests we noticed that Halo suffers from a small loss rate by briefly freezing or hanging while Quake 3 shows no degradation until we reach very high loss rates. For Halo the user perceived quality decreases straight down to 1.5 for a loss rate of 4% (significant correlation r=-0.85). With Halo the loss rate even affects players on the server, which sometimes freeze at the higher loss rates (significant correlation r=-0.61). Quake 3 players are reasonably happy even up to 35% loss rate, although there is a slight decreasing tendency (correlation r=-0.19). Quake 3's perceived performance here is also probably due to the client-side prediction algorithm working around the missing information associated with lost packets.
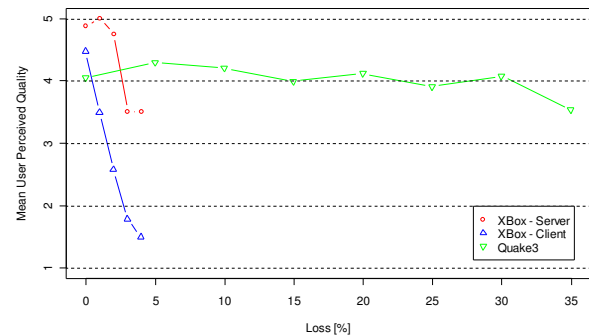

Figure 3: Mean perceived quality for different loss rates

The following two figures show the percentage of player wanting to leave the game because of the poor playing experience. The results are similar to the previous results. Xbox server players are not affected by the delay (no significant correlation) while client players are affected heavily (significant correlation r=0.64). Also similar the fraction of Quake 3 players that want to leave first increases but then stays almost constant at 400ms and above (significant correlation r=0.34). While Halo players also want to leave for high loss rates (significant correlation for server players r=0.33 and client players r=0.72) there is no such motivation for the Quake 3 players (no significant correlation).
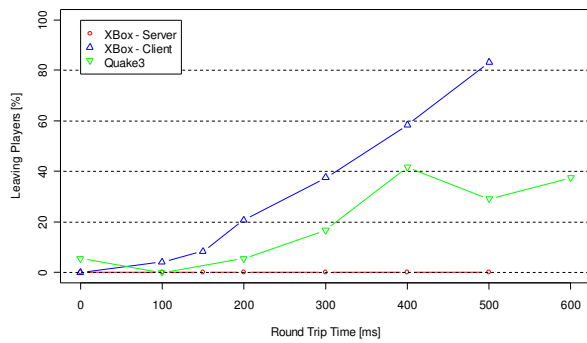
Figure 4: Percentage of players wanting to leave for different round trip times
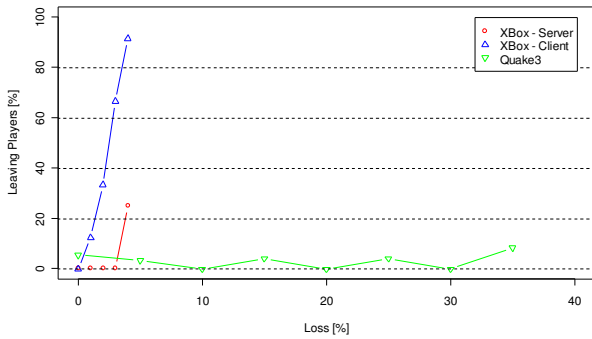

Figure 5: Percentage of players wanting to leave for different loss rates

We conclude that Xbox client players perceive high loss and delay as bad while Xbox server players only notice problems from loss. Quake 3 players perceive high delay and loss as bad, but they are substantially less affected than Xbox client players.

B. *Player Performance*

Now we investigate whether the QoS degradation, although it was perceived as negative in most cases by the players, actually had an influence on their gaming performance. Therefore we compute the mean number of kills and deaths over the different round trip times and loss rates and the correlation as previously explained.

Figure 6 shows the mean number of kills over the delay. While the number of kills is decreasing for increasing delay on the Xbox (significant correlation $r=-0.25$) there is no significant decrease for Quake 3 (actually the figure shows a slight increase).
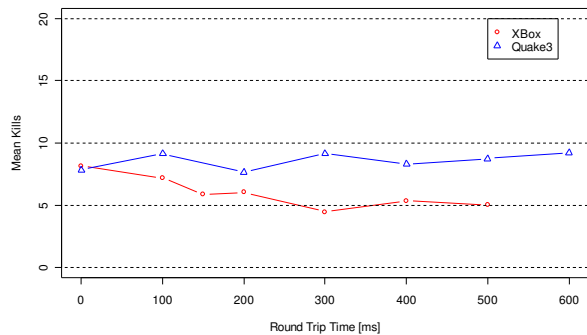
Figure 6: Mean total kills for different round trip times

The result for the players' deaths is very similar to the last figure and therefore not shown. While for Quake 3 there is no significant change, for Xbox Halo the number of deaths significantly decreases with increasing loss (significant correlation $r=-0.47$).

Figure 7 shows the number of kills over loss. It looks like for increasing loss the number of kills decrease for Halo and increase for Quake 3 but we find those trends not statistically significant. Again, the result for the deaths is very similar and not shown.
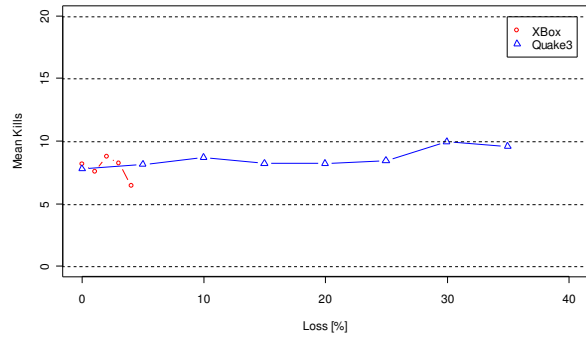

Figure 7: Mean total kills for different loss rates

The fact that the number of kills decrease for Halo and increase for Quake 3 may be surprising at first but can be explained. The number of kills increase for Quake 3 because with increasing delay and loss the games took longer which enabled the non-winning players to accumulate more kills. Figure 8 shows the mean game duration over the round trip time and loss. Again it seems that delay has a larger impact on the player's performance, because in delay games it took longer to reach the 15 kills necessary for ending a game.
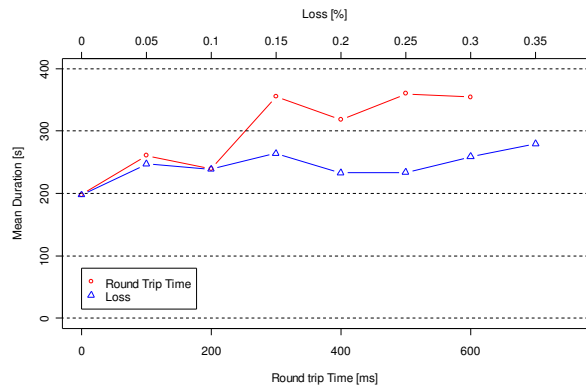

Figure 8: Mean game duration depending on round trip times and loss for Quake 3

Figure 9 shows the mean kills per minute for the winning player (reaching 15 kills first), the best 3 players and the worst 3 players. The decision of whether a player is among the best or worst players is based on the number of kills in each game. Separating the players according to their total kills (skill) shows similar but less clear trends because even a very good player sometimes has a bad game.

The graph shows that better players are more affected by the increasing delay. While the winning players drop

from almost 5 to below 3 kills/minute the number does not decrease much for the worst 3 players. Interestingly for 300ms and above the mean number of kills/minute is not significantly decreasing further although the user perceived quality is still decreasing.
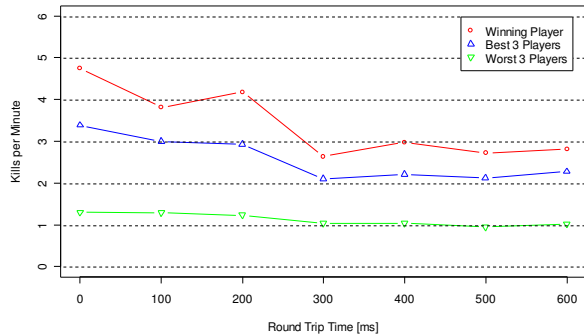

Figure 9: Mean kills per minute for different player groups over round trip time for Quake 3

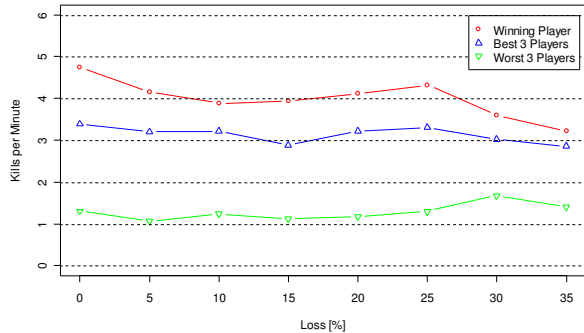Figure 10 shows similar results for loss but the impact of loss on the kill rate is smaller.

Figure 10: Mean kills per minute for different player groups over loss for Quake 3

Figure 11 shows the mean number of kills for each player group. The mean number of kills is increasing with increasing delay for the best and worst players meaning all of them getting closer to the winning player. This effect causes the overall increase of kills for increasing delays (see Figure 6). The graph for loss (not shown here) is similar and has an even higher increase towards large loss rates.
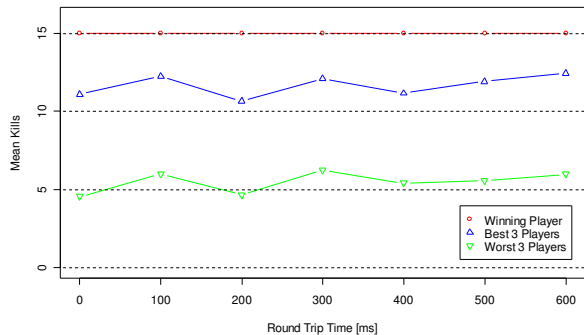

Figure 11: Mean kills for different player groups over round trip time for Quake 3

In contrast to the Quake 3 results the mean total kills

over delay clearly decrease for the Xbox because the games were not fair. The two server players who do not experienced quality degradation gained an advantage over the other players with increasing delay (see next section). For round trip times larger than 100ms 90% of the games have been won by one of the server players.

C. *Game Fairness*

For Xbox Halo we investigate the fairness of the games by comparing the kills and deaths of server-based and client-based players. We normalize the kills and deaths to the total number of kills and deaths per game to avoid any bias depending on the particular game (whether it had many or few kills/deaths).

Figure 12 shows the mean kill and death fractions for both groups. The number of kills increases for the server players while it decreases for the clients. For the maximum delay the two server players have over 60% of the total kills while the six client players together have only 40%. There is no major difference in the deaths for both groups. We believe this is because the main objective in the game is to maximize the number of kills and not to minimize the number of deaths.

We perform chi-square tests of difference between server and client players. The kill distributions are significantly different at 95% confidence level ($p < 2.2e-16$) but the death distributions are not different.
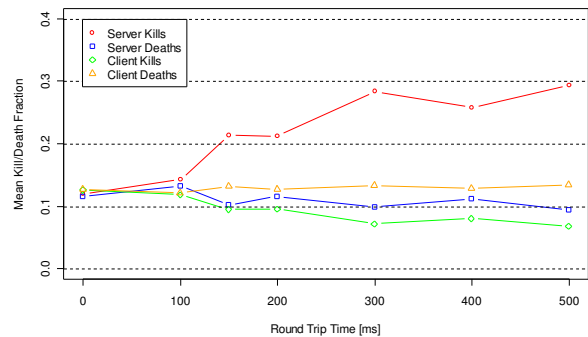

Figure 12: Fairnes for Xbox Halo for different round trip times

Figure 13 shows the kill and death fractions over loss. It can be seen that the server players are slightly better in terms of kills but the difference is much smaller than in the previous graph. There is no difference in deaths for server and client players.
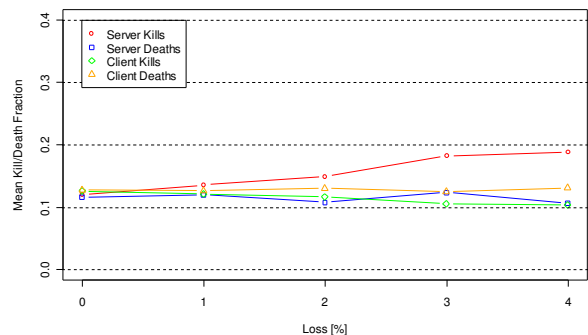

Figure 13: Fairness for Xbox Halo for different loss rates

Again we perform chi-square tests of difference and

515

find that for the kills the distributions are significantly different at 95% confidence level (p=0.0003805) but not for deaths the distributions are not different.

D. *Differences in Player Perception*

To investigate whether good players perceive quality different than bad players we use the Quake 3 data and separate the players into two groups. First we determine their 'skill' by calculating the total number of kills per player over all games and then classify the three players with the most total kills as good and the others as bad. This approach assumes that a player's skill correlates with the total number of kills and has been chosen to avoid any game-specific bias (e.g. players that achieved a high number of kills in a game may be positively biased and vice versa).

Figure 14 shows the mean perceived quality of the good and the bad players. We only show the result for the delay trials because delay has a larger impact on both the subjective and objective measurements than loss (we find no difference between both groups for the loss trials). We cannot do this analysis for Xbox Halo because players changed between different gaming sessions, which lead to an overall averaging of the total kill statistics per player (very small standard deviation). The same six players played every Quake 3 trial, and they can be easily separated into good and bad players.
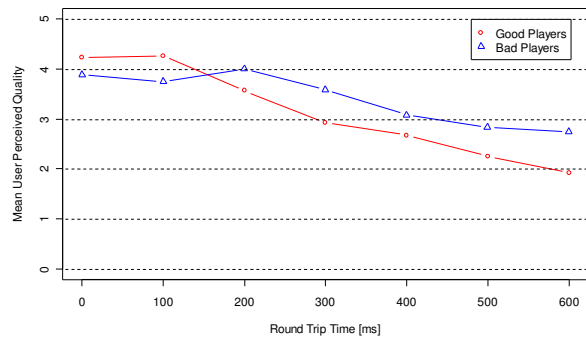

Figure 14: Mean perceived quality of bad players vs. good players for Quake 3

The figure does not show a huge difference between the two groups but bad players perceive the quality as worse at 100ms and below but better for 200ms and above. 100ms delay is hardly noticeable and so the players' perception may be biased by their actual performance. At 200ms and above the good players perceived the degrading quality as worse even though they still perform better on average. This result may be slight evidence that good (and probably more experienced) players are more sensitive to degradation in the network connection quality. However, we cannot preclude the fact that their lower perceived quality has been caused by their larger performance decrease.

E. *Comparison with Previous Work*

In [8] is is suggested that Quake 3 players prefer servers where the round trip delays are smaller than 150-180ms. In contrast [10] found that adding up to 250ms of additional delay to a Half-Life server created no strong trend of users leaving the server. Both previous papers do not deal with packet loss, and it is not clear what loss rates where being experienced by players during the time they gathered their results.

Interestingly, our Quake 3 results are broadly consistent with [8] and [11] - the number of players wanting to leave a game rose sharply to 20-40% when the latency headed into the 200-300ms region and the user perceived quality drops below average for round trip delays in the range of 300-400ms.

Work based on public servers has an important advantage over our work described here - there is a huge number of clients/players available to public servers. We used only a small number of volunteers, increasing the statistical uncertainties normally associated with human perception trials. Nevertheless the similarities between our results and public server trials provide some encouragement that trials under controlled network conditions are a valuable method of determining the latency and loss rate targets to which ISPs should aim.

V. REFINING THE EXPERIMENTAL APPROACH

There are a number of possible things that can be done to further refine our experiments. First, we should aim for a larger data set of players, games played and permutations of network conditions. Before doing a larger trial the necessary sample size should be determined. For estimating means it can be calculated using the following formula:

$$n = z_{\alpha/2}{}^2 \left(\frac{\sigma}{\varepsilon} \right)^2 \tag{1}$$

In equation (1) n is the minimum sample size, z is the z-score for a two-sided confidence interval with $1-\alpha$ confidence level, σ is the (estimated) standard deviation and ε is the error. If the minimum sample size is computed with (1) the estimated mean will be within $\pm\varepsilon$ of the true value at the given confidence level.

For example providing a proper estimation of the mean perceived quality would require 62 samples for each delay and loss value using a 95% confidence level, assuming σ=1 (estimate from the data already collected) and ε=0.25. In our study we only collected 24 samples per value, which are not even independent because in many cases the same player contributed more than one sample. However the effort for providing statistically adequate results is very high. It would require having 62 different players each playing 14 games (with different delay or loss). Where in some studies participants can fill out questionnaires quickly in 15-20 minutes for our study each one would have to spend at least 1-2 hours gaming.

It would also be instructive to add loss and/or delay to only a subset of players, in order to establish the degree to which player perceived quality depends on absolute or relative network conditions. Furthermore we should identify each player's own self-assessment of their skill level prior to each trial, and explore the relationships between their self-assessed skill level, their actual performance, and their reported perceived quality ratings [22]. This would allow a better correlation between each player's skill level and the perceived

quality. Another important information to be collected in future trials is the length of the games and the time of the day because that may have an influence on the player's perceived quality and performance. The length of the games must also be recorded to compute the kills per minute statistics. Information that is difficult to collect but also could have an influence is the personal relationship between players. Players with strong relationships to other players may tolerate a higher QoS degradation because they are more engrossed in the game.

VI. Conclusions and Future Work

In this paper we have presented an empirical study about the QoS sensivity of users playing highly interactive, network games. In contrast to previous work (which inferred the sensitivity by correlating observed user affinity for particular servers with observed network conditions over time) we conducted direct trials with a small user group. The users played in a controlled setup with emulated network delay and packet loss rates. A questionnaire was used to collect information about the perceived quality and the game performance. Despite collecting a limited amount of data our findings broadly confirm many existing assumptions:

- Different QoS clearly leads to unfairness or imbalanced games when there are no mechanisms for mitigating the QoS differences.

- Games should only be played over Internet paths if they incorporate specific mechanisms (e.g. client-side prediction) to 'hide' latency and loss. Because of their intolerance to packet loss, LAN-based games are likely to behave quite poorly if naively tunneled over the Internet. This can cause a decrease in perceived quality even if it has no influence on the player's performance.

- Player perceived quality is not the sole predictor of their likelihood of immediately leaving a game server. There appear to be other variables (unmeasured in our trials) that influence a player's decision to stay or leave.

- Delay has a larger impact on the player's perceived quality and performance than loss (for values typical occurring in the Internet).

- The influence of delay and loss on the gaming performance in terms of kills/minute depends on the player's skill. The negative impact is larger for better players than for worse players.

- Our results suggest that more successful (and presumably more experienced) players are more aware of QoS degradation than less experienced players, but that the differences are slight and not significant.

In the future we plan collecting a larger data set for providing more adequate statistical results, extending our study towards new games and different game types (e.g. car racing) and including new type of information in our study. We are also developing new trials to study the relative impact of controlled jitter [21].

References

[1] ISP planet news, http://www.isp-planet.com/news/2002/gamez_021202.html (December 2002)

[2] M.S. Borella, "Source models of network game traffic", Proceedings of networld+interop '99, Las Vegas, NV, May 1999

[3] J. Faerber, "Network game traffic modelling", Proceedings of the 1st ACM workshop on Network and System Support for games, April 2002

[4] W. Feng, F. Chang, W. Feng, J. Walpole, "Provisioning On-line Games: A Traffic Analysis of a Busy Counter-Strike Server", SIGCOMM Internet Measurement Workshop, November 2002

[5] T. Lang, G. Armitage, "A Ns2 model for the System Link game Halo", Proceedings of the Australian Telecommunications Networks & Applications Conference 2003 (ATNAC 2003), Melbourne, Australia, December 2003

[6] T.Lang, P.Branch, G.Armitage "A Synthetic Traffic Model for Quake 3", ACM SIGCHI ACE2004 conference, Singapore, June 2004

[7] T.Lang, G.Armitage, P.Branch, H-Y.Choo. "A Synthetic Traffic Model for Half Life," Proceedings of the Australian Telecommunications Networks & Applications Conference 2003 (ATNAC 2003), Melbourne, Australia, December 2003.

[8] G.Armitage, "An Experimental Estimation of Latency Sensitivity in Multiplayer Quake3", Proceedings 11th IEEE International Conference on Networks (ICON) 2003, Sydney, Australia, September 2003

[9] N. Sheldon, E. Girard, S. Borg, M. Claypool, E. Agu. "The Effect of Latency on User Performance in Warcraft III", Technical Report WPICS-TR-03-07, Computer Science Department, Worcester Polytechnic Institute, March 2003, ftp://ftp.cs.wpi.edu/pub/techreports/pdf/03-07.pdf

[10] T. Henderson, S. Bhati. Networked games — a QoS-sensitive application for QoS-insensitive users?. SIGCOMM RIPQoS Workshop 2003, Karlsruhe, Germany, August 2003.

[11] T. Henderson, "Latency and user behaviour on a multiplayer game server," Proceedings of the 3rd International Workshop on Networked Group Communications (NGC), London, UK, November.2001

[12] G.Armitage, L.Stewart "Limitations of using Real-World, Public Servers to Estimate Jitter Tolerance Of First Person Shooter Games", ACM SIGCHI ACE2004 conference, Singapore, June 2004

[13] Quake 3, http://www.idsoftware.com/ (as of July 2004)

[14] S. McCreary and k. claffy, "Trends in wide area IP traffic patterns - A view from Ames Internet Exchange", in ITC Specialist Seminar, Monterey, CA, 18-20 Sep 2000.

[15] XBConnect, http://www.xbconnect.com (as of July 2004)

[16] GameSpy tunnel support, http://www.gamespyarcade.com/support/tunnel_xbox.shtml (as of July 2004)

[17] Xbox Gateway, http://www.xboxgw.com (as of July 2004)

[18] Xbox Live, http://www.xbox.com/LIVE/default.htm (as of July 2004)

[19] Xbox Live Community, http://www.xboxlivecommunity.com/index.php (as of July 2004)

[20] Luigi Rizzo, "Dummynet: a simple approach to the evaluation of network protocols," ACM Computer Communication Review, Vol. 27, No.1, pp.31-41, January 1997

[21] G.Armitage, L.Stewart "Some Thoughts on Emulating Jitter for User Experience Trials," Proceedings of the NetGames2004 Workshop, ACM SIGCOMM2004, Portland, Oregon, USA, August 2004

[22] J. Kruger, D. Dunning "Unskilled and Unaware of It: How Difficulties in Recognizing One's Own Incompetence Lead to Inflated Self-Assessments", Journal of Personality and Social Psychology, Vol. 77, No. 6, 1121-1134, December 1999

Australian Telecommunications Networks & Applications Conference 2004 (ATNAC2004), Sydney, Australia, December 8-10 2004

517

APPENDIX

We provide boxplots for the different distributions. The bottom of a box is the 1st quartile, the line within a box is the median and the top of a box is the 3rd quartile. Whiskers extend 1.5 times the inter quartile range (IQR) and dots denote outliers.
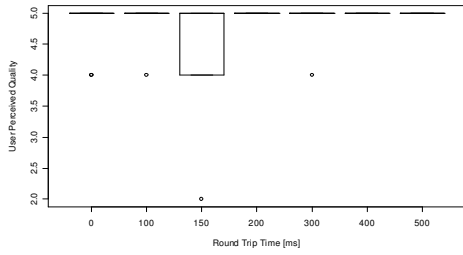

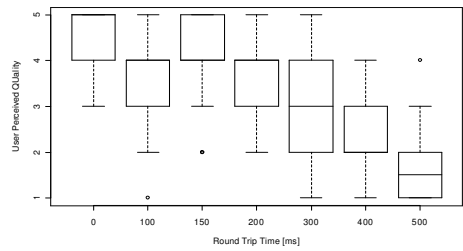Figure 15: Xbox server players perceived quality over delay

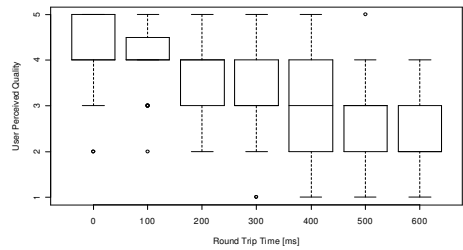Figure 16: Xbox client players perceived quality over delay
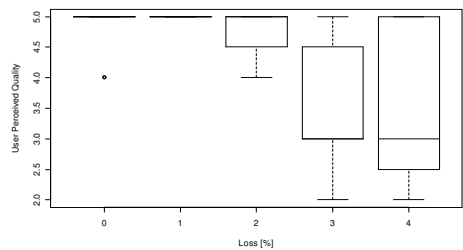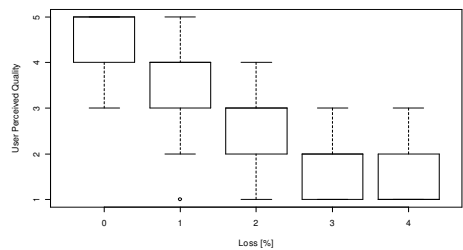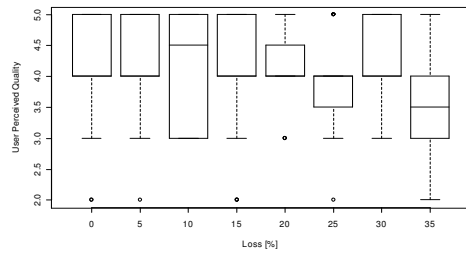

Figure 17: Quake 3 players perceived quality over delay

Figure 18: Xbox server players perceived quality over loss

Figure 19: Xbox client players perceived quality over loss

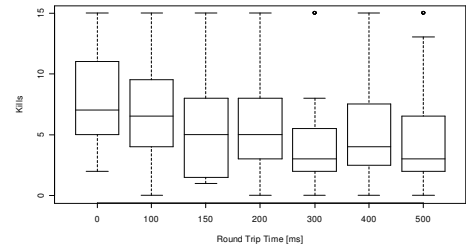Figure 20: Quake 3 players perceived quality over loss
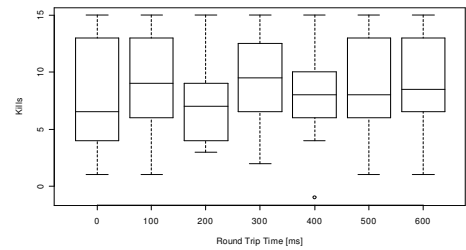

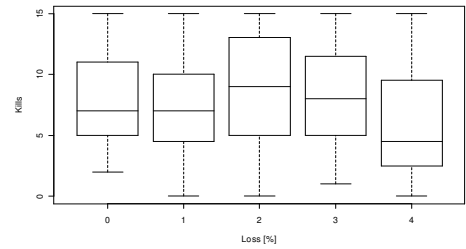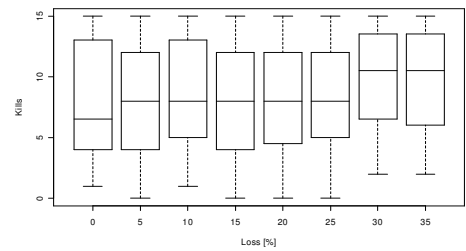Figure 21: Xbox kills over delay

Figure 22: Quake 3 kills over delay

Figure 23: Xbox kills over loss

Figure 24: Quake 3 kills over loss