

A Traffic Model for the Xbox Game Halo 2

Sebastian Zander, Grenville Armitage
Centre for Advanced Internet Architectures (CAIA)
Swinburne University of Technology
Melbourne, Australia
+61 3 9214 4835,8373
{szander,garmitage}@swin.edu.au

ABSTRACT

This paper analyses the traffic characteristics of, and proposes a traffic model for, the Xbox game Halo 2. Our goal is to help players and network providers to estimate the amount of traffic caused by the game and the impact on access links or provider networks. It also enables other researchers to use a realistic Halo 2 traffic model in network simulations. We focus on the following characteristics: bandwidth, packet rate and distribution of packet inter-arrival times and packet lengths. We compare the results with a previous analysis of Halo 1 and find some major differences – the client packet rate has been reduced, packet sizes have no longer a single fixed value per game and the mean packet size has decreased (so Halo 2 requires less bandwidth). Finally we develop traffic simulation models for Halo 2 and compare them against the experimentally obtained data.

Categories and Subject Descriptors

C.2.3 [Computer-Communication Networks]: Network Operations – *Network management, Network monitoring*

General Terms: Measurement.

Keywords: Computer Games, Traffic Model, Xbox Halo 2.

1. INTRODUCTION

Recent years have seen substantial growth in the popularity and prevalence of interactive network games and game traffic on the Internet. Such traffic has stricter quality of service (QoS) requirements than current web or email applications. Internet Service Providers (ISPs) are realising that improved support for the on-line game community can lead to better customer retention or even new revenue streams. To adequately engineer their infrastructure for such premium services ISPs must have knowledge of the network load caused by game traffic. Traffic from a number of different popular games has been characterized to provide suitable traffic models for testing existing or planned network designs (e.g. in [1], [2], [3], [4], [5], [6] and [7]).

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

NOSSDAV'05, June 13–14, 2005, Stevenson, Washington, USA.
Copyright 2005 ACM 1-58113-987-X/05/0006...\$5.00.

In this paper we investigate the traffic characteristics of Halo 2 (sequel to the popular game Halo, which we will refer to as Halo 1), compare Halo 2's characteristics with the characteristics of Halo 1 described in [4] and develop a traffic model that can be used in network simulations. Halo 1 was designed for LAN-based System Link games, and several tunnelling solutions are available to connect Xboxes over the Internet [8]-[10]. Halo 1's traffic characteristics are simple and quite different from PC games that have been developed for playing over the Internet. Halo 1 performs poorly in comparison to Quake 3 when larger network delay or packet loss occurs in the network [11]. Halo 2 was released at the end of 2004 and supports Xbox Live [12], allowing Xboxes to natively use Internet connections for multiplayer games without tunnelling. The variety of different tunnelling solutions, the reported success of Xbox Live [13] and reports of an up-tick in Xbox traffic after Halo 2's release (e.g. [14]) suggest that ISPs should pay close attention to Halo 2 traffic in their networks.

Our paper is structured as follows. Section 2 gives an overview about related work. Section 3 describes our experimental setup. Section 4 presents the traffic characteristics and compares the results with a previous analysis of Halo 1. Section 5 develops a traffic model and evaluates it by comparing it with the measured data. Section 6 concludes and outlines future work.

2. RELATED WORK

A number of papers exist on the modelling of game traffic. Early work in [1] presented a traffic model for Quake 1 and Quake 2. A traffic model for the newer Quake 3 is proposed in [5]. The network traffic and server workload of the game Half-Life is characterised in [2], [3] and [6]. The authors of [4] present a traffic model for the Xbox game Halo 1. The authors of [7] have developed models for the games Quake 2, Grand Prix 3, Ages of Empires II and Panzer General 3D.

3. EXPERIMENTAL SETUP

Several experiments were carried out to obtain data for the traffic analysis using System Link connections. The System Link feature allows up to 4 Xboxes to be linked together in a LAN, with 1 to 4 players playing on each Xbox. System Link uses standard UDP/IP/Ethernet packets to transmit data. However, the IP address and UDP port of every packet is fixed to 0.0.0.1 and 3074 respectively – only the MAC address of every Ethernet frame is used to differentiate between clients and server. We assume that Halo 2's Xbox Live traffic patterns will have essentially the same statistical properties, given that the primary functional difference (compared to System Link) is the need for valid, routable UDP/IP information in the packet headers.

We performed experiments with 3 and 4 Xboxes and a packet sniffer sharing a single Ethernet hub (Figure 1). Players fought in 6-minute, time limited deathmatch games using the map *Coagulation*. Nine games were played in total and we varied the number of players on a single client (each client is a console capable of handling between one and four players) as well as the overall total number of players (3-11).

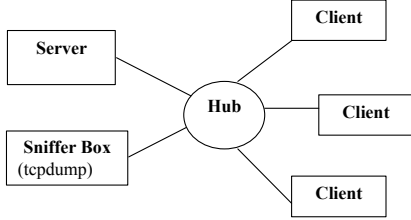


Figure 1: Experimental setup

The packet sniffer was an 800MHz Intel PC running FreeBSD 4.10 with an Intel Pro 10/100B/100+ Ethernet card. The hub was a 10Mbit/sec CentreCOM AT-MR820TR. Using a NetCom Systems Smartbits 2000 we established the sniffer’s time stamping accuracy to be low hundreds of microseconds, accurate enough for the millisecond granularity of Xbox traffic patterns.

Traffic was captured with tcpdump [15] and then post-analysed with pkthisto [16]. pkthisto creates packet length histograms, packet interarrival histograms, average bandwidth and packet per second (pps) rates for each observed flow (defined by source and destination IP addresses and port numbers) and the aggregate traffic to and from the server. For System Link frames pkthisto creates fake IP addresses from 4 bytes of each Ethernet frame’s MAC address to differentiate ‘flows’. Each histogram, pps and rate value represents 1500 consecutive packets (approximately one minute of game play between each client and server).

pkthisto reports bandwidth and packet size distributions based on the length of the IP packet. Actual link layer lengths require adding back the link layer header (e.g. another 14 bytes for Ethernet). This is reasonable since our focus is on the link-independent IP level traffic patterns.

4. HALO 2 TRAFFIC CHARACTERISTICS

In this section we characterise the Halo 2 traffic and compare our findings with the results for Halo 1 [4]. We focus on the in-game traffic, when the players actually engage in interactive play, rather than the traffic between games (start and end screen – preliminary results show that the inter-arrival time distribution is the same as during games but the packet length is fixed 52 bytes).

4.1 Packet Rate and Inter-arrival Times

First we characterise the individual connections between a single client and the server in each direction (client-server and server-client). We observed a packet transmission rate of 25 ± 0.11 pps (client-server) and 25 ± 0.26 pps (server-client). The packet inter-arrival time (IAT) is fairly constant 40 ± 0.18 ms (client-server) and 40 ± 0.43 ms (server-client). The standard deviations of the server-client traffic are higher because the server-client IAT distribution has a longer tail.

Next we characterise the aggregated traffic to and from the server. Each 40ms the server sends an update to each of the N clients back-to-back. Therefore the aggregate IAT distribution of the

server-client traffic has a peak around 0ms and a second peak at 40ms. The aggregate IAT distribution of the client-server traffic has N peaks but their locations are different in each game and we believe they depend on the timing when the clients join a game. Because of clock drift these peaks can also move over time (as observed in [4]). The mean of the IATs times of the aggregate traffic can be calculated as follows:

$$IAT_{aggr} = \left(1 - \frac{(N-1)}{N}\right) \cdot IAT_{ind} \quad (1)$$

This also means that for the server-client traffic $100(N-1)/N$ percent of the IATs are near 0ms and the rest are close to 40ms. The aggregated packet rate can also be computed based on the number of clients:

$$PPS_{aggr} = N \cdot PPS_{ind} \quad (2)$$

We verified Equation (1) and (2) with the measurement data for three clients ($N=3$). The computed and empirical found mean IAT and packet rate are the same (13ms and 75pps respectively).

Finally we analyse the IAT distributions of individual connections. Figure 2 shows the distributions of a single flow between client and server in both directions. The server-client distribution has a longer tail towards high values and the peak is smaller. These results are very similar for all flows we measured.

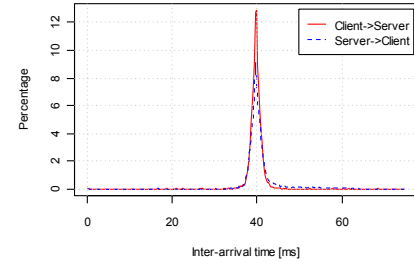


Figure 2: Inter-arrival time distributions

Figure 3 plots the contours of the distributions and shows that they are fairly stable over all one-minute histograms.

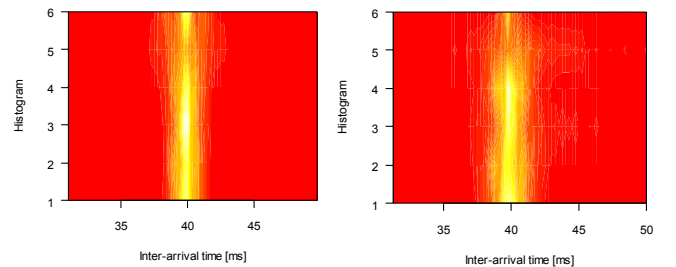


Figure 3: Inter-arrival times over different histograms for client-server (left) and server-client (right) traffic

4.2 Packet Length Distribution

The client-server packet length distributions depend on the number of players on a client (P_{client}) and the server-client packet length distributions depend on the total number of players. The relation between the client-server packet size and P_{client} is linear:

$$PS(P_{client}) = 19.04 \cdot P_{client} + 55.12 \quad (3)$$

Figure 4 shows the measured mean packet size and the estimated packet size using Equation (3). Top and bottom of the error bars

are one standard deviation away from the mean of the measured packet sizes.

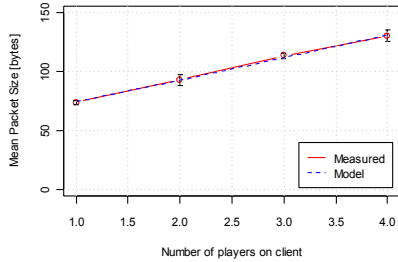


Figure 4: Measured, estimated mean client-server packet size

Figure 5 shows the mean packet size of server-client traffic depending on the total number of players. The error bars indicate the standard deviation. The packet size increases with the total number of players but there is no strict linear relationship. The standard deviation is larger than for client-server traffic.

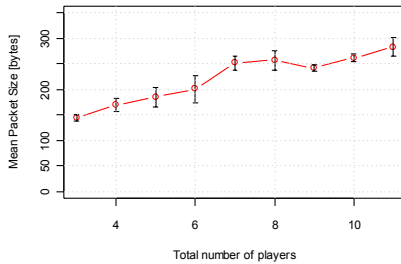


Figure 5: Mean server-client packet size

Figure 6 shows example packet length distributions for client-server traffic (3 players on the client) and server-client traffic (11 players in total). The packet length distributions are not continuous (in the packet length space) but rather discrete as certain packet sizes have peaks and packet lengths between these peaks do not occur. The characteristic packet lengths are always multiple of 4 bytes and spaced 8 bytes apart.

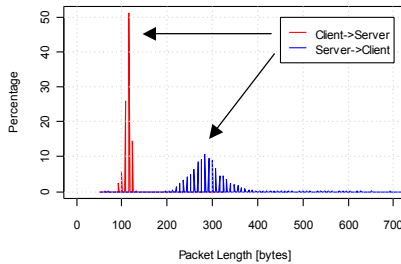


Figure 6: Packet length distributions for client-server (left) and server-client (right)

The contour plot of the packet length over time (Figure 7) shows that the distribution varies over different one-minute histograms but the most characteristic packet lengths are always present.

Figure 8 shows the envelopes of the client-server packet length distributions depending on the number of players on the client. The graph shows that the mean packet length increases and the distribution becomes wider with increasing P_{client} . Figure 9 shows the envelopes of the server-client distributions. For the sake of clarity it only contains the distributions for odd player numbers.

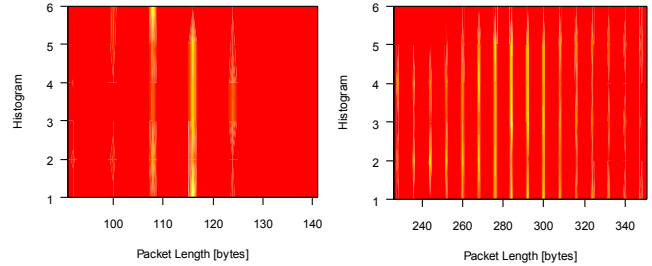


Figure 7: Packet length over different histograms for client-server (left) and server-client (right) traffic

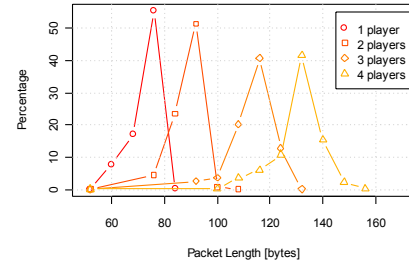


Figure 8: Packet length distributions of client-server traffic depending on the number of client players

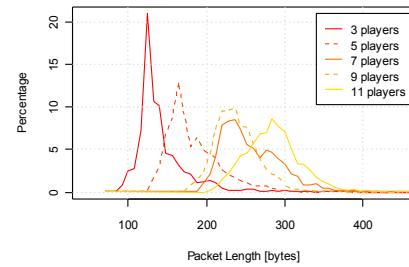


Figure 9: Packet length distributions of the server-client traffic depending on the total number of players

4.3 Bandwidth

Bandwidth consumption depends on packet rate and the packet size distribution. (As noted earlier, we report the link-independent bandwidth at the IP layer.) Figure 10 shows the bandwidth over time for all individual flows of a single game with 11 players.

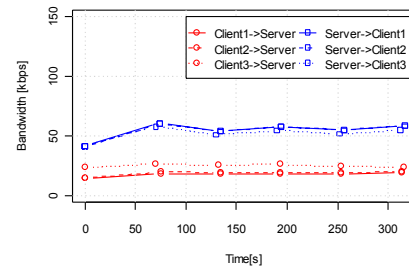


Figure 10: Bandwidth for individual flows

The last measurement interval, which includes some time in the end of game screen, has been omitted. During the game the bandwidth is fairly stable with the client-server bandwidth being more constant than the server-client bandwidth. Client 3 had four players and therefore sent larger packets than client 1 and client 2, which had two players each. Client 3 received less data from the server than the other two clients but the difference is only slight.

Figure 11 shows the aggregated bandwidth over time. The last interval with the post-game traffic has been excluded. Again the bandwidth during the game is fairly stable with the client-server traffic being more constant than the server-client traffic.



Figure 11: Aggregate bandwidth to and from server

The average client-server bandwidth linearly depends on the number of players on the client (P_{client}). Using linear regression it can be estimated to:

$$BW(P_{client}) = 3.62 \cdot P_{client} + 11.33 \quad (4)$$

Figure 12 shows the mean bandwidth of all individual client-server flows and the estimated bandwidth using Equation (4). The top and bottom of the error bars are one standard deviation away from the mean of the measured bandwidth.

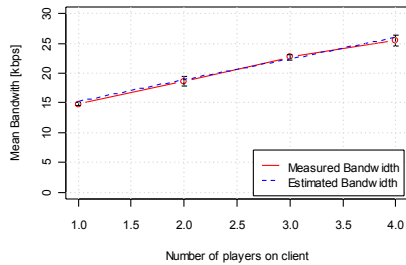


Figure 12: Mean bandwidth of individual client-server connections depending on the number of players on the client

Figure 13 shows the mean bandwidth of the aggregated server-client flows based on the total number of players in the game. The error bars show the standard deviation. In general the mean bandwidth increases with the total number of players but the increase is not strictly linear. The standard deviations are larger than for the client-server bandwidth.

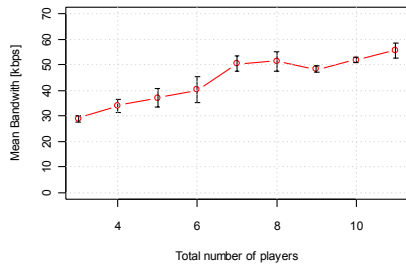


Figure 13: Mean bandwidth of individual server-client connections depending on the total number of players

4.4 Comparison with Halo 1

The server-client packet rate is 25pps for Halo 1 and Halo 2. The client-server packet rate is 25pps for Halo 2 in contrast to 30pps for Halo 1 because the 5 ‘additional packets’ sent every 201ms (as reported in [4]) have disappeared. This has also changed the

client-server IAT distribution. The packet length has decreased both for client-server and for server-client traffic. For client-server traffic the mean packet size has been reduced down to approximately 66%. The mean packet length of the server-client traffic has been reduced down to 70-80%. Therefore the required bandwidth has also been reduced by similar percentages. The packet length distribution has changed. Halo 1 only used one or two characteristic packet lengths in both directions whereas Halo 2 uses a set of different packet lengths. However, the number of characteristic packet lengths is still small (especially for client-server traffic). The mean client-server packet length still linearly depends on the number of players on a client but the mean server-client packet length no longer has a strictly linear relationship to the total number of players.

5. HALO 2 TRAFFIC MODEL

Based on the observed traffic characteristics we have developed models for the IAT and packet length distributions of Halo 2 clients and servers. We compared the experimental data with a number of theoretical distributions, especially distributions that had been used previously such as deterministic, extreme, exponential, normal and lognormal distributions. We use the same goodness of fit test as [1] and [7]. The distributions are binned using a fixed bin size of

$$w = 3.45\sigma n^{-1/3} \quad (5)$$

where σ is an estimate of the standard deviation and n the number of samples. We compute the λ^2 measure of the fit that is a non-negative value for the discrepancy between the actual and the assumed statistical model. The smaller the discrepancy, the better the model. The estimator is:

$$\hat{\lambda}^2 = \frac{\chi^2 - K - df}{n - 1} \quad (6)$$

χ^2 is the chi-square distribution, df the degrees of freedom (number of bins minus estimated parameters) and n the number of samples. K is defined as

$$K = \sum_{i=1}^N \frac{Y_i - M_i}{Y_i} \quad (7)$$

where Y_i is the number of items in bin i and M_i is the number of items in bin i of the model distribution. To visualize the quality of the fitting we use QQ plots.

Besides maximizing the goodness of fit we have also tried to select reasonably simple models and avoid over fitting. In some cases we found that mixed distributions performed slightly better but the difference was very small and therefore we selected the simpler model. In the IAT distributions we found small percentages of the samples are in extreme tails. We mostly ignored the tails and indicate the percentage of samples not captured by the model.

5.1 Packet Inter-arrival Times Model

The client-server inter-arrival times for Halo 2 are very closely distributed around the median but the distribution has tails. We find that a normal distribution provides a good fit (see Figure 14). The empirical distribution has longer tails than the normal distribution but the ends of the tails account for less than 2% of the samples.

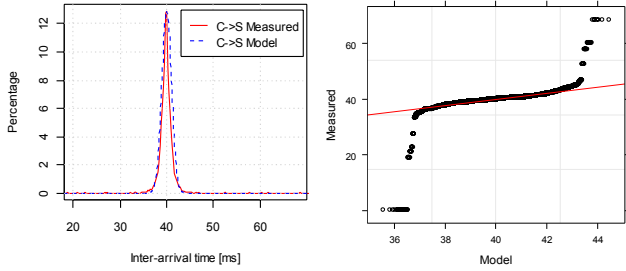


Figure 14: Client-server inter-arrival time distribution: measured data and model (left) and QQ plot (right)

The server-client IAT distribution looks very similar but has a larger tail towards high IATs. We find an extreme distribution provides the best fit (see Figure 15). This result is similar to previous findings in [1] and [2]. The QQ plot shows that the fit is very close for more than 95% of the data (except the tails).

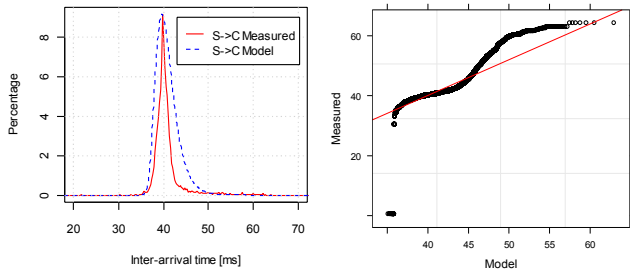


Figure 15: Server-client inter-arrival time distribution: empirical distribution and model (left) and QQ plot (right)

We have tried modelling the server-client distribution with more complex models such as a combination of different models but the goodness of fit did not improve significantly. The low λ^2 values indicate that both IAT models are fairly good (see [1] and [7]).

5.2 Packet Length Model

A difference between Halo 2 and previously modelled games is that the packet length distribution is not continuous but has characteristic peaks spaced 8 bytes apart and the intermediate packet lengths do not occur. We found that the extreme distribution provides a good fit for the envelope (similar to [1], [2] and [7]).

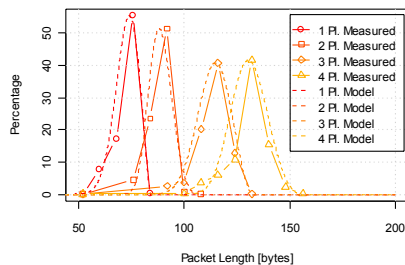


Figure 16: Client-server packet lengths measured and models

When using these models to create artificial traffic the packet lengths obtained from the extreme distribution must be rounded to the nearest packet length used by Halo 2 (see pseudo code in Section 5.3). Figure 16 shows the envelopes of the empirical client-server packet length distributions and the corresponding models (model parameters in Table 1).

Figure 17 and Figure 18 show the envelopes of the empirical server-client packet length distributions and the corresponding models (model parameters in Table 1). The graph for seven players has been omitted for improved clarity of Figure 17.

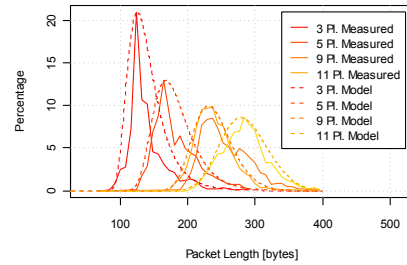


Figure 17: Server-client measured packet length and models for 3, 5, 9, 11 players

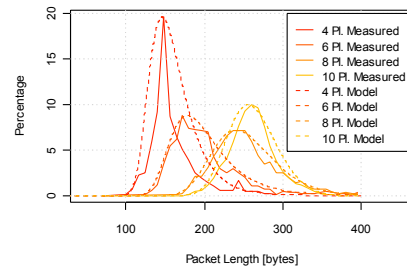


Figure 18: Server-client measured packet length and models for 4, 6, 8, 10 players

The discrete extreme distribution model fits well for all packet length distributions except client-server traffic with one player on the client. Figure 19 shows only two examples QQ plots because of space reasons.

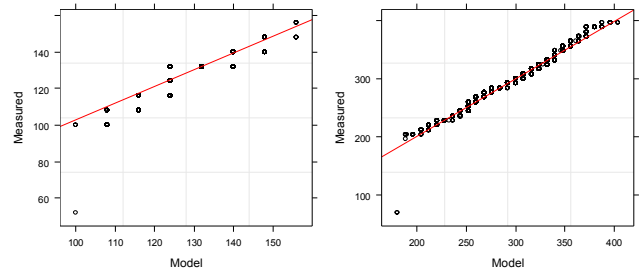


Figure 19: Packet length distribution QQ plots: client-server 4 players on client (left) and server-client 11 players total (right)

5.3 Summary

We have implemented both models, generated artificial traffic and compared the means and standard deviations of the measured and simulated traffic. The differences are always small (as expected) and for space reasons we have not included the results. Table 1 summarizes the model parameters and goodness of fit test results.

The following pseudo code shows how to implement the traffic model for a network simulator (e.g. ns-2):

```
// client to server traffic model
setMillisecondTimer(Random::normal(40,1))
...
timerExpired() {
  case numberOfPlayers == 1 :
    pktSize = Random::extreme(71.2, 5.7)
  case numberOfPlayers == 2 :
    pktSize = Random::extreme(86.9, 5.1)
```

```

...
// round to nearest 8 byte packet size
pktSize = round((pktSize-52)/8)*8 + 52
sendPkt(pktSize)
}

```

Table 1: Halo 2 traffic model parameters and results of the goodness of fit tests

	Model	Parameters	λ^2	Tail
Client-server Inter-arrival Times				
	Normal	a=40, b=1	0.19	2%
Server-client Inter-arrival Times				
	Extreme	a=39.7, b=1.9	0.51	5%
Client-server Packet Length				
1 player	Extreme	a=71.2, b=5.7	17.7	-
2 players	“	a=86.9, b=5.1	0.82	-
3 players	“	a=111.5, b=7.7	0.23	-
4 players	“	a=127.7, b=8.2	0.22	-
Server-client Packet Length				
3 players	Extreme	a=126.9, b=20.4	0.18	-
4 players	“	a=146.7, b=22.3	0.16	-
5 players	“	a=167.6, b=25.1	0.06	-
6 players	“	a=180.5, b=31.2	0.07	-
7 players	“	a=241.3, b=31.6	0.08	-
8 players	“	a=236.4, b=35.8	0.02	-
9 players	“	a=226.8, b=26.2	0.01	-
10 players	“	a=249.8, b=28.9	0.02	-
11 players	“	a=271.0, b=33.0	0.06	-

6. CONCLUSIONS AND FUTURE WORK

In this paper we have analysed the traffic characteristics of the Xbox game Halo 2 and developed a traffic model that can be used to simulate Halo 2 clients and servers. We have analysed and modelled the packet rate, bandwidth, packet length and inter-arrival times.

When comparing the results with those of earlier measurements for Halo 1 we found major differences. The client-server packet rate has been reduced from 30pps to 25pps. The mean packet length is smaller and therefore Halo 2 needs less bandwidth than Halo 1. The packet length distribution does no longer follow the simple model found for Halo 1. We found that the inter-arrival times of Halo 2 can be modelled with normal and extreme distributions. The lengths of Halo 2 packets depend on the number of players and can be modelled with extreme distributions and rounding to the nearest characteristic packet length.

We plan to collect more traffic data using different maps and different player combinations. We also plan to conduct usability trials and investigate if Halo 2 performs better in the presence of

network delay and packet loss. Furthermore, we work on simulating the impact of game traffic on typical ISP networks using this Halo 2 traffic model as well as previously developed models for other games.

7. ACKNOWLEDGMENTS

Many thanks to Chris Anthony for generously providing another monitor, David Kennedy for supplying a 4th Xbox, all the volunteer players for helping us getting the data and the anonymous reviewers for their helpful comments.

8. REFERENCES

- [1] M.S. Borella. *Source models of network game traffic*. Proceedings of network+interop '99, Las Vegas, NV, May 1999.
- [2] J. Faerber. *Network game traffic modelling*. ACM NetGames Workshop 2002, Germany, April 2002.
- [3] 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.
- [4] T. Lang, G. Armitage. *A ns-2 model for the System Link game Halo*. Australian Telecommunications Networks & Applications Conference (ATNAC) 2003, Melbourne, Australia, December 2003.
- [5] T. Lang, P. Branch, G. Armitage. *A Synthetic Traffic Model for Quake 3*. ACM SIGCHI ACE2004 conference, Singapore, June 2004.
- [6] T. Lang, G. Armitage, P. Branch, H-Y. Choo. *A Synthetic Traffic Model for Half Life*. Australian Telecommunications Networks & Applications Conference (ATNAC) 2003, Melbourne, Australia, December 2003.
- [7] J. Lakkakorpi, A. Heinerb, J. Ruutuc. *Measurement and characterization of Internet gaming traffic*. Research Seminar on Networking, Helsinki University of Technology, Networking Laboratory, Espoo, Finland, February 2002.
- [8] XBConnect, <http://www.xbconnect.com> (as of April 2005).
- [9] GameSpy tunnel support, http://www.gamespyarcade.com/support/tunnel_xbox.shtml (as of April 2005).
- [10] Xbox Gateway, <http://www.xboxgw.com> (as of April 2005).
- [11] S. Zander, G. Armitage. *Empirically Measuring the QoS Sensitivity of Interactive Online Game Players*. Australian Telecommunications Networks & Applications Conference (ATNAC) 2004, Sydney, Australia December 8-10 2004.
- [12] Xbox Live, <http://www.xbox.com/LIVE/default.htm> (as of April 2005).
- [13] Xbox Live Community, <http://www.xboxlivecommunity.com/index.php> (as of April 2005).
- [14] BBC News, <http://news.bbc.co.uk/1/hi/technology/4079397.stm>, December 2004 (as of April 2005).
- [15] tcpdump, www.tcpdump.org (as of April 2005).
- [16] pkthisto, <http://caia.swin.edu.au/genius/genius-tools.html> (as of April 2005).