Network Game Traffic Modelling

Johannes Färber

University of Stuttgart, Inst. of Communication Networks and Computer Engineering
Pfaffenwaldring 47, 70569 Stuttgart, Germany

E-mail: faerber@ind.uni-stuttgart.de

ABSTRACT

A significant share of today's Internet traffic is generated by network gaming. This kind of traffic is interesting in regard to it's market potential as well as to it's real time requirements on the network. For the consideration of game traffic in network dimensioning, traffic models are required that allow to generate a characteristic load for analytical or simulative performance evaluation of networks. In this paper we evaluate the fast action multiplayer game „Counter Strike" from a 36 hour LAN party measurement and present traffic models for client and server. The paper concludes with remarks on the use of game traffic models in simulations and on QoS metrics for an adequate evaluation of simulation results.

1. INTRODUCTION

Network game traffic generates a significant share of today's Internet traffic. In [2] it is reported that 3-4% of all packets in a backbone could be associated with only 6 popular games. A high market potential, increasing usage as well as sharp real time requirements make this kind of traffic interesting for Internet service providers and manufacturers. In order to profit from the high popularity of online gaming, networks are enhanced for gamers [7], i.e. components and protocols are optimized for game traffic. To verify the efficiency of such measures before their realization they can be described in system models which then are evaluated by analysis or simulation. In both cases traffic models are needed which impose a realistic load on the model.

In 1999 Borella presented a traffic model for the first person shooter „Quake 2" [5]. Since then many successful multiplayer games have been developed. Although there are other popular online games emerging with more focus on strategy or roleplaying, first person shooters are still the most popular multiplayer games found in the Internet and they impose the hardest real time requirements on the network. Thus, we choose to characterize the traffic patterns of „Counter Strike", a very popular first person shooter based on the Quake engine, to answer the question whether Borella's findings for „Quake 2" are still useful for newer evaluations.

In „Counter Strike" players join one of two teams and attack or defend against the other team. It is a very fast paced game where a player's „life" usually ends within few minutes. „Respawning", i.e re-entering the match with a new „life", is not allowed until the next turn with a turn lasting at most 6 minutes. The games communication model follows the client server approach and uses UDP packets for the exchange of small update information.

We have captured game traffic at a 36 hour LAN party with 50 participants. We observed several matches with 8 to 30 active players lasting 30 to 90 minutes each (6.5 hours in total). In section 1.1 we present a characterization of the patterns of client and server generated traffic with focus on packet size and packet interarrival times. In section 2 we present a traffic model framework and the corresponding parameter descriptions for client traffic and server traffic per client. We find that Borella's game traffic model is in general still valid.

1.1 Traffic Characteristics

The observed game traffic still follows the transmit cycle described in [5]: the server sends game state information to each client where packets are read and processed. Clients synchronize the server game state with their local game state, process player commands and return update packets with the players movement and status information. Since slower client machines require more processing time for rendering, their packet rate may be lower. Both, update and server information packets are usually very small since they only contain movement and status information.

Figure 1 shows a typical client traffic rate plot with almost constant behaviour. Figure 1 also shows that traffic from server to clients is more variable, but still rather smooth during a game turn. Between turns server rates may drop to zero for a short time. The observed data rate generated by the server was 16.4 kbit/s to each client and the observed data rate of client generated traffic was 15.7 kbit/s.

1.2 Server Traffic

The main characteristic of server traffic besides its slightly varying packet rate is its bursty nature. In each transmit cycle the server generates a burst of packets - one packet for every active client. Consequently, the total data rate depends on the number of active clients. Thus, it makes sense to evaluate the server traffic per client instead of it' summary traffic. This also allows to identify client specific variations.

Figure 1 shows that server traffic may stop for a short time. These stops mark pauses in the match which are due to changing a scenario or options. We do not want to describe those pauses and only

NetGames2002, April 16-17, 2002, Braunschweig, Germany.
A9 2002 ACM 1-58113-493-2/02/0004 5.00

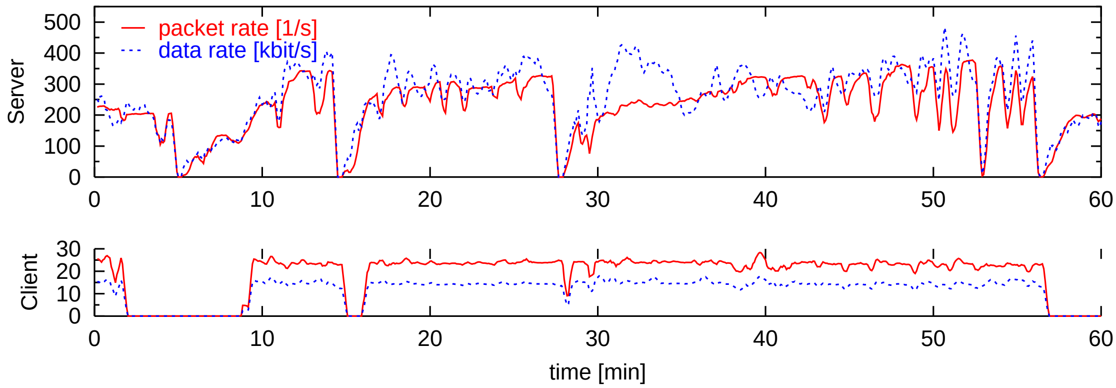

Figure 1: Example of server and typical client traffic of a 1h session

consider busy periods. This means that we have only considered interarrival times less than 1 second in the following evaluations.

Figure 2 shows the probability density function *f(x)* of the packet interarrival time from server to client. Only 8 out of 27 clients active in the first match are depicted in this figure. While 3 clients clearly have a smaller packet interarrival time (client 2 and 3 in the figure) the other 24 clients show almost exactly the same density function. An evaluation of all clients results in a peak at 55 ms and a mean of 62 ms. The coefficient of variation is around 0.5.

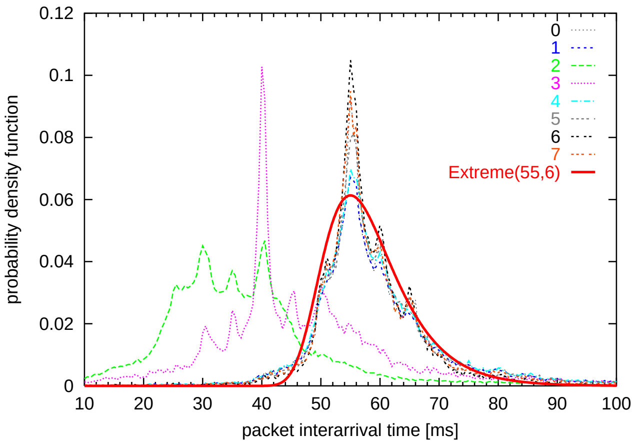

Figure 2: Probability density function of server packet interarrival time per client

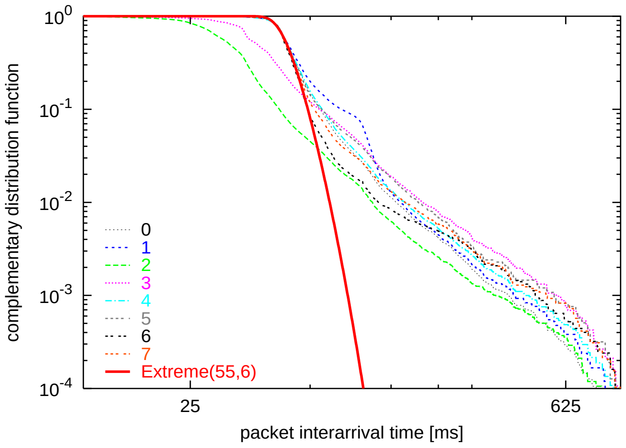

Figure 3: Complementary cumulative distribution function of server packet interarrival time per client

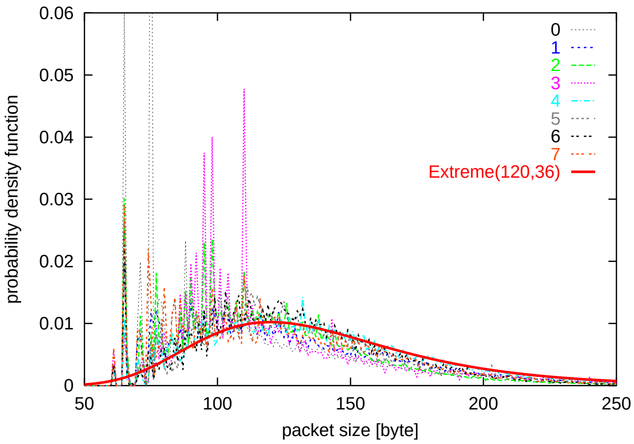

Figure 4: Probability density function of server packet size per client

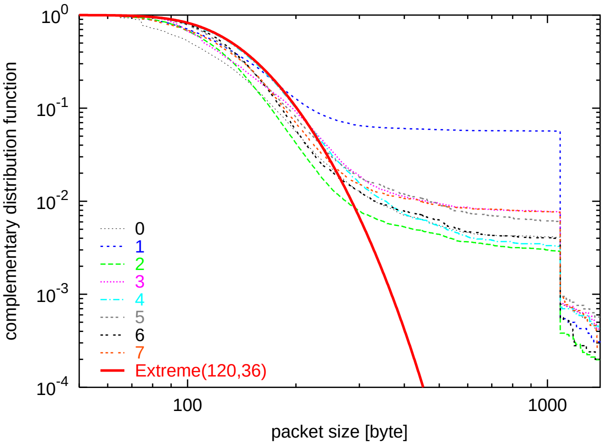

Figure 5: Complementary cumulative distribution function of server packet size per client

Figure 3 shows the complementary cumulative distribution function $F^C(x)$ of the same data. The log-log plot reveals the tail behaviour of the interarrival times: there is a significant probability for interarrival times much larger than the mean. The linear shape in the log-log plot indicates a power tail behaviour which is, however, truncated at 1 second (the Extreme function plotted here as well is discussed in section 2).

The packet size of server generated traffic shows a higher variability (see Figure 4 and Figure 5). The mean packet size was 127 Bytes with a coefficient of variation of 0.73. Around 99% of all packets were smaller than 250 Bytes and of course no packet was larger than 1500 Bytes. A significant fraction of server packets reaching a size of about 1000 Bytes may be assigned to gameplay interruptions e.g. due an end of turn or a change of scenario in which cases more information has to be transferred to the clients.

1.3 Client Traffic

Client traffic is characterized by an almost constant packet and data rate as shown in Figure 1 above. Again we evaluated the captured data for each client separately. In order to remove traffic patterns resulting from player pauses or waiting time between matches or turns we only consider packet interarrival times smaller than 1 second. In Figure 6 and Figure 7 the probability density and distribution functions for 8 of the 27 active clients in the first match are shown. We see that the packet interarrival time shows clear peaks for the clients but also that the client behaviour differs, although only within a limited range. This difference is caused by different client hardware performance and settings as Borella has shown. An evaluation of all client packets results in a mean interarrival time of 41.7 ms and a coefficient of variation of 0.24. The long tailed behaviour of the distribution function is caused by very few large interarrival times at around 600 to 800 ms.

The packet sizes of the clients vary around a mean of 82 Bytes with a coefficient of variation of 0.12. Although Figure 8 is confusing, we see that the probability density functions look similar for each client. They show a peak around 80 Bytes. The long tail shape of the distribution function (Figure 9) is caused by few packets with around 200 and around 300 Bytes. The function shows that 99% of all packets range between 60 and 110 Bytes.

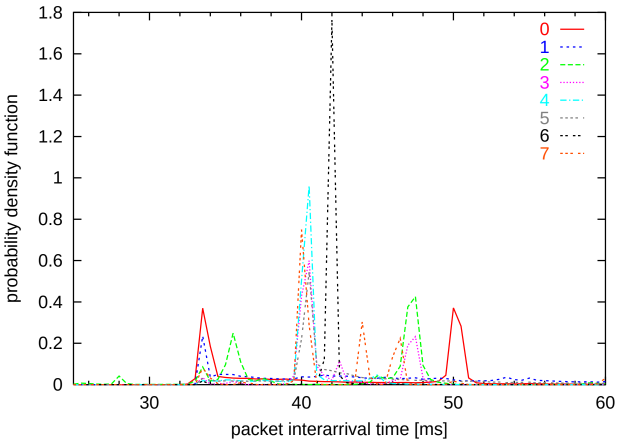

Figure 6: Probability density function of client packet interarrival time per client

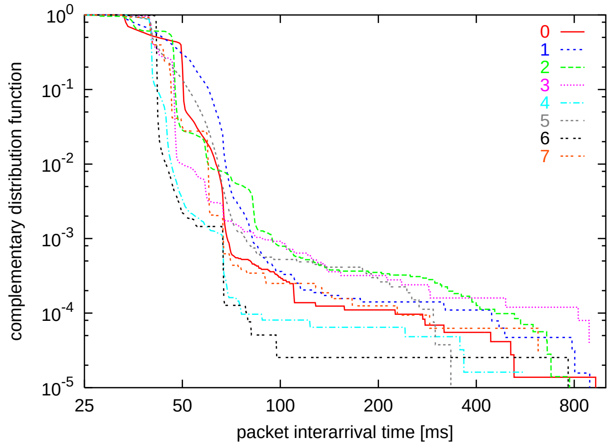

Figure 7: Complementary cumulative distribution function of client packet interarrival time per client

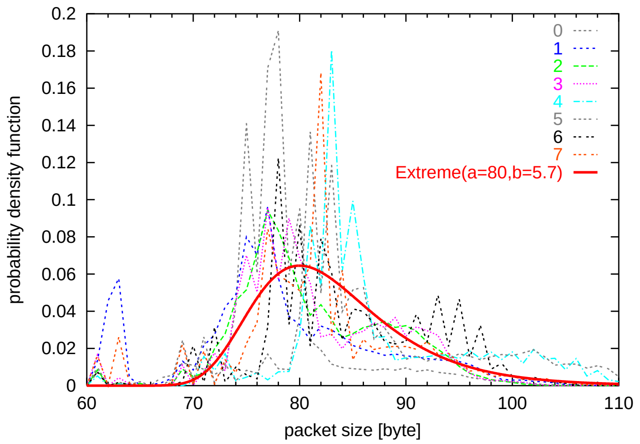

Figure 8: Probability density function of client packet size per client

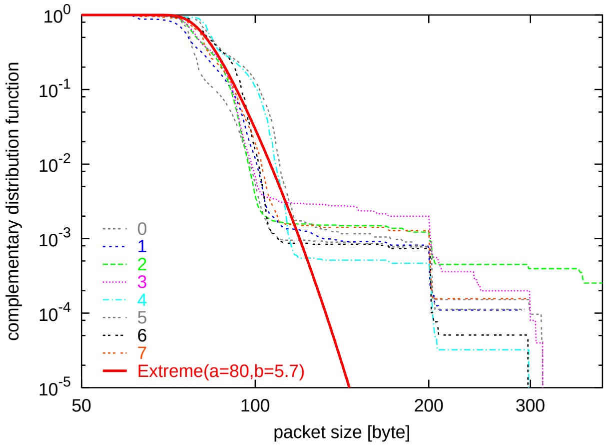

Figure 9: Complementary cumulative distribution function of client packet size per client

Table 1. Counter Strike traffic characteristics and suggested approximation

	Server (per client)		**Client**	
	characteristic	**approximation**	**characteristic**	**approximation**
(burst) interarrival time	peak = 55 ms mean = 62 ms coeff. of variation = 0.5	Extreme (a=55,b=6)	mean = 41.7 ms coeff. of variation = 0.24	Deterministic (40 ms)
packet size	mean = 127 Bytes coeff. of variation = 0.74	Extreme (a=120,b=36)	mean = 82 Bytes coeff. of variation = 0.123	Extreme (a=80,b=5.7)

2. GAME TRAFFIC MODEL

Our intention is to provide a simple traffic model for fast action multiplayer games. Although multiplayer game traffic shows strong correlations due to a shared game state we have shown in section 1.1 that the variance is small, i.e. these dependencies only lead to slight traffic changes. Thus, the game traffic can be modelled by independent traffic streams from each client to the server and a burst traffic stream from the server to the clients. In our approach we assume that (1) clients behave independent of each other, (2) server traffic per client is independent of the number of clients and (3) client traffic is independent of the corresponding server traffic.

Based on the scope of the evaluation the modelled traffic only reflects active game phases without interruptions due to change of scenario or game options. During game interruptions client and server traffic may pause for a short time after which larger update packets are transferred to synchronize all clients. Note, that this traffic is not time critical. Those dynamics are out of the scope of this work and have to be modelled on a higher level if desired.

Our game traffic model consists of only two independent modules, the client traffic model and the server traffic model with a burst size equal to the number of clients participating in the simulated traffic.

For a mathematical description of the distribution functions for interarrival time or packet size we need to find a function of similar shape and fit its parameters to the empirical data. As Borella has identified the Extreme Value distribution to fit best for Quake traffic, we also choose this function for better comparison. Similar functions as shifted Lognormal or shifted Weibull lead to acceptable fits as well.

Extreme Value distribution:

$$\boxed{F^C(x) = e^{-e^{-\frac{x-a}{b}}}} \quad \boxed{f(x) = \frac{1}{b}\, e^{-\frac{x-a}{b}}\, e^{-e^{-\frac{x-a}{b}}}} \quad b > 0$$

For finding the best parameter values for a selected function we perform a least square fitting to the probability density function. As this method neglects small probabilities especially in the tail of the distribution functions, the resulting parameters will lead to a smaller variability than actually observed. Note, that the empirical data still contains packet sizes or interarrival times captured during game interruptions. As we do not want to consider this in our description of the very regular in-game behaviour of server and clients modelling a long tailed distribution is not desired here.

2.1 Server-Model

The interarrival time for the server denotes the burst interarrival time. Within a burst a packet is sent to every client as soon as possible. Packet sizes are generated independently for each destination. Table 1 shows traffic characteristics of the observed data as well as the suggested distribution. The corresponding plots for the extreme value distribution can be found in Figure 2 to Figure 5. The neglection of the tail behaviour is clearly visible in Figure 3 and Figure 5. Extreme peaks shown in Figure 4 are client specific and are not modelled here.

For matches with a small number of players we have found that interarrival times of server bursts show four clear peaks comparable to client interarrival times, i.e. at 50 ms, 55 ms, 60 ms and 65 ms instead of a continuous distribution function as obtained for matches with many players. We assume that this behaviour is caused by the server nearing its performance limit in games with many clients.

2.2 Client-Model

As the distribution functions of client packet interarrival times is characterized by one to three peaks a multimodal distribution is suggested. Significant peaks are identified at 34 ms, 42 ms, 50 ms and 60 ms. As most observed clients show their peak at 42 ms we suggest a deterministic distribution for this interarrival time (see Table 1).

The client packet size probability function shown in Figure 8 leads to the conclusion that packet sizes are similar for all clients. Thus we fitted an extreme value function to an empirical probability function of all captured client packets.

2.3 Use of Game Traffic Model

The simplicity of the presented model allows to use it either to simulate traffic on a link to and from a subset of clients as well as traffic to and from the server communicating with all active clients. The number of active clients as well as session durations have to be set for the duration of the simulation or must be described on a higher model level, e.g. using the results of [8]. The game traffic model is not suited to provide background traffic for evaluations of other traffic flows. Its use is clearly in the evaluation of quality of service (QoS) aspects of networks in respect to games.

In order to asses the impact of packet delay or packet loss experienced in a simulation, it is necessary to define QoS metrics for gaming applications. Today’s games can cope with an enormous lag (ping, round trip time) and loss. These applications are thought to be used over the Internet with a typical round trip time of 50 to 150 ms. If analog modems are used, each use introduces an additional latency of 30 to 40 ms, i.e. an additional 120 to 160 ms to the round trip time for a dial-up player [12]. Ping times frequently show 300 ms and more. Consideration of loss and lag are an essential part of the game design. Game designers try to optimize for 200 to 250 ms round trip time and provide robustness for larger lag. This is achieved by client-side prediction of the game state, i.e. movement of objects and other players [3,6]. By combining movement with inertia or reducing maximum velocity of objects prediction is even more effective [12].

Such considerations result in very robust games tolerating lag up to one second and loss up to 40%. However, these values should not be taken as criteria for good or bad QoS since acceptable gameplay requires far better performance. Ping times of 50 ms or 150 ms make a huge difference [11]. In [1] an evaluation of player effectiveness over that players ping time shows that players with lower ping times score significantly more kills than others. In a discussion on [10] players give comments on impact of ping times on gameplay. Some players report that they adapt their playing strategies to high ping times and may even enjoy a game with 200 ms lag. The impact of lag also depends on the game. As „Quake III Arena“ is very fast and responsive the ping time almost automatically decides on winning or losing. „Quake World“ or „Unreal Tournament“ are reported to behave much better in this regard, i.e. ping times are not as decisive for successful playing.

Based on [11] and [10] we find that a ping below 50 ms is associated with excellent game play. A ping below 100 ms is good and above that, playability decreases noticeably. Ping times above 150 ms are often reported to be intolerable but many players claim to have no problems with ping times around 200 ms. An evaluation on „Half Life“ reported in [9] shows that players who experience high ping times of over 225 ms do not quit and look for a faster server but stay and continue to play with this high lag. We assume that those players use 56k modems and do not expect to get a better connection elsewhere. The study reveals that many gamers (40%) play with a high lag of over 225 ms despite of the decreased playability.

The impact of packet loss is rarely discussed as it is experienced as lag as well. However, a high ping time without packet loss is preferable to a small ping time with packet loss of around 10%.

3. CONCLUSIONS

We have presented a traffic characterization of the popular multiplayer game „Counter Strike“. For simple source modelling of this traffic we treat clients and server independently and focus on in-game phases with very regular traffic patterns. Although the model neglects effects of correlations between clients and server it allows the performance evaluation of network systems in regard to popular game traffic. Note that the evaluated trace was taken from a LAN and that true Internet game traffic may look different.

As our observations are very close to those found by Borella for „Quake 2“ as well as to a brief evaluation we have done for „Unreal Tournament“ (another very popular first person shooter) we feel that the whole class of fast action multiplayer games can be described with a general traffic model. The discussion on QoS metrics for first person shooters is far from complete but gives insight into network requirements of games and allows a rough assessment of simulation results.

Other game genres require different traffic models. For Age of Empires, a popular strategy game, a totally different networking architecture is used (fully meshed, max. 8 players) and QoS requirements differ as well (lag of below 250 ms is good, lag up to 500 ms payable and beyond that the lag is noticeable) [4].

In future games the quality of graphics will continue to increase enormously but game traffic is not so likely to change much from what it looks like now except for additional voice communication which will be incorporated in most online games soon.

4. REFERENCES

[1] Armitage, Grenville: Sensitivity of Quake3 Players To Network Latency, IMW2001 workshop poster session, Nov 2001, http://www.geocities.com/gj_armitage/q3/quake-results.html

[2] McCreary S., claffy k.: Trends in wide area IP traffic patterns - A view from Ames Internet Exchange, *ITC Spec. Seminar*, 2000

[3] Bernier Y.W: Latency Compensating Methods in Client/Server In-game Protocol Design and Optimization, *Game Developers Conference 2001*, http://www.gdconf.com/archives/proceedings/2001/prog_papers.html

[4] Bettner P., Terrano M.: 1500 Archers on a 28.8: Network Programming in Age of Empires and Beyond, *Game Developers Conference 2001*, http://www.gdconf.com/archives/ proceedings/2001/prog_papers.html

[5] Borella M.S.: Source Models of Network Game Traffic, *Networld + Interop'99 Engineer's Conference*, May 1999

[6] Bonham S., Grossman D., Portnoy W., Tam K.: Quake: An Example Multi-User Network Application - Problems and Solutions in Disributed Interactive Simulations, *CSE 561 Term Project Report*, University of Washington, July 2000, http://www.cs.washington.edu/homes/grossman/projects/561projects/quake/

[7] Game2k1, http://www.game2k1.com/

[8] Henderson T., Bhatti S.: Modelling user behaviour in networked games, *Multimedia 2001*, June 2001, http://www.acm.org/sigs/sigmm/MM2001/ep/henderson/

[9] Henderson T., Latency and user behaviour on a multiplayer games server. *NGC 2001*, London, UK, pp1-13, Nov 2001, http://www.cs.ucl.ac.uk/staff/T.Henderson/docs.html

[10] Slashdot, How Fast Too Slow? A Study Of Quake Pings, discussion forum, May 2001, http://slashdot.org/article.pl?sid=01/05/24/ 2044233

[11] Trier M.: Highspeed-Internet, *GameStar (Gaming Magazine)*, March 2002, pp.164-165

[12] Ng Yu-Sheng: Designing Fast-Action Games for the Internet, *Gamasutra (Online Game Developer Magazine)*, May 1997, http://www.gamasutra.com/features/19970905/ng_01.htm