Network Traffic from Anarchy Online: Analysis, Statistics and Applications

A server-side traffic trace

Andreas Petlund[1,2], Pål Halvorsen[1,2], Pål Frogner Hansen[3], Torbjörn Lindgren[3], Rui Casais[3], Carsten Griwodz[1,2]

[1]Department of Informatics, University of Oslo, Norway
[2]Simula Research Laboratory, Norway
[3]Funcom, Norway

{ apetlund, paalh, griff }@ifi.uio.no
{ palfh, tl, ruic }@funcom.com

ABSTRACT

We present a dataset – a real-world, **server-side** packet trace [8] – from Anarchy Online [9]. Anarchy Online is a science fiction-themed massively multiplayer online roleplaying game (MMORPG), published and developed by Funcom [7]. We present statistics from the network traffic and show that it is a representative dataset for similar games. From the dataset, one can extract several key characteristics from such scenarios like payload sizes, packet rates, data delivery latencies, retransmission statistics, loss rates and stream correlation. The dataset can be used several ways: by replaying the game traffic, components like congestion control mechanisms, middlewares, packet schedulers, router queue behaviour, etc. can be analysed. Based on the observed statistics from the trace, such interactive game traffic shows completely different behaviour compared to the greedy, high-rate streams most network mechanisms are designed for, e.g., file download, video streaming and web-surfing. We hope that the dataset can be used to push research forward in the field of system support for games.

Categories and Subject Descriptors

H.5.1 [**Information Systems Applications**]: Multimedia Information Systems

General Terms

Measurement

Keywords

Packet trace, game traffic behaviour

1. INTRODUCTION

Massively multi-player online games (MMOGs) allow thousands of users to interact concurrently in a persistent virtual environment. For this to work, there are stringent latency requirements whose exact nature depends on the model of interaction, which again typically differs between game genres, and the network conditions. A challenge is that games typically generate a completely different traffic pattern compared to greedy high-rate data downloads like web and file transfer traffic, or video and audio streaming.

With the recent explosion of on-line multiplayer games during the past decade and with the millions of games, computers and game-consoles sold every year, it is important to understand the behaviour and the influence this type of network traffic has in the Internet in order to design and implement the future network infrastructure. To aid analysis and experiments in this scenario, traffic traces have been used for several types of games. From such a traffic trace, knowledge of several key characteristics can be extracted, like payload sizes, packet rates, data delivery latencies, retransmission statistics, loss rates and stream correlation. Such datasets can be used in several ways, but by replaying the game traffic, components like congestion control mechanisms, middlewares, packet schedulers, router queue behaviour, etc. can be analysed where one probably will experience a completely different behaviour compared to the greedy, high-rate streams most mechanisms are designed for. Thus, they are potentially very valuable to Internet service providers for better network planning, to network equipment manufacturers for better management of such traffic and to software developers for better virtual world design.

Here, we make available a real-world, **server-side** packet trace [8]. Anarchy Online [9] is a massively multiplayer online roleplaying game (MMORPG) by Funcom [7], and after more than ten years, Anarchy Online has become one of the oldest surviving games in the genre. We are aware of several existing related datasets for other games or virtual worlds, e.g., [4, 12], but to the best of our knowledge, this is the first available dataset from a large commercial game provider captured at the game server. We present statistics from the network traffic and show that it is a representative dataset for similar games. We hope that the dataset can be used to push research forward in the field of better system support for applications like games.

2. ANARCHY ONLINE: THE GAME

> *"You are 30.000 years into the future. You're in the age... of Anarchy Online"* [9].

Anarchy Online [9] is a science fiction-themed MMORPG published and developed by Funcom [7]. The game's multiplayer nature is a "free-form" game-play and encourage creation of social networks where a player can cooperate and fight with other players. Among the most distinct game-play element of Anarchy Online is dynamic missions. Such missions, or quests, are a traditional game-play element in the role playing genre, and the player, or a group of players, is given a set of story-related tasks where they have to interact with human players and computer-controlled characters, both friendly and hostile. In return for solving the mission, the players are rewarded with experience points, items and money. The players can choose from various professions having different skills, clothes, weapons and armours. After ten years, Anarchy Online has become one of the oldest surviving games in the genre.

Anarchy Online takes place on the fictional planet Rubi-Ka, and in the game, you step almost 30,000 years into the future. Here, you are in an age where common surgical implants and microscopic nano-bots can relieve most forms of human suffering – or transform any normal being into a weapon of destructive force. On Rubi-Ka, a battle rages between opposing clans. The clans seek liberation from the all-powerful Omni-Tek corporation; the mega-corporation that rediscovered Rubi-Ka after the planet was ripped apart in a cataclysm. They pursue this liberation by peaceful means, if possible, or if not, by brute force. Leaders of both sides are desperately seeking a solution that can stop the violence ravaging the planet, against a backdrop of betrayal, military raids and conniving political games, cyborg assassins and modified humans.

The game is divided into separate world instances; there are game servers located at different places in the world that each holds an identical copy of the game's 3D virtual world[1]. In this respect, characters and players are not able to interact with one another across instances, even in the same server clusters. Furthermore, dynamic missions, like many other encounters in Anarchy Online, are also instanced where each mission area is available only to the owners of the mission. As shown in figure 1, at each location, the game world is divided into playfields (also called regions in other games) where one gameserver controls one or more playfields depending on the number of concurrent users, i.e., no playfields spans several gameservers. Furthermore, Anarchy Online keeps point-to-point, default (New Reno) TCP connections open, from one or more Linux servers to every client.

[1]This was the situation at the time of the capture. Today, there are still several publicly open instances, but they are all co-located in the same data centre.

3. THE TRACE

To be able to analyse the Anarchy Online game traffic, we have captured the packets from one of many playfield from the game servers located in the U.S. We used *tcpdump* [1] to record the packet headers (no payload), and we dumped all network traffic for about an hour (3918 seconds). We observed 119 different users in the playfield. In total, 362833 packets were sent from the server with an estimated average loss rate of 0.896 %.

For this dataset, we have anonymised all Ethernet and IP addresses in order to preserve the privacy of Funcom and all their users. Since we have captured the data at the server, we only see the last hop router Ethernet addresses and the Ethernet address of the server. These machines have been assigned the addresses `02:00:00:00:00:01` with increasing numbers in the last octet of the address. The first octet is set to "02", indicating a locally managed MAC-address.

We found 176 distinct TCP connections in our packet dump originating from 120 different IP addresses, i.e., a user may have several connections with different port numbers, for example when moving in and out of different playfields in the game. The endpoints of these connections have been given IP addresses in the range of `0.0.0.1` to `0.0.0.120` regardless of the original IP address[2]. Thus, it is not possible to extract player identifications or locations from the trace, or game specific content. However, you can still see the original packet patterns with the original, real-world timing (spacing between the packets) per client, including all sent and retransmitted packets and their TCP acknowledgements.

Since no other modifications have been performed to the trace, it can easily be read using existing tools like *tcpdump* [1] and *wireshark* [2], or using the *libpcap* library [1] to write own tools for analysis. A selection of our tools used for the processing of this dump is available at [8].

4. TRAFFIC ANALYSIS

We have analysed the packet trace containing all the packets from one of a few hundred game playfields in Anarchy Online. In our analysis, we have mostly looked at server to client communication since we have an interest in studying behaviour of sender-side mechanisms of TCP. In summary, the lag experienced by players with lossy connections can be huge. Simultaneously, the congestion window grows very slowly even in slow start since several writes will be merged into a single packet, and repeated losses can keep the congestion window size close to 1. Thus, the main challenge for this kind of traffic is not the bandwidth, but the delay due to retransmissions that is typical for this kind of distributed multimedia application today. Therefore, it is important to note and deal with the most important observation which is that the datastreams are so thin, meaning that each packet is very small and there are only a few sent packets per RTT (see table 1). Among the effects of these properties are: 1) the connections hardly ever trigger fast retransmissions but mainly retransmit data by timeouts and 2) TCP's congestion control does not apply, i.e., the TCP stream cannot back off (as the congestion window is already at the minimum). Ad-

[2]The addresses were assigned in order of appearance, and for this trace, the server was given the address 0.0.0.2.

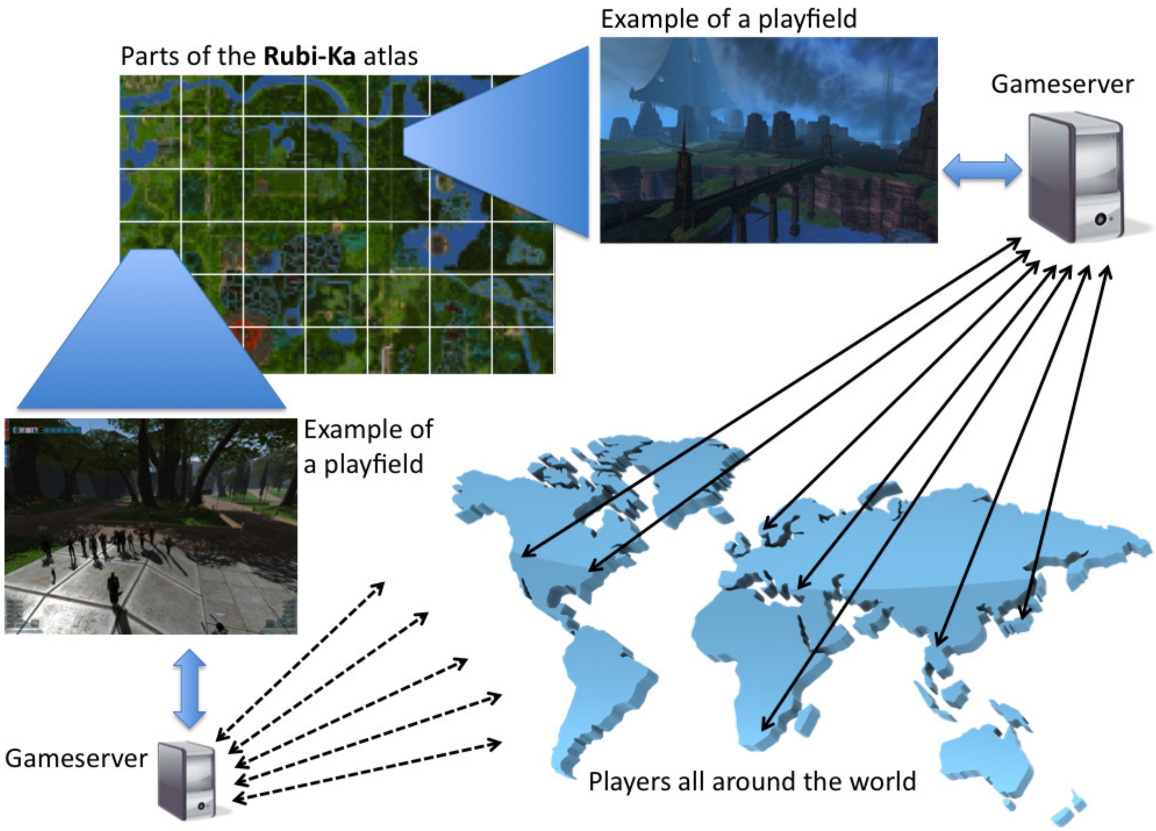


Figure 1: The Anarchy Online world is at each server location divided into smaller areas, or playfields, each controlled by a gameserver which again potentially serve clients world-wide. (Note that the figure is an illustration only, i.e., the screenshots [9] do not necessarily stem from the shown area on the map.)

ditionally, there are several connections that probably share links in the path to the server, and we should use this observation to conserve network resources.

4.1 Observed RTTs and delays

A critical parameter in time-dependent applications like games and virtual worlds is the timeliness of the packet delivery. In this trace, the packet delivery latency vary a lot. From figure 2, we see that the average RTT (measured TCP acknowledgement latencies) is somewhat above 250 ms with variations up to one second. These RTTs make the game playable [6], but looking at the maximum application delay (time between first transmission and the receipt of a successful acknowledgement) which may include several retransmissions as a result of packet loss, we have extreme worst case delays of up 67 (!) seconds. This extreme latency were caused by six consecutive losses of the same segment, handled by timeout retransmission with an exponentially increasing RTO[3]. Obviously, we cannot distinguish between lost packets and lost acknowledgements in this server-sided trace, but we can see the possibility of several second-long delays in delivering packets to the application on the client side, i.e., delays that severely hurt the users' quality of experience. Furthermore, knowing that the servers are located in the US, one can assume from the observed minimum latencies that there are players concurrently located in at least three different continents – America, Europe and Asia.

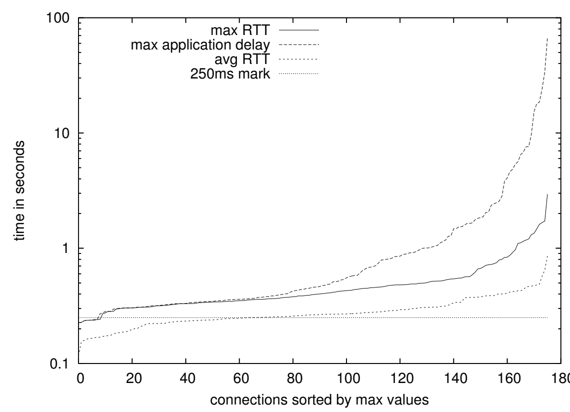


Figure 2: Observed RTTs and maximum, delays

[3]The low packet rates prevent TCP from retransmitting by *fast retransmit* meaning that loss is detected by timeout.

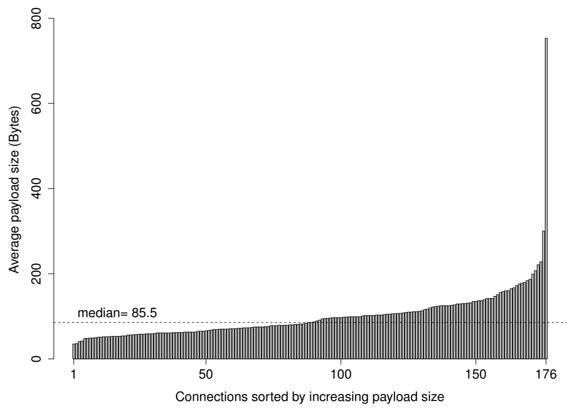


Figure 3: Observed average payload size per connection

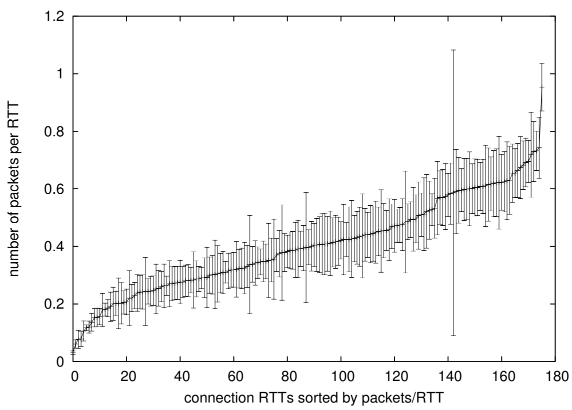


Figure 4: Packets per RTT with standard deviation sorted by the interarrival time

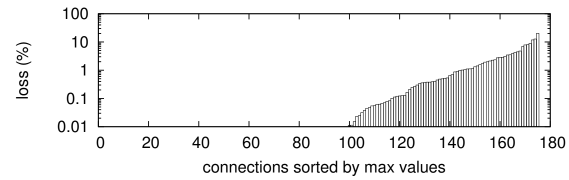
4.2 Payload sizes and packet rates

A statistical overview of the whole trace can be found in table 1. As we can see, the packets are in average very small, with a payload usually below 100 bytes. We did not see, in the entire trace, a full MTU, i.e., a strong contrast to traditional download and streaming applications.

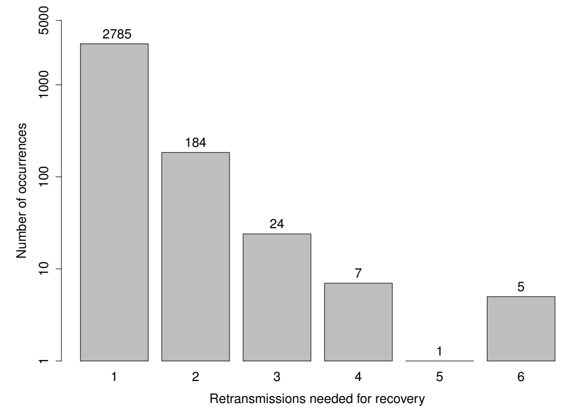
With respect to packet frequency, or the packet interarrival times, the normal is to send a packet every half second, i.e., often less than one packet per RTT. This depends on the activity in the game, and also possibly the distance to the server, and is clearly visible if we look at the per connection data in figure 4.

4.3 Loss rates and number of retransmissions

With respect to losses, the trace shows a total loss probability of slightly less than 1 %. In figure 5(a), we have plotted the per-stream loss rate. As expected, some clients are very well connected and hardly see any loss at all, whereas others experience a lot of loss with the worst-case of about 20 % loss. Furthermore, as seen in figure 5(b), the trace also contains several instances of many consecutive retransmissions



(a) Per-stream loss rate (log scale y-axis)



(b) Number of retransmissions needed to recover data

Figure 5: Loss rates and retransmission statistics

of the same packet; the maximum is 6, which caused the large delays reported in section 4.1. This implies that it is not the loss rate itself that is unacceptable, but the occasional huge delays when multiple retransmissions are needed. Moreover, we have not found any correlation between losses of the various connections in the trace, and we would like to conclude that losses across the connections in the trace are uncorrelated. This would imply that losses are not due to server-sided bottlenecks. We use this as a working assumption, however, the thinness of the streams may actually hide loss correlation.

4.4 Sharing paths

In such a dataset, it might also be interesting to predict if many clients share (parts of) a data path in order to better schedule the consumption of, and possibly preserve, network resources. Figure 6 visualises the results of a Wilcoxon test [11] of connection pairs that checks whether their RTT values stem from the same RTT base value set. A dot in the matrix shows that two connections with high probability share a path. Thus, there is a large possibility that groups of the connections in our trace have shared paths.

4.5 Relevance to other games

To test whether the captured game traffic from Anarchy Online also is representative for other game traffic, we have captured the traffic of several other games, as seen in table 1. The upper line (Anarchy Online) shows the statistics from our server side trace counting all streams that sent more than 1.000 packets. The other rows show statistics for client-

application	tested (platform)	prot-ocol	payload size (bytes) average	min	max	packet interarrival time (ms) average	median	min	max	percentiles 1%	99%	avg. bandwidth requirement (pps)	(bps)
Anarchy Online‡ *	**PC**	**TCP**	**98**	**8**	**1333**	**632**	**449**	**7**	**17032**	**83**	**4195**	**1.582**	**2168**
World of Warcraft	PC	TCP	26	6	1228	314	133	0	14855	0	3785	3.185	2046
Counter Strike‡ *	PC	UDP	36	25	1342	124	65	0	66354	34	575	8.064	19604
Halo 3 - high intensity	Xbox 360	UDP	247	32	1264	36	33	0	1403	32	182	27.778	60223
Halo 3 - moderate intensity	Xbox 360	UDP	270	32	280	67	66	32	716	64	69	14.925	35888
Gears of War	Xbox 360	UDP	66	32	705	457	113	3	10155	14	8953	2.188	10264
Tony Hawk's Project 8	Xbox 360	UDP	90	32	576	308	163	0	4070	53	2332	3.247	5812
Test Drive Unlimited	Xbox 360	UDP	80	34	104	40	33	0	298	0	158	25.000	22912
BZFlag	PC	TCP	30	4	1448	24	0	0	540	0	151	41.667	31370
World in Conflict*	PC	UDP	365	4	1361	104	100	<1	315	<1	300	9.615	31.000
World in Conflict	PC	UDP	4	4	113	105	100	16	1022	44	299	9.524	4443
YouTube stream	*PC*	*TCP*	*1446*	*112*	*1448*	*9*	*<1*	*<1*	*1335*	*<1*	*127*	*111.11*	*1278K*
HTTP download	*PC*	*TCP*	*1447*	*64*	*1448*	*<1*	*<1*	*<1*	*186*	*<1*	*8*	*>1000*	*14M*
FTP download	*PC*	*TCP*	*1447*	*40*	*1448*	*<1*	*<1*	*<1*	*339*	*<1*	*<1*	*>1000*	*82M*

‡ The presented values are average values over all players (sending minimum 1000 packets) within the period of the trace.
* Server side trace.

Table 1: Examples of packet statistics based on analysis of packet traces for several types of games (and high-rate streams).

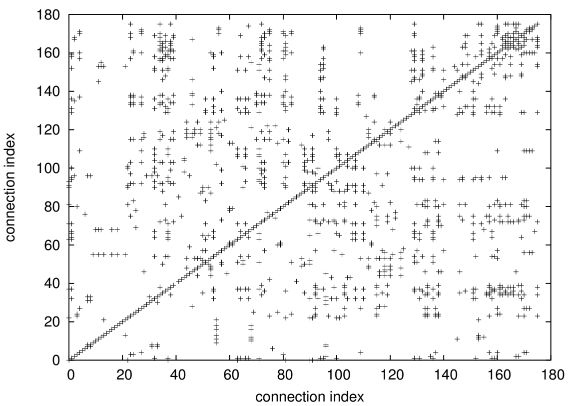


Figure 6: Connections that probably share a path

side traces, i.e., we did not have access to the servers[4]. As we can see, similar trends can be observed for all the games, at least compared to traditional greedy traffic, making our trace a representative example, even for other games than MMORPGs.

5. POTENTIAL USE SCENARIOS

The available dataset can be used several ways. The statistics from the trace analysis give valuable knowledge about this type of traffic with respect to packet sizes and packet transmission frequencies, i.e., the game traffic properties are greatly different from traditional greedy streams.

Using trace-replay tools one can evaluate the behaviour of system components with a real-world traffic pattern. For example, traditional protocol behaviour will not occur since the existing protocols are designed for other types of traffic. Thus, replaying the game traffic, components like congestion control mechanisms, middlewares, packet schedulers, router queue behaviour, etc. can be analysed.

The dataset gives statistics of loss rates and retransmissions for a selection of users from around the world in a gaming scenario. This data can be fed into a network emulator to create similar conditions for replayed experiments.

6. RELATED DATASETS

There exist several papers that have looked at game traffic characteristics (e.g., [3, 5, 10]), and there are some datasets available. For example, Wu-chang Feng et. al. [4] have presented characteristics for a Counter-Strike scenario where one can observe that Counter Strike sends even smaller packets, but a little faster due to the faster game pace in a first-person shooter (see also table 1 for a presentation of our analysis of the data). Furthermore, Claypool [6][5] investigated the effect of latency on user performance in Real-Time Strategy games. Moreover, Wang et. al. [12][6]. have made available a dataset from Second Life looking for example at throughput, packet size and interarrival times. However, even though game traffic has been analysed before, only a few datasets are publicly available on the Internet. These existing traces are captures at small (yet possibly popular) servers managed by the authors, or they are client-side packet dumps. Thus, to the best of our knowledge, this Anarchy Online dataset is the first available packet trace from a commercial game provider captured **server-side** at the game server data centre.

7. CONCLUSIONS

We have here presented and made available a **server-side** packet trace [8] of the game Anarchy Online from Funcom, i.e., a large-scale commercial game provider. We have presented packet characteristics and described the network behaviour. We use this analysis to show the influence of trans-

[4]There are server-side traces for "World in Conflict" and "Counter Strike", but this is locally hosted game servers with fewer clients, and not in the scale of a full-fledged MMORPG server-side trace.

[5]Traces available at http://perform.wpi.edu/downloads/
[6]Traces available at http://12.71.54.173/sl/

mission patterns on perceived data delivery latencies. Using such a trace, one can replay and simulate real-world traffic and analyse the software and hardware components in the datapath with respect to the performance. We hope that the dataset can be used to push research forward in the field of system support for games.

Acknowledgements

This work has been performed in the context of the iAD (Information Access Disruptions) centre for Research-based Innovation funded by Norwegian Research Council, project number 174867.

8. REFERENCES

[1] tcpdump & libpcap. http://www.tcpdump.org.

[2] wireshark. http://www.wireshark.org.

[3] Borella, M. S. Source models of network game traffic. 403–410.

[4] chang Feng, W., Chang, F., chi Feng, W., and Walpole, J. A traffic characterization of popular on-line games. 488–500.

[5] Chen, K.-T., Huang, P., Huang, C.-Y., and Lei, C.-L. Games traffic analysis: An MMORPG perspective. pp. 19–24.

[6] Claypool, M. The effect of latency on user performance in real-time strategy games. 52–70.

[7] Funcom. http://www.funcom.com/.

[8] Funcom. Server-side trace from the MMORPG Anarchy Online. `http://home.ifi.uio.no/paalh/dataset/ao`.

[9] Funcom. Anarchy online. http://www.anarchy-online.com/, Sept. 2009.

[10] Henderson, T., and Bhatti, S. Modelling user behaviour in networked games. In *Proceedings of the 9th ACM International Conference on Multimedia (ACM Multimedia)* (Oct. 2001), pp. 212–220.

[11] Larsen, R. J., and Marx, M. L. *An Introduction to Mathematical Statistics and Its Applications*. Prentice Hall, 1986.

[12] Wang, Y., Hsu, C.-H., Singh, J. P., and Liu, X. Network traces of virtual worlds: Measurements and applications. In *Proceedings of Multimedia Systems (MMSys)* (Feb. 2011), pp. 105–110.