Game Traffic Analysis: An MMORPG Perspective[*]

Kuan-Ta Chen[1][2], Polly Huang[3]†, Chun-Ying Huang[1]‡, Chin-Laung Lei[3]
{jethro,huangant‡}@fractal.ee.ntu.edu.tw, {phuang†,lei}@cc.ee.ntu.edu.tw

[1]Department of Electrical Engineering
National Taiwan University

[2]Institute of Information Science
Academia Sinica

[3]Department of Electrical Engineering and Graduate Institute of Networking and Multimedia
National Taiwan University

ABSTRACT

Online gaming is one of the most profitable businesses over the Internet. Among all genres of the online games, the popularity of the MMORPG (Massive Multiplayer Online Role Playing Games) is especially prominent in Asia. Opting for a better understanding of the game traffic and the economic well being of the Internet, we analyze a 1,356-million-packet trace from a sizeable MMORPG, *ShenZhou Online*. This work is, as far as we know, the first formal analysis on the MMORPG server traces.

We find that the MMORPG and FPS (First-Person Shooting) games are similar in that they both generate small packets and require low bandwidths. In particular, the bandwidth requirement of MMORPG is even lower due to the less real-time game play. More distinctive are the strong periodicity, temporal locality, and irregularity observed in the MMORPG traffic. The periodicity is due to a common practice in game implementation, where the game state updates are accumulated within a fixed time window before transmission. The temporal locality in the game traffic is largely due to the game nature where one action leads to another. The irregularity, particular unique in MMORPG traffic, is due to the diversity of game design where the user behavior can be drastically different depending on the quest at hand.

Categories and Subject Descriptors

C.2.5 [**Local and Wide-Area Networks**]: Internet; H.4.3 [**Information Systems Applications**]: Communications Applications; K.8.0 [**Personal Computing**]: General—*Games*

General Terms

Measurement, Human Factors

Keywords

Network Games, Traffic Analysis, Internet Measurement

1. INTRODUCTION

As online games, especially MMOG (Massive Multiplayer Online Games) [10], grow popular, the share of game traffic on the Internet has become increasingly significant. Reported in a backbone traffic analysis [7], about 3%–4% of the traffic is attributed to 6 popular games. Given the significant share of game traffic and the dissimilar nature of games from dominant Internet applications such as the World Wide Web, peer-to-peer filing sharing, and streaming, a better understanding of the game traffic is vital.

We aim particularly at MMORPG (Massive Multiplayer Online Role Playing Games) for two reasons. First of all, MMORPG is the most prominent game genre in Asia. In Taiwan, Gamania[1], operator of the popular game – Lineage[2], owns more than 4,000 Mbps dedicated links[3] for game traffic. According to Game Flier[4], the top game, Ragnarok Online, has claimed a record of 370,000 players online simultaneously. This is about 1.5% of the population in Taiwan. Furthermore, most MMORPG in Asia exchange messages using TCP. Although TCP is not designed for real-time communication, it is not clear as yet whether TCP is not suitable for the MMORPG traffic transmission. To the best of our knowledge, this work is the first analyzing and characterizing the MMORPG traffic.

We trace *ShenZhou Online* [1], a mid-sized MMORPG, and analyze a 1,356-million-packet trace. Our analysis focuses on the packet size, bandwidth usage, packet interarrival within a connection, and the arrival process of the aggregated traffic. The major findings are summarized as follows:

- Most of the packets are *small*. 98% packets sent by the game clients are smaller than 71 bytes. This suggests that the overhead of packet headers and TCP acknowledgments is high relative to other Internet applications. In total client traffic, headers occupy 73% bytes, and TCP acknowledgement packets take up 30%.
- The average bandwidth requirement per client is about 7 Kbps, which is *much lower* than the 40 Kbps average

---

[*] This research is supported in part by the National Science Council of the Republic of China under grant NSC 93-2218-E-002-103.

NOSSDAV'05, June 13–14, 2005, Stevenson, Washington, USA.

[1] Gamania, http://www.gamania.com/
[2] NCsoft Corporation, http://www.lineage.com/
[3] TWNIC, http://map.twnic.net.tw/
[4] Game Flier, http://www.gameflier.com/

observed from *Counter-Strike* [5]. We believe the lower bandwidth requirement is due to the relatively slow motion or action pace in MMORPG.

- The traffic of either direction exhibits short-term positive auto-correlations within connections. In the client traffic (traffic going from the client to the server), the positive auto-correlations is due to the *temporal locality in user actions*. In the meantime, the same effect in server traffic is due to the *spatial locality in the number of nearby characters*. The spatial locality shows up in terms of temporal locality in the traffic as the characters move continuously on the map.
- Positive auto-correlations still exist in aggregated client traffic. We consider it is owing to the *global events* in games, which cause the "flash crowds" effect. Furthermore, the aggregated traffic of either direction exhibit strong *periodicity*. It implies the game processing for all clients are synchronized.

The remainder of this paper discuss these issues in more detail. Section 2 describes related works and the game *ShenZhou Online*. We show the trace measurement methodology and trace summary in Section 3. In Section 4, we present a detailed analysis on the game traffic from various aspects. Finally we conclude in Section 5.

2. BACKGROUND

2.1 Related Work

As network games are becoming a noteworthy contributor to the overall Internet traffic, many efforts have been dedicated to the traffic analysis and modeling for network games. Bangun *et al.* [2] analyzed a network trace captured at an Internet Café for two LAN games, *Quake* and *Starcraft*. Their works focus on how the payload sizes and inter-packet times vary with the number of players. Later Bangun and Dutkiewicz proposed models of payloads and inter-packet times based on another trace of a LAN game *Starsiege Tribes*. Borella proposed source models for a popular FPS (First-Person Shooting) game *Quake* where the packet interarrival times and packet sizes are modelled as extreme distribution, exponential distribution, or deterministic [3]. A similar work by Färber characterized traffic for another FPS game *Counter-Strike* [4].

Feng *et al.* [5] analyzed a 500-million-packet trace of a busy *Counter-Strike* server. Their analysis revealed that game traffic is highly predictable, however, contains bursts of tiny packets. Though MMORPG is a very different game genre from FPS games, we find MMORPG are similar to FPS games in terms of packet size and periodicity. In [5] the authors continued to evaluate the impact of tiny packets on network infrastructures by a pressure test upon a commercial off-the-shelf NAT device. The experiment indicates the NAT device cannot handle the game packets well.

2.2 About ShenZhou Online

ShenZhou Online is a mid-scale, commercial MMORPG in Taiwan [1]. There are thousands of players online at any time. To play, the players purchase the "game points" either from the convenient stores or online. A screen shot of *ShenZhou Online* is shown in Fig. 1. The character played by the author is the man in the center of the screen and with a smile face on his top. He is in a typical market place while other players are keeping stalls. Very typical of an MMORPG, the players can engage fights with the other players or random creatures, train oneself of special trade skills, participate in marketplace commerce, or take on a quest journey.

Figure 1: A screen shot of *ShenZhou Online*

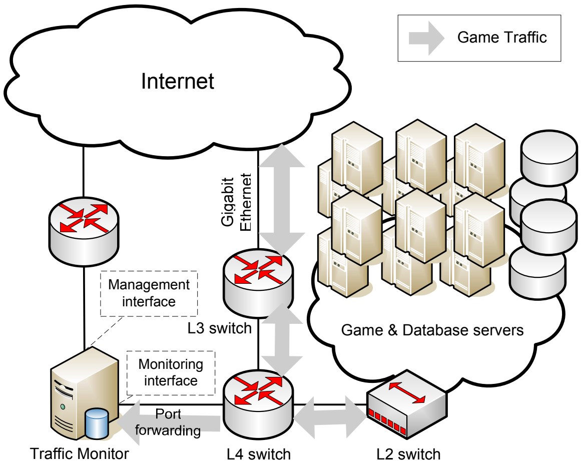

Figure 2: Network setup of traffic measurement

3. GAME TRAFFIC TRACES

With the help of the *ShenZhou Online* staffs, we set up a traffic monitor aside the game servers. The traffic monitor is attached to a layer-4 switch, upstream the LAN containing the game servers (we shall call it the "game LAN"). The port forwarding capability of the tapped layer-4 switch is enabled so that all inbound/outbound game traffic is forwarded to our monitor as a copy. In order to minimize the impact of monitoring, all remote management operations are conducted via additional network path, i.e., the game traffic and management traffic do not interfere each other. The network topology and setup of the game servers and the traffic monitor are depicted in Fig. 2.

The traffic monitor is a FreeBSD PC equipped with Pentium 4, 1.5 GHz and 256 MB RAM. We use *tcpdump* [6] with the kernel built-in BPF to obtain traffic traces. Because of the restriction of network topology, the switch has forwarded all traffic sent to and sent from the game LAN, including non game playing traffic such as HTTP, DNS and

Table 1: Summary of Game Traffic Traces

Trace	Sets	Date	Time	Period	Drops†	Conn. (Cens.)	Pkt. (in / out / both)	Bytes (in / out / both)
$\mathcal{N}_1$	3	8/29/04	15:00	8 hr.	0.003%	57,945 (6.5%)	342M / 353M / 695M	4.7TB / 27.3TB / 32.0TB
$\mathcal{N}_2$	2	8/30/04	13:00	12 hr.	?‡	54,424 (3.5%)	325M / 336M / 661M	4.7TB / 21.7TB / 26.5TB

† The column gives the kernel drop count reported by *tcpdump*.

‡ The reported kernel drop count is zero, but we actually found some packets are dropped at the monitor.

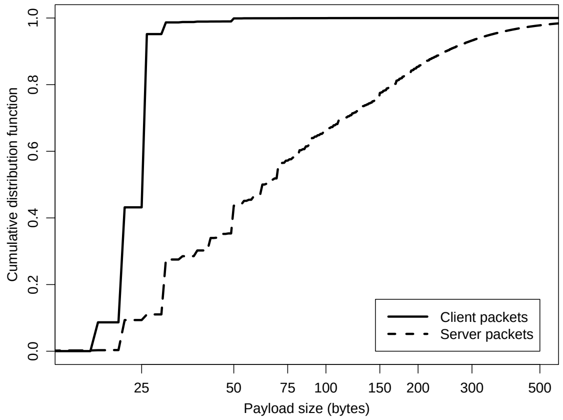

Figure 3: Payload size distribution

SMB packets. These unwanted traffic types are filtered out using filtering support of *tcpdump*. Considering data privacy and storage, only IP and TCP headers are recorded.

We randomly choose a subset of game sets in each trace; only packets belonging to selected game sets are logged. A game set is logically a "game server" from the viewpoint of players. Each game set comprises an entry server, several map servers, and a database server. All game sets are equivalent in functionality but *isolated*, and the partition of game sets is merely the consequence of limited scalability. We took two packet traces $\mathcal{N}_1$ and $\mathcal{N}_2$, each records traffic for two and three game sets, respectively. The two traces each spans across 8 hour and 12 hour respectively and contains more than 1,356 million packets in all. The traffic traces are summarized in Table 1.

4. TRAFFIC CHARACTERIZATION

In this section, we first inspect the distribution of packet size, and then examine packet load and bandwidth usage for each client. Next, we search for patterns in inter-packet times within each connection. Lastly patterns in aggregate packet arrival processes are investigated.

To keep terms short, we shall denote "client packets" as packets sent by game clients, including data packets and TCP acknowledgement packets, and "client traffic" as all traffic sent by clients. The same rules apply to "server packets" and "server traffic."

4.1 Packet Size

Fig. 3 shows the cumulative distribution function (CDF) of the payload size, which is packet size excluding TCP/IP header of 40 bytes. Pure TCP ack packets do not count. As the figure shows, client packets and server packets are drastically different in payload size. The discrepancy conforms to our intuition since client packets contain *one* player's commands, while server packets convey nearby players' actions as well as system messages. The client packets are extremely *small*: 98% of client packets has payload size smaller than or equal to 31 bytes. The two modes 23 and 27 bytes, which comprises 36% and 52% of packets respectively, exhibit that user actions are *dominated* by few popular commands such as walk and attack. On the other hand, server packets have much wider distribution with average payload size of 114 bytes. The payload size distributions generally agree with findings in [5], that is, the packet size in game traffic significantly differs from aggregate traffic seen at Internet exchange points in that the mean packet size observed are above 400 bytes [7].

Furthermore, among the traces about 38% of packets are pure TCP acks. Disabling the delayed ack option and one-way position updates are the major cause of the high ratio of pure acks. An average packet size of 84 bytes is yielded if we count overall game traffic. While routers are often designed with the assumption that average packet size are within 125 to 250 bytes [8], the popularity of online games may challenge the router lookup mechanism against considerable quantities of tiny packets.

4.2 Packet Load and Bandwidth Usage

To know the bandwidth usage, we compute the average packet load and average bandwidth for each connection. As shown in Fig. 4 and 5, the packet load and bandwidth needed for playing a MMORPG is surprisingly *low*. For most connections, the average server packet rates are smaller than 5 pkt/sec. Even with the consideration of TCP acks, 99% of connections have packet load for both directions less than 15 pkt/sec.

By the low load and small size of packets, the bandwidth requirement of MMORPG is even lower than the narrowest last-mile link, 56K modems. Nearly all clients consume less than 3 Kbps for client data packets, and less than 8 Kbps is used with the consideration of TCP acks. For overall client traffic, we can calculate the overhead of TCP/IP header and TCP acks by the ratio of bytes used; the cost is noteworthy: TCP/IP header takes up 73%, and TCP acknowledgements are responsible for 30%.

Server traffic make use of more bandwidth, but only 7 Kbps is required in average. The overall bandwidth usage is *much lower* than average 40 Kbps for *Counter-Strike* [5]. We consider the difference is due to the game nature: MMORPG is relatively slow-paced while FPS games usually require players making sub-second decisions. On the other hand, the bandwidth usage we obtained is comparable to that of an online RTS (Realtime Strategy) game, Warcraft III [9]. Indeed, the pace of gaming is similar for MMORPG and RTS games, while FPS games are much more fast-paced.

Though the bandwidth requirement of MMORPG for each client is relatively low, but generally much more concurrent

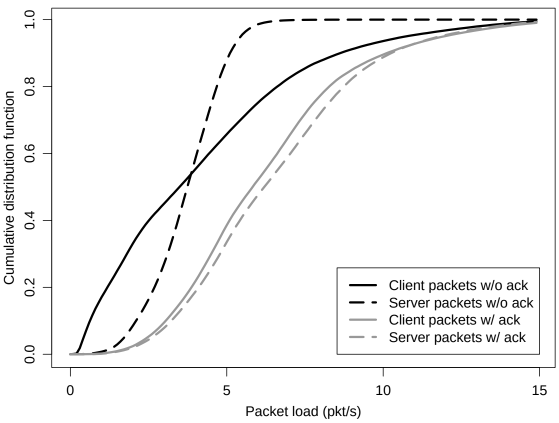

Figure 4: Packet load distribution

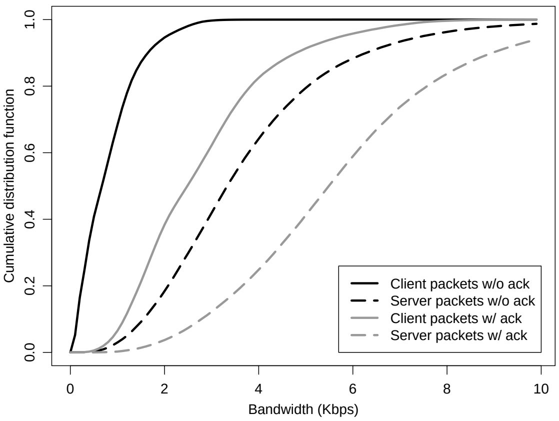

Figure 5: Bandwidth distribution

connections are made; for instance, Ragnarok Onine in Taiwan announced a record of 370,000 players online simultaneously. The record implies at least 3.7 Gbps bandwidth is required assuming each client needs 10 Kbps in average, and the amount is just for *one in hundreds of MMORPGs*. Considering the growing popularity of MMOGs, we believe their impact on Internet traffic should not be overlooked.

4.3 Packet Interarrivals within a Connection

To explore traffic patterns in each connection, we first observe the distribution of inter-packet times for data packets. As Fig. 6 shows, most of packet interarrival times are spread over 0 though 600 ms. The empirical CDF is not very close to the best-fit exponential distribution with rate 8 pkt/sec. At time scales larger than 200 ms, deviation from the exponential distribution becomes much apparent. A detailed investigation shows the situation comes from the *diversity in user behaviors*, which could be a distinct feature of MMORPG-like games.

For game genres such as FPS, RTS, and FTG (Fighting Game), during the game play, players must participate the game with high activity, or they will be defeated by others. Players are forced idle if they are defeated, but they will be active again in the next round. In contrast, MMORPG and adventure-oriented games do not need players to be active all the time. Players can do anything at will: they can thoroughly view their own equipments; they can wait somewhere for friends or some events. Unless they are in a battlefield, players are free to go anytime. We found most of players have been idled at least for a while during the trace period and certain players even keep idle most of time. We picked out some connections from the trace, which are either very active (almost no idle time) or very inactive (almost keep idle). The distributions of packet interarrival times of active sessions are nearly exponential, and that of idle sessions are far from exponential—most of inter-packet times are around 5 seconds, the interval of keep-alive timers, and inter-packet time distributions of other sessions are in-between these two extremes. Therefore, the distribution in Fig. 6 is actually a hybrid—which is close to exponential in small time scales and has much longer tail due to inactive sessions. The diversity of user behaviors makes it difficult if not impossible to define a *typical player* and consequently increase the difficulty in user behavior modelling and source traffic modelling for MMORPGs.

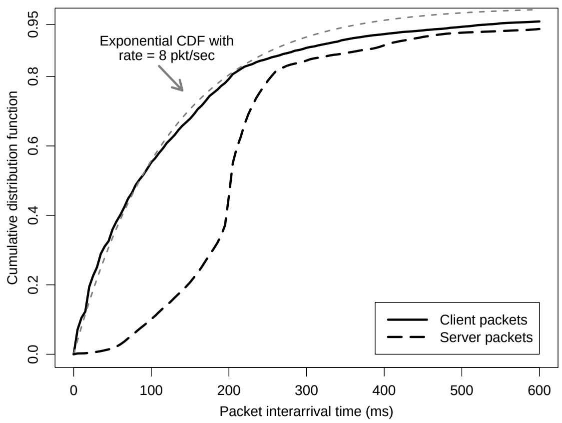

Figure 6: Packet interarrival distribution

On the other hand, server packet interarrivals are much more regular—about half of interarrivals are around 200 ms. The interval reflects the server processing is *round-based*, i.e., servers broadcast information to players on a regular basis. The periodicity of server traffic will be more clear with a frequency domain analysis in Section 4.5.

4.4 Temporal and Spatial Locality in Game Nature

Since the data packet interarrivals from clients are close to exponential for active players, one may wonder if user commands conform to Poisson. An initial check via autocorrelation function (ACF), shown in Fig. 7, indicates packet interarrivals for both directions exhibit *positive temporal dependence*. Thus the assumption of Poisson input is rejected.

In the graph, client packet inter-arrivals exhibit positive correlations up to around one minute. The phenomenon can be attributable to the *clustering nature of player actions*. The actions of players are often successive and in bursts, for example, common behaviors for a player are walking, chatting, resting, fighting, examining loot, trading, and so forth. During the time when a player chatting with other players, viewing equipments, and in business transactions, clients send nearly no traffic; on the other hand, fighting and movement result in packet bursts. All these actions usually last more than tens of seconds. Since the rate of

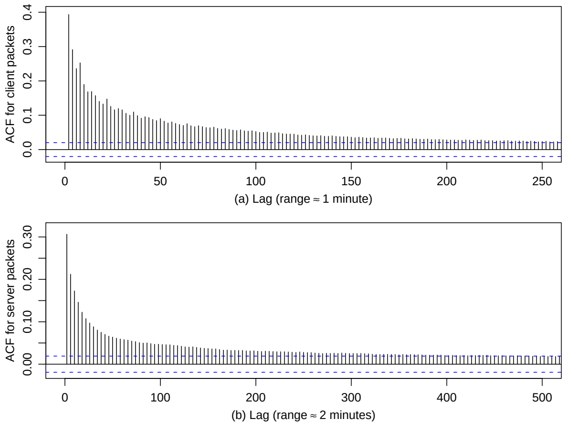

Figure 7: Correlograms of within-connection packet interarrivals

client packets depends on the nature of player actions, and player actions exhibit the above *temporal locality*, thus positive auto-correlations, up to one minute, is formed in client traffic, as shown in Fig. 7(a).

On the other hand, server packets primarily convey position updates for characters, including the player character, non-player characters (NPC), and nearby characters played by other gamers. We shall use the term "nearby characters" to denote characters around the player character, no matter they are NPC or played by other gamers. On a per-round basis, servers will notify update-to-date positions of nearby characters for each character. Specifically, servers will send out position updating messages every round except no other characters are around, or they have stayed since last position updates. Therefore, the rate of position updates is roughly proportional to the number of nearby characters. Since the game map is continuous and characters are spread over the map, with a limited scope, the number of characters across the map possesses the property of *spatial locality*. And, for a character continuously moving on the map, the spatial locality in the number of nearby characters is *transformed* to the *temporal locality* in the rate of position updates, i.e., the time series formed by the number of nearly characters is temporal dependent. An exception to break the continuity in spatial locality is that a character can "teleport" to another place instantly via a scroll or magic, but it is not so frequently used comparing with walking. Consequently, the server packet interarrivals possess the attribute of temporal locality, that is, positive auto-correlations up to a time scale of two minutes, as shown in Fig. 7(b).

4.5 Patterns in Aggregate Packet Arrivals

In the prior sections the traffic patterns within connections are explored, where periodicity and temporal dependence are shown existing. From now on, we seek to identify whether those within-connection patterns continue to exist in aggregated arrival processes of data packets.

To obtain aggregated packet arrivals, we count the number of packets in every 10 ms for inbound and outbound traffic, respectively. We ensure the stationarity in time series by selecting a subset of connections from a game set, which span a selected interval of two hours, and only packets belong to these connections during the selected interval are sampled. We examine the temporal dependence in the aggregate traffic via correlograms in Fig. 8. In the graph, the ACF are plotted with a maximum lag of 100, equivalent to time difference up to one second. As shown in Fig. 8(a), the server traffic is apparently periodic with a cycle of 200 ms, which implies the round-based position updates are *synchronous* for all clients. By the synchrony, position updating messages are sent in bursts in each round. We suspect the burstiness in position updates are unnecessary and could lead to performance problems for a large number of clients.

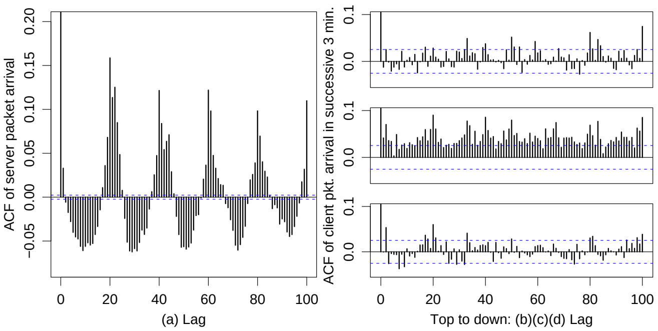

Figure 8: Correlograms of aggregate packet arrivals

Unlike server packet arrivals, client packet arrivals exhibit sustained positive auto-correlations up to *three minutes*. We have already seen the same effect on within-connection interpacket times (see Fig. 7), but the aggregated arrivals possess dependence of longer range (3 minutes versus 1 minute). The temporal dependence in aggregate packet arrivals seems counter-intuitive at the first glance. While the correlations within each connection comes from the clustering nature in one's actions, temporal locality has no reason to exist for aggregated commands from thousands of players, i.e., it is unreasonable that a number of players act or idle in synchronicity. Following a detailed inspection, we found the phenomenon is caused by the design of *global events* in the game, that is, random events that are automatically held by the system, e.g., "a horde of monsters appear in the town, and the mayor calls for help from players." When these events occur, players near the scene would join forces to eliminate the monsters. These global events are repeatedly occurred, in a frequency of several minutes. A event is ended once the mission is complete, which usually takes one minute or so. As a result, the global events cause flash crowds-like activities.

To demonstrate the effect of global events, we depict correlograms of client data packet arrivals for successive three minutes in Fig. 8(b)(c)(d). Fig. 8(c) shows significant temporal dependence, however, the effect is not exist in both previous and following minutes. The considerable difference between auto-correlations of successive minutes reveal the existence of flash crowds, that is, a number of players act at the same time during some time and dismiss immediately after the event. Due to their significance and frequency, the global events has left sustained positive auto-correlations up to three minutes in aggregate client packet arrivals.

4.6 Frequency Components

Power spectral density (PSD) is a more direct way to inspect frequency components in time series. With the same packet arrival series obtained in the prior subsection, Fig. 9 indicates strong periodicity in both directions of traffic—

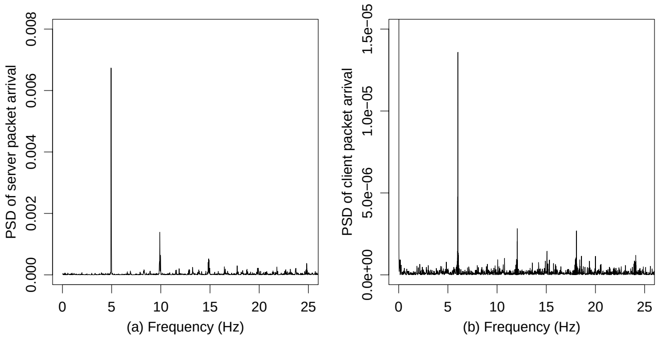

Figure 9: Power spectral density of aggregate packet arrivals

multiples of 5 Hz in server traffic and multiples of 6 Hz in client traffic. According to the graph, server traffic is more regular than client traffic since there are less peaks in Fig. 9(a). We believe it is not only due to the highly periodic feature in server traffic, but because we take the traffic trace at the server side: the client packets are timestamped after they have travelled in the network while server packets have not.

The high proportion of 200 ms server packet interarrival times, shown in Section 4.3, clearly give proof to the identified frequency 5 Hz in Fig. 9. We found multiples of 5 Hz also exist—servers seem to adapt the frequency of position updating by certain metrics such as the number of nearby characters. At the same time, we found the multiples of 6 Hz frequency components in client traffic are due to automatically generated commands, e.g., a player can switch to an "auto-movement" mode by pressing left mouse button for two seconds, then the character will continuously move toward the mouse cursor before switching back to normal mode. The attack actions can also be automatic by entering an "auto-attack" mode. In implementation, a timer with multiples of 6 Hz is used to send out movement or attack commands for the player; the frequency seems to be chosen by level and skill of the character and the weapons he/she holds. We note that the periodicity should be cancelled out if each client has its own timer, however, it exists. We consider that there is some form of synchronization mechanism in the client side to keep game clients acting in phase.

We remark different forms of synchronization in network games as a common design pattern for ease of implementation and synchronization of game states. From the aspect of network communications, however, while batched message dispatch (with a loop) is relatively intuitive in practice, they can lead to adverse impact on network performance. Further analysis is required to assess the performance implication of synchronization mechanisms. We are addressing this as part of ongoing work.

5. CONCLUSION

In this paper, we present an analysis on a packet trace from *ShenZhou Online*, a TCP-based MMORPG. The trace reveals that MMORPG traffic is very different from the traffic of dominant Internet applications, for example, file transfer and web surfing. In summary, we identify the following properties in the MMORPG traffic: 1) tiny packets, 2) periodicity, and 3) temporal dependence in packet arrivals within connections and aggregate traffic. We also provide explanations from the features of MMORPG to understand the traffic characteristics: 1) the diversity of user behaviors, 2) temporal locality in user inputs, and 3) the flash crowds effect.

As to the question of whether TCP is suitable or not for MMORPG, we observe a significant amount of overhead from the TCP/IP header and TCP acknowledgment packets. The former accounts for 73% of transmitted bytes and the latter 30%. This suggests that TCP with a sizeable header and positive acknowledgement mechanism is an overkill for applications such as MMORPG. The effect of TCP on the end-to-end delay of MMORPG is yet to be explored in the future.

Acknowledgments

This work would not have been possible without the extensive traffic trace. The authors are much indebted to the following people who helped us to gather the trace: Tsing-San Cheng, Lawrence Ho, Chen-Hsi Li, and especially to Yen-Shuo Su, who between them made the datasets available. The authors also acknowledge Wu-chang Feng and anonymous referees for their constructive criticisms.

6. REFERENCES

[1] ShenZhou Online. http://www.ewsoft.com.tw/.

[2] R. A. Bangun, E. Dutkiewicz, and G. J. Anido. An analysis of multi-player network games traffic. In *Proceedings of IEEE MMSP '99*, 1999.

[3] M. S. Borella. Source models of network game traffic. In *Proceedings of networld+interop '99*, 1999.

[4] J. Färber. Network game traffic modelling. In *NetGames '02: Proceedings of the 1st Workshop on Network and System Support for Games*, pages 53–57. ACM Press, 2002.

[5] W. C. Feng, F. Chang, W. C. Feng, and J. Walpole. A traffic characterization of popular on-line games. *IEEE/ACM Transactions on Networking*, June 2005.

[6] V. Jacobson, C. Leres, and S. McCanne. `tcpdump`, 1989. ftp://ftp.ee.lbl.gov.

[7] S. McCreary and K. Claffy. Trends in wide area IP traffic patterns: A view from Ames Internet exchange. In *Proceedings of 13th ITC Specialist Seminar on Measurement and Modeling of IP Traffic*, pages 1–11, 2000.

[8] C. Partridge, P. P. Carvey, E. Burgess, I. Castineyra, T. Clarke, L. Graham, M. Hathaway, P. Herman, A. King, S. Kohalmi, T. Ma, J. Mcallen, T. Mendez, W. C. Milliken, R. Pettyjohn, J. Rokosz, J. Seeger, M. Sollins, S. Storch, B. Tober, and G. D. Troxel. A 50-Gb/s IP router. *IEEE/ACM Transactions on Networking*, 6(3):237–248, 1998.

[9] N. Sheldon, E. Girard, S. Borg, M. Claypool, and E. Agu. The effect of latency on user performance in Warcraft III. In *NetGames '03: Proceedings of the 2nd Workshop on Network and System Support for Games*, pages 3–14. ACM Press, 2003.

[10] B. S. Woodcock. An analysis of MMOG subscription growth - version 15.0. http://www.mmogchart.com/.