Effect of Network Quality on Player Departure Behavior in Online Games

Kuan-Ta Chen[1], Polly Huang[2], and Chin-Laung Lei[2]
ktchen@iis.sinica.edu.tw, *{phuang,lei}@cc.ee.ntu.edu.tw*
[1]Institute of Information Science, Academia Sinica
[2]Department of Electrical Engineering, National Taiwan University

***Abstract*—Understanding the impact of network conditions on player satisfaction, which is one of the major concerns of network game designers, is a popular research topic. Of the various ways to gauge user satisfaction, in this paper, we focus on how network quality affects *a player's decision to leave a game prematurely*. To answer this question, we analyze a $1,356$-million-packet trace from a large commercial MMORPG called *ShenZhou Online*.**

We show that both network delay and network loss significantly affect a player's decision to leave a game prematurely. It is feasible to predict whether players will quit prematurely based on the network conditions they experience. The proposed model can determine the relative impact of different types of network impairment. For our traces, the degrees of player intolerance of network delay, delay jitter, client packet loss, and server packet loss are in the proportion of 1:2:4:3 approximately. The model can also be used to make system design decisions. Through simulations, we show that by prioritizing server processing according to the goodness of network conditions, employing de-jitter buffers, or replacing TCP with a more lightweight transport protocol, the probability of premature departure can be significantly reduced. In this way, we demonstrate how our model of players' network experience provides feedback for the design of online games.

***Index Terms*—Departure Analysis, Internet Measurement, Logistic Regression, MMORPG, Quality of Service, User Behavior**

I. Introduction

Of the various research areas related to online games, assessing the impact of network conditions on user experience is one of the most popular topics. Many studies, e.g., [3, 4, 7, 8, 16, 18, 25, 27, 28, 33, 37, 39], try to answer questions like: *Are game players sensitive to network conditions?* If the answer is yes, they ask: *What level of network QoS (Quality-of-Service) should be provided to maintain a satisfactory gaming experience?* The answers to the above questions are important because they could provide useful guidelines for the trade-offs in network resource planning. For instance, if we can be sure that players are less tolerant of large delay variations than high latency, then providing a smoothing buffer at the client side, which introduces additional latency but smoothes the pace of game play, would be a plus, as it still improves the overall gaming experience from the user's perspective.

Currently, there is no standard way to *objectively* quantify the satisfaction that players derive from gaming. Hence, the effect of network quality is often evaluated in terms of the users' performance in a specific context, such as the number of kills in shooting games, the time taken to complete each lap in racing games, or the capital accumulated in strategy games. However, game scores are highly dependent on a player's skills, the system design, and the game's content, so the results are not comparable and generalizable across different games. On the other hand, according to flow theory in psychology, game playing can be described as a pleasurable and exciting activity that makes players oblivious to time while they are in the game [21, 26]. The theory suggests that players will be more conscious of the real world if the feeling of involvement in the virtual world is diminished by network lags; therefore, the effect of *time distortion* will be mitigated. Furthermore, players may simply decide to quit a game as soon as they detect unacceptable lags. Thus, we conjecture that *the time players leave a game is affected, to some extent, by the network quality they experience*.

Massively Multiplayer Online Role-Playing Games (MMORPGs) have become immensely popular in recent years, with several top games reporting millions of subscribers [36]. Our conjecture is verified by real-life traces from a commercial MMORPG, *ShenZhou Online* [35], for two reasons. First, MMORPGs are deemed to be *addictive* in that about half of the players consider themselves addicted [38], so they tend to stay for a long time once they join a game. For instance, the statistics of MMORPGs in Japan [1] show that the average game session time is between 80 and 120 minutes. Most players stay for more than an hour once they join a game. If players leave in the first few minutes, it may indicate that they have an unsatisfactory gaming experience due to poor QoS. The second reason is that MMORPGs are relatively slow-paced compared to other popular genres, such as first-person shooting (FPS) games, which usually require players to make sub-second decisions. Slow-action games undoubtedly have less stringent service requirements than fast-action games. Thus, MMORPGs could be seen as a baseline for real-time interactive games so that if network QoS frustrates MMORPG players, it should also affect gamers of other genres.

In this paper, we analyze the player departure patterns in *ShenZhou Online* and their relationship to network quality. We find that both network latency and network loss have a significant influence on players' decisions to leave a game *prematurely*. We detail our major findings in the following question-and-answer format.

1) **Do game players leave a game prematurely due to unfavorable network conditions?** Yes. Generally speaking, the worse the network quality, the earlier

players will leave the game. For example, sessions with a low packet loss rate ($\leq 1\%$) have an average duration of 160 minutes, while those with a high packet loss rate ($> 1\%$) have an average duration of 70 minutes. If we only observe whether players quit in the first 10 minutes of a game, only 3% of players who experience a low loss rate leave in that time, compared to 20% of players who experience a high loss rate.

2) **Is it possible to predict whether a player will still be online at a given instant?** Yes. Using a logistic regression approach [20], we show that it is possible to predict premature departures based on the network conditions the players experience. In our traces, the network quality can explain 32% of the variability of a player's decision to leave a game after playing for 10 minutes, as shown by the following equation:

$$\begin{aligned} lp &= 12.5 \times \text{rtt.mean} + 86.1 \times \text{rtt.sd} + \\ &\quad 1.1 \times \text{log(closs)} + 1.2 \times \text{log(sloss)}, \end{aligned}$$

$$\Pr[\text{stay} < \text{10min}] \;=\; \exp(lp)/(1+\exp(lp)),$$

where $rtt.mean$, $rtt.sd$, $closs$, and $sloss$ stand for average RTT (round-trip time), standard deviation of RTT, client packet loss rate, and server packet loss rate, respectively.

3) **What is the relative influence of different kinds of network impairment?** Quantitatively, the degree of player "intolerance" to network delay, delay jitter, client packet loss, and server packet loss is in the proportion of 1:2:4:3 approximately. In other words, if players quit because they are frustrated by unfavorable network conditions, on average, 10% of their dissatisfaction is caused by network latency, 20% by network delay jitter, 40% by client packet loss, and 30% by server packet loss. These findings also suggest that, while current network games rely primarily on a "ping time" to select a server for a smooth game, delay jitter should also be considered in the server selection process.

4) **Is it possible to encourage players to remain with a game based on predictions about departure?** Yes. We first show that *the player departure rate declines over time*; that is, the longer a player stays, the less likely s/he is to leave the game any time. Moreover, we find that *the influence of network quality on players declines over time.* This may be because extraneous factors, such as social bonds, affect a player's decision to leave when s/he has been in a game for a period of time. Having considered both properties, we can encourage users to remain with a game by *temporarily allocating more resources to players who tend to leave prematurely* until they settle down and become tied to the game play. We show via simulations that, by prioritizing server processing according to the goodness of network conditions and employing de-jitter buffers, the probability of premature departure can be reduced by 10% and 4% respectively.

5) **Can we provide a better gaming experience by improving transport protocols?** Many MMORPGs, including the game we studied, use TCP as the underlying transport protocol because of its reliable and ordered transmission mechanism. However, TCP may cause performance degradation, as stream-oriented delivery is not actually required for every game message. With TCP, a single dropped packet causes a stall in the transmission of subsequent network data until that packet is successfully delivered. In our traces, the delay jitters increased from an average of 18 ms to 32 ms due to the TCP's in-order delivery policy. Based on the model developed in Section V, we estimate that the odds of premature departure (defined as a player quitting a game within 10 minutes of joining it) would be reduced by a factor of 2.8 if the additional delay jitter due to in-order delivery could be avoided. This corresponds to a 12% reduction in the premature departure probability (from 20% to 8%) in our case.

The remainder of this paper is organized as follows. Section II describes related work. We briefly introduce the studied game and summarize the collected traces in Section III. In Section IV, we analyze the player departure patterns and their correlations with network QoS. In Section V, we develop a logistic model that describes the relationship between QoS factors and premature departures. We discuss the model's implications and applications in Section VI. Then, in Section VII, we present our conclusions.

II. Related Work

Although a QoS infrastructure is not widely available on the Internet, real-time interactive online games, which are generally considered QoS-sensitive, are becoming increasingly prevalent. The reason could be that either QoS is not important, or players have simply become accustomed to unfavorable network conditions. A number of *experimental* studies based on users' performance in controlled network environments have addressed this question [4, 25, 28, 32, 33, 37, 39]. For example, Beigbeder et alfound that typical ranges of packet loss and latency do not significantly affect the outcome of the game *Unreal Tournament 2003* [4], while Sheldon et al. concluded that, overall, high latency has a negligible effect on the outcome of *Warcraft III* [33]. However, Nichols and Claypool showed that user performance is degraded by almost 30% when latency is higher than 500 ms in *NFL Football* [25].

Meanwhile, some studies have explored the problem using an *observational* approach [3, 8, 16]. Henderson found that the effect of network delay is outweighed by game design or exogenous effects, and players seem to be remarkably tolerant of network conditions [16]. Armitage suggested that players prefer a *Quake 3* server with a ping time that is less than 150 to 180 ms from their locations [3]. In a previous work [8], we proposed estimating players' awareness of network quality by the amount of time they spend in a game. We found that game session times are closely related to the network conditions players experience, and we derived the players' intolerance to various types of network impairment, e.g., latency, delay jitter, and packet loss. The present study extends our work in [8] by incorporating 1) the *variability* of players' QoS-sensitivity during a game, instead of treating it as constant; 2)

Fig. 1. A screen shot of *ShenZhou Online*

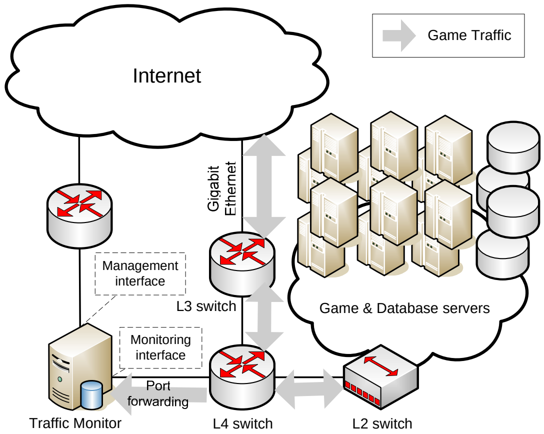


Fig. 2. Network setup for traffic measurement

predictability analysis of player departure events in terms of network quality; and 3) the representativeness and sampling methods of network QoS factors. We also explain how the proposed regression model can be used in making system design decisions.

While a number of previous works have suggested remarkable QoS tolerance on the part of game players [4, 16, 33], our findings based on the *ShenZhou Online* trace show that network quality has a significant influence on players' departure patterns. We believe that the discrepancy is due to both the nature of the game genre and the design and implementation of each particular game, such as dead reckoning schemes [2, 29, 34], and transport protocols. For example, TCP provides in-order delivery, which incurs additional delay and jitter for each packet loss event. As games employ different designs and transport protocols, it is inevitable that players will have diverse levels of QoS-sensitivity in different games, unless we can *separate* the effects of network QoS, system design, and transport protocols on players. This issue remains to be solved.

III. Trace Collection

ShenZhou Online is a mid-scale, commercial MMORPG that is popular in Taiwan [35], where there are thousands of players online at any one time. To play, the participants purchase game points from a convenience store or online. A screen shot of *ShenZhou Online* is shown in Fig. 1. The character played by the author is the man under the tree with a round smiling face above him. He is in a typical market place, where other players keep stalls. As is normal in MMORPGs, a player can engage in fights with random creatures, train himself in special skills, participate in marketplace commerce, or take on a quest.

With the help of the *ShenZhou Online* staff, we set up a traffic monitor beside the game servers. The monitor was attached to a layer-4 switch upstream of the LAN containing the game servers (we call it the "game LAN"). The port forwarding capability of the tapped layer-4 switch was enabled so that a copy of all inbound/outbound game traffic was forwarded to our monitor. To minimize the impact of monitoring, all remote management operations were conducted via an additional network path, i.e., the game traffic and management traffic did not interfere with each other. The network topology and setup of the game servers and the traffic monitor are shown in Fig. 2. The traffic monitor was a FreeBSD PC equipped with 1.5 GHz Pentium 4 and 256 MB RAM. We used *tcpdump* with the kernel built-in BPF to obtain traffic traces. In each trace, we randomly chose a subset of game sets, and only packets belonging to the selected game sets were logged. A game set, which is logically a "game server" from a player's viewpoint, comprises an entry server, several map servers, and a database server. All game sets are equivalent in content, but *isolated*. The reason for providing identical game sets is to distribute the players over a number of servers with limited game content, e.g., terrain, missions, and creatures in the virtual world. We took two packet traces, $\mathcal{N}_1$ and $\mathcal{N}_2$, which recorded traffic for two and three game sets, respectively. The two traces, which spanned 8 and 12 hours, respectively, and contained more than $1,356$ million packets, are summarized in Table I. Interested readers may refer to [6] for more details about the characteristics of game traffic. The full data set is available for research purposes on request[1].

Although the traced game servers are centrally located at one ISP, players are spread over 13 countries and hundreds of autonomous systems. More specifically, the average RTTs experienced by game sessions range from 95 ms to 580 ms, and the loss rates incurred range from zero to 20% (computed by one percentile and 99 percentile, respectively). The heterogeneous network path characteristics manifest that our trace is not specific to a particular configuration.

IV. The Player Departure Process and Its Sensitivity to Network QoS

In this section, we first analyze the general departure process of game players without considering the effect of network conditions. We then present a correlation analysis of the relationship between the player departure process and the network conditions they experience. The purpose of the correlation analysis is twofold: 1) to confirm the influence of

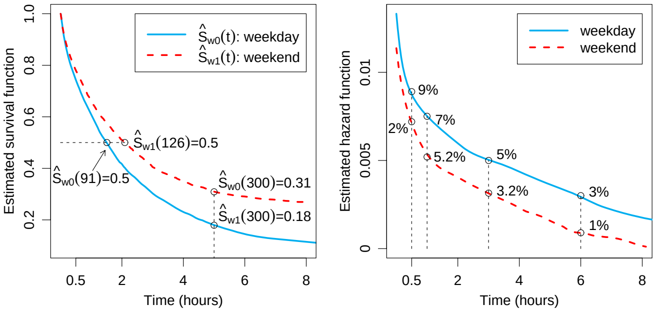
[1] Please visit *http://mmnet.iis.sinica.edu.tw/download.html* to request the *ShenZhou Online* traffic traces.

TABLE I
SUMMARY OF GAME TRAFFIC TRACES

Trace	Sets	Date	Time	Period	Drops[†]	Conn.	Session	Pkt. (in / out / both)	Bytes (in / out / both)
$\mathcal{N}_1$	3	8/29/04 (Sun.)	15:00	8 hr.	0.003%	$57,945$	$7,597$	342M / 353M / 695M	4.7TB / 27.3TB / 32.0TB
$\mathcal{N}_2$	2	8/30/04 (Mon.)	13:00	12 hr.	?[‡]	$54,424$	$7,543$	325M / 336M / 661M	4.7TB / 21.7TB / 26.5TB

[†] This column gives the kernel drop count reported by *tcpdump*.

[‡] The drop count reported by *tcpdump* is zero, but we actually found some packets are dropped at the monitor.



(a) Survival functions

(b) Hazard functions

Fig. 3. Estimated hazard functions and survival functions for the observed game sessions. The right-hand graph shows the probability estimate that a player who has already played for time t will leave the game within the next 10 minutes. Note that, on the graph, a probability of 5% is denoted as 0.005, which is multiplied by 10 because we are considering the departure probability within the next 10 minutes instead of 1 minute.

network QoS on players' premature departure patterns; and 2) to serve as a quick way to identify factors that significantly influence player behavior.

A. General Player Departure Pattern

We now investigate the general pattern of how much time players invest in game playing. First, we consider the estimated survival functions [23], which are semantically equivalent to complementary cumulative distribution functions (CCDF), for sessions on a weekend and a weekday respectively. As shown in Fig. 3(a), the median game session time is 127 minutes and 92 minutes for the weekend and weekday, respectively. This supports the common intuition that people generally have more time for leisure activities on weekends than on weekdays. We can highlight this difference in another way: while 31% of players spend more than 5 hours in the game on a weekend, only 18% of players do so on a weekday. Furthermore, both survival curves are concave upwards (i.e., convex functions), which indicates that *players tend to stay longer when they have been in the game for a long time*.

The hazard function provides us a more direct way of checking the departure "rate" of participants. The function, also known as the conditional failure rate in reliability engineering, or the intensity function in stochastic processes, is defined by

$$h(t) = \lim_{\Delta t \to 0} \frac{\Pr[t \le T < t + \Delta t | T \ge t]}{\Delta t}.$$

The hazard function gives the *instantaneous rate* at which failures occur for observations that have survived at time t. In our context, the quantity $h(t)\Delta t$ can therefore be seen as the *approximate probability* that a player who has been in a game for time t will leave the game in the next Δt period, given that Δt is small.

We illustrate the estimated hazard functions in Fig. 3(b). Both functions present continuous downward trends. The hazard function on weekdays shows that the probability a player will leave a game within a short time (10 minutes) when s/he has been in the game for 30 minutes, 1 hour, 3 hours, and 6 hours is approximately 9%, 7%, 5%, and 3%, respectively. The decreasing failure rate indicates that *a player who has played a game for a longer time has lower probability of leaving any time*, which is a remarkable feature of heavy-tailed distributions [31].

The distribution of the observed game sessions was very different from that reported in earlier studies of FPS (First-Person Shooting) games [5, 13, 17], in which the session times were not heavy-tailed. We attribute this discrepancy to the difference in game genres. First, since FPS games are round-based, players are forced to take a break after each round so that they have a chance to regain consciousness of the real world. In contrast, the adventures in MMORPGs are continuous and endless, and no explicit mechanism exists to give players a pause. According to flow theory, the time distortion effect is more significant when a player is more involved in the virtual world [26], i.e., players lose their sense of time and therefore tend to spend more time in the virtual world. Second, MMORPG players are likely to be locked into the game by "social bonds." For example, a player may endeavor to stay in a game until the current mission is completed, because leaving prematurely would affect the whole group's adventure and possibly damage his/her reputation in the game. While team playing is also common in FPS games, the social bonds tend to be short-lived because of the games' round-based nature.

B. Correlation Analysis

Before evaluating the degree of untimely logouts from a game owing to unfavorable network conditions, we define "premature" player departure.

Definition 4.1: A *premature departure* occurs at time t, denoted as $PD(t)$, if a player leaves the game before playing for t minutes. The time span t is called the *observation period*.

In the following, we check if different levels of network impairment (i.e., network delay, delay variations, and packet loss) annoy players and cause them leave the game earlier than they would if the network quality were "perfect." Below, we define six QoS factors that could be relevant to gamers' premature departures.

- *Average RTT*: the average round-trip transmission latency of game data packets, which is a measure of *interactivity* and *responsiveness* of game play.

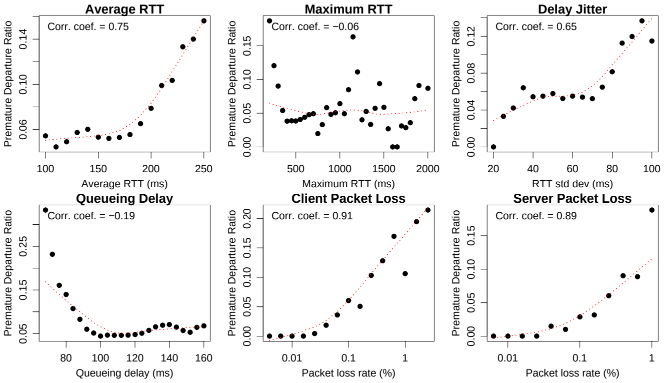


Fig. 4. Correlation of premature departures and network QoS factors

- *Maximum RTT*: the maximum round-trip transmission latency of game data packets, which accounts for the most unpleasant “lag” experience.
- *Delay jitter*: defined as the standard deviation of packet round-trip times, which measures the *instability* of the game’s response time.
- *Average queueing delay*: computed as the average round-trip time minus the minimum round-trip time, which is an estimate of the average queueing time accumulated during network transmission.
- *Client packet loss rate*: the loss ratio of packets sent to the server by game clients, which accounts for the additional latency before a player’s command can be processed (by the server), as loss detection and recovery cost some time.
- *Server packet loss rate*: the loss ratio of packets sent from the game server to the client, which accounts for additional latency before game messages or state updates can be displayed on the client’s screen (i.e., presented to the game player).

Our procedure for assessing whether a particular QoS factor affects the occurrence rate of premature departure events is as follows. First, the range of each QoS factor is divided into several equal intervals. Then, we classify all the game sessions into different groups according to which interval their QoS factors fall into, and compute the *proportion of prematurely departed sessions within each group*. The computed quantity $PD_{ratio}(x)$ is an approximation of the conditional probability $\Pr(\mathrm{PD}|x)$, where x denotes the midpoints of the intervals of a QoS factor. The scatter plots for PD(10), i.e., premature departures that occurred within 10 minutes of joining a game, which represent the relationship between $PD_{ratio}(x)$ and x, are shown in Fig. 4. For each plot, we use Kendall’s rank correlation coefficient τ [24] to quantify the strength of the relationship between the QoS factors and the ratio of premature departure events. A lowess smooth curve [9] is also plotted to facilitate visual detection of the trend.

1) Factor Analysis: In Fig. 4, except for the maximum RTT and the mean queueing delay, the factors show generally positive correlations with the rate of premature departures. This basically confirms our hypothesis that more serious network impairment annoys players such that they are likely to leave the game earlier (even though they may come back later).

Effect of Queueing Delay: The average queueing delay, however, has negative correlations with premature departures when it is small, and shows no correlation with premature departures when it is moderate to high. A detailed analysis reveals that this is because sessions with short queueing delays have much higher packet loss rates than those with long queueing delays. Specifically, the median packet loss rate for sessions with queueing delays shorter than 50 ms is 0.84%, but for higher-queueing-delay sessions it is 0.08%, a ratio of approximately 10:1. The combination of high packet loss and short queueing delay could be due to certain congested links that incur a high packet drop rate; however, since the capacity is high, the queueing time is relatively short (the queueing time is decided by both the queueing length and the outgoing link bandwidth). On the other hand, there is no correlation between moderate to long queueing delays and premature departures. This suggests that queueing delay is not a good indicator of network quality, as it does not directly affect players’ perceptions of game responsiveness and interactivity. In other words, players cannot distinguish between specific components of the delay time (i.e., processing delay, propagation delay, transmission delay, and queueing delay). Instead, they only care about the total delay time that they actually experience in the form of game “lags,” “jumps,” slow responses, or inconsistent states between different peers.

Effect of Maximum RTT: There is no correlation between maximum RTT and premature departures. This may because the maximum RTT captures the worst network lags players experience during the session, instead of the players’ average experience. Even if the worst lag is intolerable, users may be patient and wait for conditions to return to normal (as network quality changes constantly over time). In this case, the maximum RTT factor cannot capture the true feelings of players based on premature departures.

Threshold Effect: The trend of the lowess curves in Fig. 4 indicates that the average RTT, delay jitter, and both packet loss factors have a “threshold” effect, i.e., the impact of a factor remains unchanged when its magnitude is small. For example, the threshold of the average RTT is around 180 ms, so the premature departure probability only increases with the average RTT when the latter is higher than 180 ms. This indicates that players may be insensitive to a small amount of network impairment. The threshold effect is commonly seen in measures of physiological reactions to external substances [11]. For example, human responses to drugs in terms of enzyme activity, membrane potential, heart rate, or muscle contraction usually have a threshold effect. Hence, the threshold effect we identified here could be seen as evidence that premature departures successfully capture players’ perceptions of network impairment.

2) Effect of the Observation Period: Fig. 4 shows the effect of network QoS factors on premature departures with an observation period of 10 minutes. We now examine whether the effect of network impairment remains the same with different definitions of premature departure. To do so, we plot the rank correlation coefficient between premature departures and QoS factors with different observation times, as shown in Fig. 5.

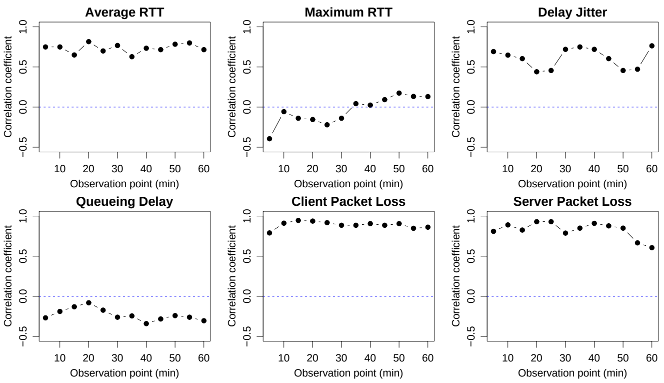


Fig. 5. Correlation of premature departures and network QoS factors

Of the six factors, the average RTT, delay jitter, and packet loss rates in either direction all show strong correlations with the premature departure ratio, regardless of the observation points, as their correlation coefficients are consistently higher than 0.5.

In contrast, the maximum RTT has a very weak and unstable correlation (τ is between -0.4 and 0.2). Also, the queueing delay has a unreasonable negative correlation. We discuss this point in the subsection entitled "Effect of Queueing Delay". As neither factor has a consistent relationship with premature departures, we exclude queueing delays and maximum RTT from our considerations hereafter.

3) Summary: Although the correlation analysis presented in this section reveals the effect of network conditions on players' departure times, it cannot *quantify* the full impact of individual QoS factors exactly because of *the collinearity among the factors*. For instance, the correlation coefficient between the client packet loss rate and server packet loss rate is strong (0.73); while the average RTT and delay jitter have non-trivial correlations with the overall packet loss rates (0.2 and 0.15 respectively). Given that these factors have significant positive associations with premature departures, we still need to determine which one causes the most user dissatisfaction. Players may be particularly unhappy because of one factor, or they may be sensitive to all of them, with different levels of intolerance. To determine the effect of individual QoS factors, in the next section, we perform regression analysis, which models each QoS factor as a predictor of the probability of premature departures.

V. Modeling The Probability of Premature Departures

In this section, for the sake of clarity, we assume an observation time of 10 minutes. The effect of different observation times is discussed in Section VI-B. We begin by describing the logistic regression model, after which we discuss some issues related to the model's development, including the sampling of QoS factors, adjustment of factor forms, predictability analysis, and validation. We conclude this section with an interpretation of the developed model.

A. Logistic Regression Model

For each game session, we now have two sets of data: 1) a set of network QoS factors, which measure the network impairment the session experienced and serve as *predictors*; and 2) the record of whether a premature departure event occurred (yes or no), which serves as the *response* variable, therefore it seems appropriate to apply ordinary linear regression modeling. However, our case is not suitable for linear regression because the required conditions for linear regression, including the normality of errors and homoscedasticity of variance, are violated. Moreover, dichotomous response variables are difficult to deal with because they have ceilings and floors. That is, if we treat the probability of premature departures as the response variable in an ordinal linear regression, we will always obtain a negative "probability" with sufficiently small predictors, and a "probability" above one with sufficiently large predictors. How to interpret these nonsensical probabilities is a problem.

For the above problems, we apply logistic regression [10, 20] to model the impact of network QoS factors on premature departure events. The logistic regression model, which belongs to a class of models known as generalized linear models [15], is one of the most popular methods for predicting the probability of the occurrence of an event by data fitting. It resolves the above-mentioned problems by incorporating binomial errors and a transformation of the linear predictor to the *logit*, i.e., the *logged odds*. The *odds* of a probability p are defined as $p/(1-p)$ so that the corresponding logit is $\ln(p/(1-p))$. Assume that the risk vector (the network QoS factors in our case) associated with a session i is $\mathbf{Z_i}$, then the logistic regression equation can be formulated as

$$P_i = \Pr(\mathrm{PD}|\mathbf{Z_i}) = \frac{\exp(L_i)}{1+\exp(L_i)} = \frac{\exp(\boldsymbol{\beta}^t\mathbf{Z_i})}{1+\exp(\boldsymbol{\beta}^t\mathbf{Z_i})}, \quad (1)$$

where P_i and L_i are the predicted probability and logit of premature departures for session i respectively, and $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_p)^t$ is the coefficient vector that corresponds to the "intolerance" for one unit increment of each QoS factor. In other words, the logit, which ranges from $-\infty$ to $+\infty$, rather than the probability, which ranges from 0 to 1, is used as the response variable in logistic regression. This explains why logistic regression is appropriate for predicting the probability of a certain event, as its response variable is always between 0 and 1 regardless of the magnitude of the predictors. To solve a logistic regression equation, the parameter vector $\boldsymbol{\beta}$ is usually estimated by maximizing the likelihood function

$$\prod\{P_i^{Y_i} \times (1-P_i)^{1-Y_i}\}, \quad (2)$$

where Y_i indicates whether a premature departure event actually occurred (0 or 1) in session i.

B. Sampling of QoS Factors

To ensure that the model is tractable, we use *a scalar value* for each risk factor to capture its effect on players' gaming experiences in a session. However, QoS factors, such as the round-trip delay time, are not constant, but keep changing during the game. Extracting a representative value for each

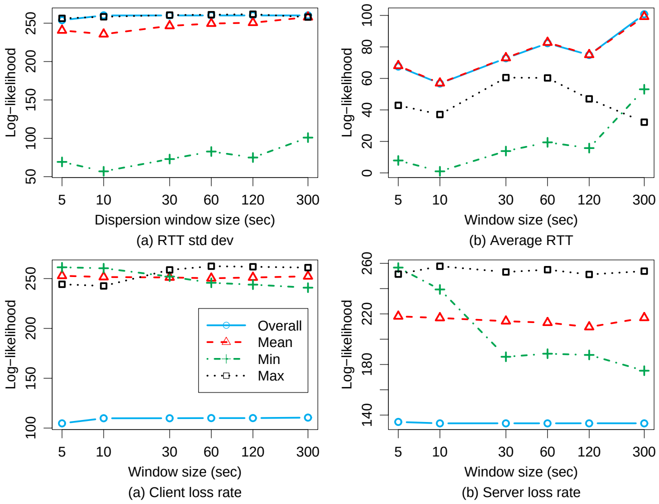


Fig. 6. Evaluation of the sampling method for network QoS factors

factor in a session, which is analogous to feature vector extraction in pattern recognition, is the key to determining how well the model fits the observed departure behavior of players.

Intuitively, the quantities averaged over the whole session time should be a good way to obtain a representative feature. However, *(relatively) extreme conditions* may have much more influence on users' overall perception than other conditions. For example, users may quit a game immediately because of serious network lags in a short period, but be unaware of mild and moderate lags that occur all the time. Moreover, players might be more sensitive to adverse network conditions than desirable network quality (i.e., as it is "supposed" to be), or vice versa. Players might still be happy if the network quality is satisfactory most of the time, even if it is is intolerable sometimes, or they may only consider the unsatisfactory part, and leave as soon as they feel the playing conditions are intolerable. These behavior patterns are only a few of numerous possible ways a player might react to network impairment. As no general and well-established perceptual and behavioral models exist to describe players' reactions to perceived network impairment, we investigate how to derive the most representative risk vectors.

We propose three measures to account for variations in network quality over time, namely, the minimum, the average, and the maximum of a factor, with two-level sampling. That is, the original time series s is divided into a number of sub-series of length w, from which network conditions are sampled. This sub-series approach is intended to confine the measures of network quality within time spans of length w, thereby *excluding the effect of large-scale variations*. The respective minimum, average, and maximum measures with lengths equal to $\lceil |s|/w \rceil$ are computed for each sampled QoS factor. We then decide which of the three measures is the most representative for describing a user's perceived experience during the game.

We evaluate different combinations of measures and window sizes by fitting the extracted QoS factors into a logistic model and checking the models' log-likelihood value, which is an indicator of goodness-of-fit. As Fig. 6 shows, the client packet loss rate and server packet loss rate are best sampled with an overall average in the whole session time, i.e., with $w = |s|$. We believe this result is due to the following reasons: 1) the game packet rate is low (generally less than 10 packets/second), and 2) packet loss is rare. Thus, a large window would be more appropriate because a short time series may not contain enough samples to capture the true packet loss probability along the network path.

On the other hand, the minimum values of the average RTT and the RTT standard deviation in consecutive windows are the most representative. That is, we choose the *minimum average RTT* and *minimum RTT standard deviation* and sample both with a window size of 10 seconds. (For simplicity, we use *delay* and *delay jitter* to refer to the sampled average RTT and RTT standard deviation variables respectively.) The small window size implies that players are more sensitive to *short-term, rather than long-term, effects of network quality*. This behavior is reasonable because long-term fluctuations in network quality should have no influence on the *real-timeliness* of game playing. The sampling method of the RTT standard deviation indicates that players are tolerant of infrequent extreme variations in network latency, and more sensitive to delay fluctuations that occur in every 10-second period. The sampling of both RTT-related factors consistently chooses the value that represents *the best (averaged) quality a player experienced*. This interesting finding could be further verified by cognitive models that explain why good experiences (rather than bad experiences) have a stronger effect on players' departure decisions.

C. Model Fitting

Since our two traces were recorded on a weekday and a weekend respectively, *the day of the week effect* should be incorporated into the modeling, if appropriate. For a 10-minute observation period, the proportion of premature departures on weekdays and weekends was 6.8% and 7.0% respectively, which yields an odds ratio of 0.96. This difference between the two groups of sessions yields a p-value of 0.67 in Fisher's test [14], which fails to reject the null hypothesis that their odds are equal. Furthermore, if we take the binary variable $weekend$ as the only predictor in the logistic regression, both the Wald statistic and the likelihood ratio test indicate that $weekend$ is insignificant with a critical value of 0.2. All of these tests indicate that the day of the week does not cause players to leave a game prematurely. Compared to the discussion in Sec. IV-A, the phenomenon indicates that, although users generally spend less time playing games on weekdays, the time constraint on weekdays is not so stressful that players are forced to quit the game within a short time, e.g., 10 minutes. Thus, we do not include the $weekend$ variable in the model.

Like ordinary linear regression models, the logistic model assumes that *the contribution of each risk factor to the response variable is linear and additive on the logistic scale.* To check whether our QoS factors confirm this assumption, we fit the data into a generalized addictive model with smoothing

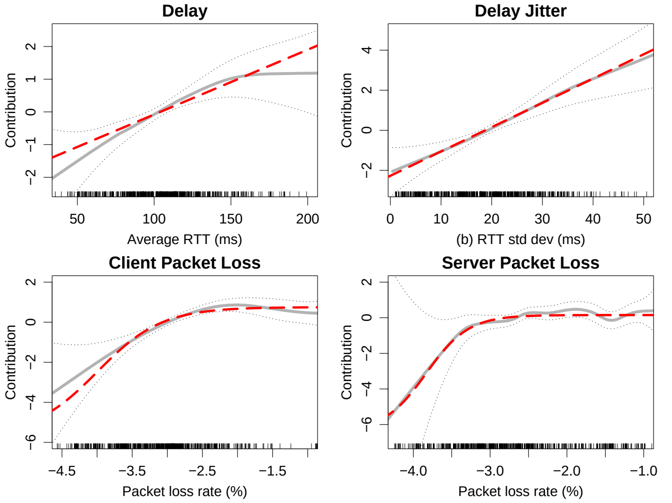


Fig. 7. The functional form of the four factors

TABLE II
COEFFICIENTS IN THE FINAL MODEL

Variable	Coef	Std. Err.	z	$P > \|z\|$
$delay$	12.49	2.11	5.93	0.00
$jitter$	86.14	6.35	13.57	0.00
$closs$	1.07	0.18	6.12	0.00
$sloss$	1.16	0.45	2.57	0.01

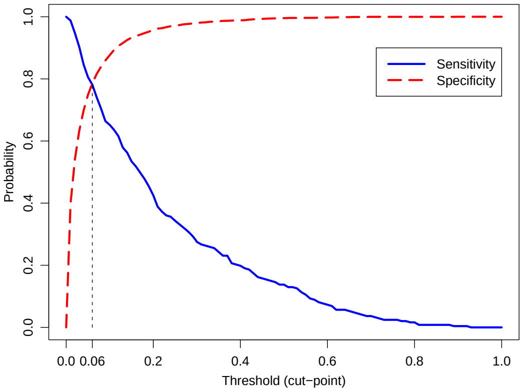


Fig. 8. Seeking a classification threshold for the estimated probability

splines [15]. The estimated impact of the four factors, as well as the two-standard-error confidence bands, versus their magnitude, are plotted in Fig. 7. The Y-axis represents the contribution of each factor to the logit of the response variable; specifically, it is an estimate of $\beta_X X$ of factor X with different magnitudes. A nonlinear relationship between the impact of and the magnitude of a factor X indicates that its coefficient β_X is not constant over all levels, which contradicts the assumption of the binary logistic regression model.

According to Fig. 7, both the delay and the delay jitter have approximately proportional impact on the premature departure probability; thus, no adjustment should be made. On the other hand, packet loss rates exhibit very different behavior compared to the delay-related factors. First, we observe that the influence of packet loss rates in either direction is not proportional to their magnitude. A common solution to modeling non-proportional variables is to use *scale transformation*. By taking logarithms, packet loss rates have a smoother influence on the logit of premature departures (indicated by the gray lines in Fig. 7(c)(d)), and yield much better goodness-of-fit. This indicates that the premature departure probability is more proportional to the *scale of packet loss rate*, than its magnitude. In other words, if we denote the impact of the loss rate p on premature departures as $imp(p)$, $imp(p) \propto p$ if the impact of the packet loss on premature departures is proportional, and $imp(p) \propto log(p)$ if the impact of packet loss is proportional to the scale of the loss rate. However, the logged packet loss rates still have a nonlinear impact on premature departures, and present a threshold effect. As a result, the impact of packet loss on premature departures seems to have an upper limit, instead of increasing unboundedly. This implies that players are already intolerant of moderate packet loss and tend to leave prematurely; thus, a higher packet loss rate would not cause further behavioral changes. We solve the disproportionality by mapping the logged packet loss rates to a logisitic equation (sometimes called the Verhulst model or logistic growth curve), which is commonly used to model a dose-response curve [11]. The general form of a logistic equation is defined as

$$f(t) = \frac{asym}{1 + \exp((xmid - t)/scale)},$$

where $asym$, $xmid$, and $scale$ are real parameters. The logistic equation has a sigmoid shape so that it can capture the threshold effect of our factors. From Fig. 7(c)(d), we observe that the logistic mapping from the packet loss rates to their impacts is reasonable in that the red lines are constantly within the 50% confidence band at all levels.

Finally, we obtain a fitted logistic regression model, as shown in Table II. To assess the overall goodness-of-fit, we use the Hosmer-le Cessie test [19], which reports $p = 0.40$, indicating that our model fits the data reasonably well.

D. Assessment of the Model Adequacy

Usually, the adequacy of a logistic model is assessed via a classification table, which is derived by classifying predicted probabilities by a cutpoint c. If the estimated probability exceeds c, it is assumed that players will quit the game prematurely. The most intuitive value for c is 0.5. However, this cutpoint does not usually yield good classification results, since the results are heavily dependent on the distribution of events, i.e., the proportion of premature departures that occurred. We can calibrate the model by choosing a cutpoint that maximizes both the sensitivity and the specificity of the classification. Fig. 8 plots the sensitivity and specificity obtained by using cutpoints in the range 0 to 1. Choosing $c = 0.06$ yields the minimum difference between two curves, where both the sensitivity and the specificity are equal to 78%. A more complete description of the classification accuracy is given by the C-index [22], which is equal to the area under the ROC (Receiver Operating Characteristic) curve [12]. The

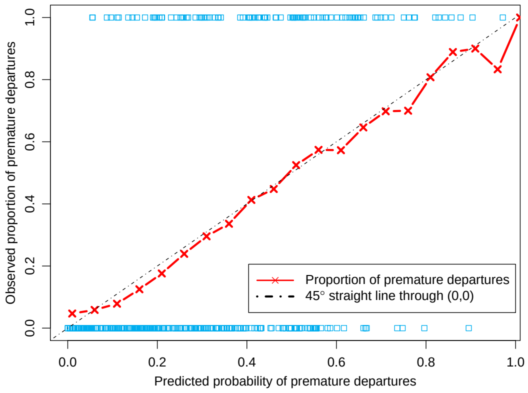


Fig. 9. Examination of the prediction accuracy of the developed model

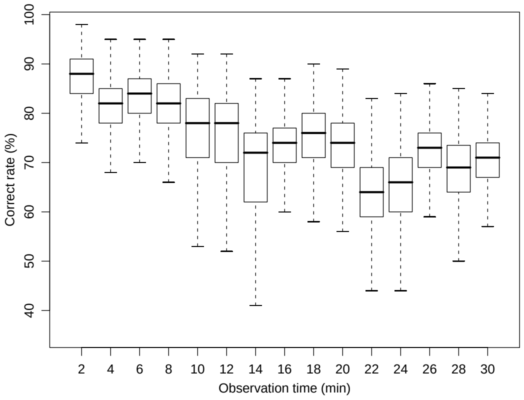


Fig. 10. Validation of model fitting with cross-validation

C-index for our model, 0.87, indicates generally good discrimination compared to a C-index of 0.5, which is equivalent to a random guess.

To demonstrate the predictive power of our model, we compare the observed proportion of premature departures and the predicted probabilities of premature departures, as shown in Fig. 9. The red crosses mark the proportion of premature departure events in each group, which is along the $45°$ straight line through the origin. The figure shows that the predicted probabilities match the actual probability well, which implies that the general prediction accuracy of our model is good.

E. Model Cross-Validation

Our model's prediction accuracy might be due to the fact that it actually captures the relationship between variables, or it might be due to *overfitting*. To confirm that the model does not overfit the data, we use cross-validation to further verify its adequacy.

The cross-validation steps are as follows: 1) randomly divide all the game sessions into two equal-sized groups: a modeling group and a validation group; 2) fit a logistic model with the modeling group; 3) predict whether premature departure events have occurred for the validation group based on the fitted model, and compute the prediction accuracy; and 4) repeat steps 1–3 one hundred times.

Fig. 10 shows the cross-validation results. The prediction accuracy differs according to the length of the observation time. The median correct rates are generally higher than 70% for observation times shorter than 30 minutes, and higher than 80% for times shorter than 10 minutes. We find that the correct rates in the worst cases can be quite low in some scenarios, e.g., with the observation time of 14 minutes. We attribute this phenomenon to the high variability of game session times. In addition to the quality of network conditions, there are many exogenous factors that could affect players' decisions to continue with a game or leave it. For example, players may be tied by quests on hand or social bonds, even when network conditions are poor and screen updates are jerky. On the other hand, they may leave a game because of prearranged events, schedule constraints, or physical conditions. Although the worst-case prediction performance in the model's cross-validation is not good, overall, the median correct rates are acceptably high, especially when the observation time is shorter than 10 minutes. This demonstrates that the model's accuracy is not a consequence of data overfitting.

F. Model Interpretation

In Table II, we present the estimated coefficients along with their standard errors and p-values for the fitted logistic model. All variables are significant at a significance level of 0.1. The coefficients of the model can be interpreted by *odds ratios*. Since the magnitude of a coefficient implies a change in the response (in the logit scale) for a one-unit increase of the covariate, the odds ratio between two risk vectors can be obtained by exponentiating their difference in logit form. For example, assume that two players experience similar network conditions, except for delay jitter of 20 ms and 10 ms respectively. The odds ratio of these two sessions can then be computed by $\exp((0.02 - 0.01) \times 86.14) \approx 2.4$, where 86.14 is the coefficient of the covariate $jitter$. That is, the odds that player A will leave the game prematurely are 2.4 times higher than the odds that player B will leave prematurely (i.e., quit the game within 10 minutes of joining).

VI. Model Implications and Applications

In this section, we first discuss the implications of our analysis results for other game genres. We then present a predictability analysis of players' premature departure behavior. Next, we investigate the relative impact of various types of network impairment on user perception. We conclude the section by discussing how our model can be used to improve system design, in terms of server processing scheduling, dejitter buffer dimensioning, and the choice of transport protocols.

A. Implications for Other Game Genres

MMORPGs are slow-paced compared to other popular genres, such as first-person shooting games, which require players to make sub-second decisions. In addition to a game's pace, there is a great deal of difference in how players control the virtual characters. In fast-action games like shooting, players instruct characters "what" actions to take and "how" to

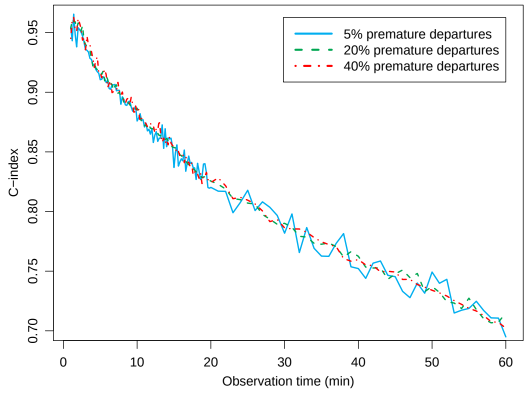


Fig. 11. Predictability of players' departures vs. observation time

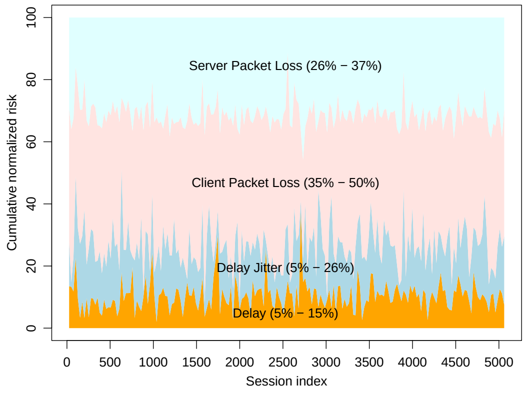


Fig. 12. Relative influence of different QoS factors in each session

perform those actions. Specifically, to move a character to a new location, a player must control each step the character takes (e.g., three steps west followed by five steps north). In contrast, in slow-action games like MMORPGs and real-time strategy games, players only instruct characters "what" to do, i.e., they only need to point out the location the character should move to, and it will automatically move toward the destination via a route that is either pre-determined or computed on-the-fly. MMORPGs are classified as slow-action games, which have less stringent service requirements than fast-action games. Therefore, as our analysis indicates that poor network QoS frustrates MMORPG players, it is reasonable to assume that it will also affect players of other online game genres that run at a faster pace.

B. *Player Predictability*

We define predictability as the degree of association between players' departure times and the network conditions they experience. The stronger the association, the easier it is to predict whether a premature departure event will occur within a specific period. This prediction analysis is motivated by the question: *What is the best time to predict whether players will leave a game prematurely?*

We evaluate the effect of the observation time on the predictability of premature departures by the C-index of the fitted logistic model. To ensue that the C-index is comparable between models with different observation times, we randomly remove a few game sessions so that all game sessions have a fixed proportion of premature departure events, say, 5%. As shown in Fig. 11, player predictability, not surprisingly, *constantly decreases with longer observation times*. The downward trend to the right indicates that departure events are less predictable for players who have stayed longer. One reasonable explanation could be the addictive feature of MMORPGs. Once players have been immersed in the virtual world for a while, they may enter the *flow state* [26] such that the effects of network impairment are mitigated. Another explanation could be that, as players have been in the game for a long time, extraneous factors, such as schedule constraints, tiredness, or social interaction, have a significant effect on their decisions to stay or leave. For example, players may want to stay in the game until they complete the current mission because they will lose all the rewards if they quit before completion; or, they may hesitate to leave because of "social bonds," as current game partners may not be online at other times.

Returning to our question about the best time to predict premature departures, we can say that *the earlier the observation is made, the better the prediction accuracy will be.* As a general rule, we consider that an observation time shorter than 20 minutes is reasonable because the resulting predictability is acceptably high (i.e., the C-index is higher than 0.8).

C. *Impact of QoS Factors*

Our model for premature departures also enables us to quantify *the relative influence of QoS factors*. The influence of a QoS factor, X, is computed as follows:

1) compute the risk score vector L with risk vectors $\mathbf{Z}$;
2) compute the risk score vector L_X with risk vectors $\mathbf{Z_X}$, where the factor X_i for each session i is set to $\min(X_i)$;
3) compute the relative influence of X as $L - L_X$, and normalize it by a total score of 100.

The computed relative influence of each QoS factor is shown in Fig. 12. On average, the degrees of players' "intolerance" to delay, delay jitter, client packet loss, and server packet loss are in the proportion 1:2:4:3. That is, a player's decision to leave a game prematurely due to unfavorable network conditions is based on the following levels of intolerance: average RTT (10%), RTT variations (20%), client packet loss (40%), and server packet loss (30%). Next, we consider the implications of these ratios.

- *Delay jitters are less tolerable than absolute delays*. While most earlier QoS-sensitivity studies completely neglected the impact of delay jitters, we argue that jitters are relevant to players' online gaming experiences. This also suggests that, while current network games rely primarily on "ping time" to select a server for smooth game play, delay jitters should also be considered in the server selection process. However, measuring delay jitter requires more time and network resources than measuring ping times; thus, how to balance the resources spent in probing network conditions and the reliability of measurement results merits further investigation.

- *Packet loss is much less tolerable than packet delay*. Comparing the overall influence of network latency and network loss, we obtain a ratio of 3:7. This result is not consistent with an earlier study of *Unreal Tournament 2003* [4], where the authors reported that network latency $<$ 200 ms and network loss $< 6\%$ have a statistically weak impact on user performance. We believe this discrepancy is due to the different *transport protocols* employed.

  While most FPS games use UDP to exchange information between game peers, many MMORPGs, including *ShenZhou Online*, use TCP. Since TCP provides in-order delivery and congestion control, a lost packet will cause subsequent packets to be buffered until it is successfully delivered, which reduces TCP's congestion window. On the other hand, packet loss does not incur any overhead in UDP. Thus, in TCP-based games, packet loss incurs *additional packet delay and delay jitters*, both of which further degrade players' gaming experience. We discuss the effect of transport protocols in the next subsection.
- *Client packet loss is slightly less tolerable than server packet loss*. We consider this to be reasonable, since client packet loss delays the players' commands to the server, whereas server packet loss delays responses to the commands as well as state updates. Current MMOGs are mostly server-centric, so a player's command is not effective until it has been processed by the server. In addition, to speed up the responses to players' commands, game clients may "cheat" by displaying the expected states in response to players' own commands on receipt of players' inputs before those inputs validated by the server. Thus, server packet loss only impacts on the consistency between players' views of the virtual world, not the responsiveness to players' inputs. As a consequence, client packet loss, which may delay the players' commands, such as attacks and spell casting, is more annoying than server packet loss, which just delays the server's responses and screen updates.

D. Impact of Transport Protocols

Although TCP is generally considered to be unsuitable for interactive and real-time communications, many MMORPGs adopt it as their underlying transport protocol. One reason is that TCP is *stream-oriented*, so that the message stream at the sender will be identical to the stream received at the destination. This property allows game developers to focus on game development, and leave issues related to network transmission to TCP. However, TCP can degrade message transmission efficiency because the stream-oriented feature is not required for each game message exchange; hence the protocol's in-order and reliable delivery might lead to overkill sometimes.

To quantify the degradation of game message transmission, we estimate the *in-order delivery overhead* in terms of *the additional delay jitters incurred*. We believe that *in-order delivery is not necessary for all game messages* for the following reasons:

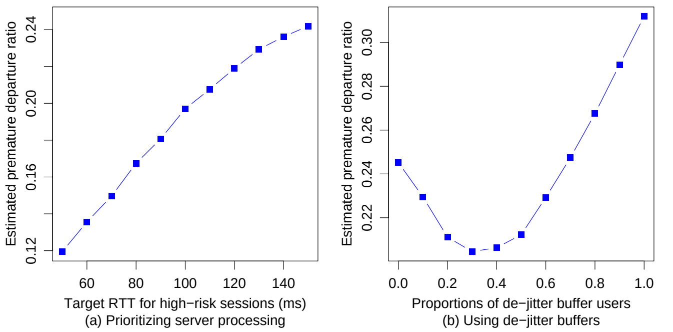


Fig. 13. Reducing the probability of players' premature departure by providing more resources for high-risk sessions to ensure shorter delays or less delay jitters.

- Many game messages are *accumulative* in nature, i.e., subsequent messages will override earlier ones. For example, state updates, especially position updates, are usually accumulated so that a missing message does not matter, unless it is the last in a series of updates. Thus, a series of accumulated commands, except for the last, could be delivered in an unreliable and out-of-order manner.
- Some game messages can be processed *in any order*. For example, server packets are primarily comprised of accumulated state updates, dialogue messages, and responses to queries, such as information about virtual items. With the exception of dialogue messages, server packages can usually be processed in any order.

To assess how much additional delay jitters are induced by enforced packet ordering, we assume an extreme case where game packets can be processed in any order. For our traces, the average delay jitters are estimated in two ways: 1) 30 ms if delays induced by retransmitted packets are considered; or 2) 18 ms if delays induced by retransmitted packets are *not* considered. Based on the premature departure prediction model developed in Section V, we estimate that the odds of premature departure would be reduced by a factor of $\exp((0.030 - 0.018) \times 86.14) \approx 2.8$ if additional delay jitters could be eliminated. This corresponds to a 12% decrease in the premature departure probability (from 20% to 8%) with an observation time of 30 minutes. In this way, we can estimate the degree of improvement if we replace TCP with a more lightweight protocol that only orders packets when necessary. Also, the result explains why packet loss generates so much more intolerance among MMORPG players than FPS players (Section VI-C).

E. Improving the Gaming Experience

In Section IV, we showed that the player departure rate generally decreases over time; that is, the longer players remain in a game, the less likely they are to leave the game in every instant. Furthermore, in Section VI-B, we showed that the relative influence of network impairment decreases over time, as extraneous factors, such as social bonds, gradually outweigh the effect of network QoS on players' decisions to continue or leave a game. By combining both properties, we propose a strategy that makes a game more sticky by *temporarily allocating more resources to players who have just joined a game and have a higher probability of leaving prematurely due to unsatisfactory network conditions.*.

Specifically, to ensure that players do not leave quickly, we can temporarily raise the packet rate if the high risk of their premature departure is due to long propagation delays or a high loss rate on a noisy link, rather than transient congestion. Alternatively, we can increase the degree of data redundancy to cope with serious network impairment. Once the players have settled down and become immersed in the game play, they are *relatively less sensitive or reactive to network impairment*, i.e., they may remain in the game even if the network quality deteriorates. It could be that players are reluctant to quit because they have invested so much time in the current session; or they are simply more tolerant of network impairment as they are in the flow state (i.e., addicted). In either case, by allocating extra scarce resources to the more demanding players, we may increase the overall game playing time and user satisfaction. Our model could also be treated as a utility function to evaluate alternative design choices. For example, suppose a number of transport protocols have been designed for a particular game, and one protocol performs better in terms of network latency than loss recovery. In this case, we can predict the probability of premature departures for each candidate protocol, and pick the protocol that yields the lowest premature departure probability.

The following examples demonstrate how our model can be used to help make design decisions that will improve players' gaming experiences.

- One way to increase the responsiveness of game play is to reduce the processing time. As server processing power is finite, we can only sacrifice the responsiveness of low-risk sessions (i.e., those with better network conditions) to enhance that of high-risk sessions, so that messages from high-risk sessions will be given higher priority. We define sessions with the highest 25% risk scores as high-risk sessions, and those with the lowest 25% risk scores as low-risk sessions. For each "target" round-trip time t, we fairly defer the processing of low-risk sessions in order to make the round-trip time of high-risk sessions no longer than t. The simulation results for this configuration are shown in Fig. 13(a). We find that the predicted premature departure probability constantly decreases with a shorter target round-trip time. If the target round-trip time is set to 60 ms, the overall premature departure probability is expected to decrease by 10%. This indicates that sacrificing the responsiveness of low-risk sessions to help high-risk sessions in terms of responsiveness is feasible, as the former may still provide an acceptable gaming experience with degraded responsiveness.
- As delay jitters are less tolerable than absolute delays (see Section VI-C), we demonstrate how de-jitter buffers, which stall arrived packets in exchange for reduced delay jitters if necessary, reduce the probability of players' premature departure. The length of our de-jitter buffer is very similar to that of the RTO (TCP's Retransmission Timer) calculation $SRTT + k \times RTTVAR$ [30], where $SRTT$ and $RTTVAR$ denote the smoothed round-trip time and round-trip time variation respectively; we choose $k = 2$ for our setting. As shown in Fig. 13(b), the premature departure probability reaches a minimum when the proportion of de-jitter buffer users is around 0.3, where sessions adopt the de-jitter buffer based on the decreasing order of their delay jitters (high-jitter sessions are considered first). The reason is that, while the de-jitter buffer is beneficial for high-jitter sessions, it introduces unnecessary overhead for low-jitter sessions by incurring long delay times. The result shows that, if sessions with the highest 30% delay jitters are equipped with de-jitter buffers, the overall premature departure probability declines by 4%.

VII. Conclusion

To understand the relationship between network quality and players' departure patterns, we analyzed a $1,356$-million-packet trace from a commercial MMORPG called *ShenZhou Online*. Our results indicate that both network delay and network loss significantly affect a player's decision to leave a game prematurely, i.e., the player quits a few minutes after joining a game. We show that it is feasible to predict whether players will quit prematurely based on the network conditions they experience. The proposed model can determine the relative impact of different types of network impairment. For our traces, the degrees of player intolerance of network delay, delay jitter, client packet loss, and server packet loss are in the proportion of 1:2:4:3 approximately. The model is very useful for evaluating system design decisions. By using the model, we have shown that 1) the premature departure probability can be significantly decreased by *prioritizing server processing* according to sessions' risk scores; 2) *de-jitter buffers* can reduce the probability of premature departure; and 3) if we replace the commonly used protocol, TCP, with a more lightweight transport protocol to eliminate the additional delay jitters caused by TCP's in-order delivery, the premature departure probability can be significantly reduced.

Although many network researchers have focused on ways to measure users' opinions about network performance *objectively*, there is still no consensus on game players' sensitivity to and intolerance of network conditions. This may be because users' perceptions are inevitably connected to the nature of the game genre, playing skills, system design and implementation details, and particularly the choice of transport protocol. A player who is intolerant of 200 ms latency and a 1% loss rate for a game may have difficulty in playing another game under the same network configuration. The key to resolving the inconsistency is to *separate* the effects of network QoS, system design, transport protocols, and their interaction on game players. How to generalize the measure of users' QoS-sensitivity so that the measures of different applications can be normalized and compared with one another remains an open question and will be a major theme of our future work.

Acknowledgments

This work would not have been possible without the extensive traffic trace of *ShenZhou Online*. The authors are much indebted to the following people who helped us gather the trace: Tsing-San Cheng, Lawrence Ho, Chen-Hsi Li, and especially to Yen-Shuo Su, who between them made the datasets available. This work was supported in part by National Science Council of the Republic of China under the grants NSC 96-2628-E-001-027 and NSC 97-2221-E-001-009.

References

[1] Aetas, Inc., “Gametrics weekly Korea MMORPG population survey,” 2007. [Online]. Available: http://www.4gamer.net/specials/gametrics/gametrics.shtml

[2] S. Aggarwal, H. Banavar, A. Khandelwal, S. Mukherjee, and S. Rangarajan, “Accuracy in dead-reckoning based distributed multi-player games,” in *Proceedings of ACM SIGCOMM 2004 workshops on NetGames ’04*. ACM Press, 2004, pp. 161–165.

[3] G. Armitage, “An experimental estimation of latency sensitivity in multiplayer Quake 3,” in *11th IEEE International Conference on Networks (ICON)*, 2003.

[4] T. Beigbeder, R. Coughlan, C. Lusher, J. Plunkett, E. Agu, and M. Claypool, “The effects of loss and latency on user performance in Unreal Tournament 2003,” in *Proceedings of NetGames’04*. ACM Press, 2004, pp. 144–151.

[5] F. Chang and W. chang Feng, “Modeling player session times of on-line games,” in *NetGames ’03: Proceedings of the 2nd Workshop on Network and System Support for Games*. ACM Press, 2003, pp. 23–26.

[6] K.-T. Chen, P. Huang, and C.-L. Lei, “Game traffic analysis: An MMORPG perspective,” *Computer Networks*, vol. 50, no. 16, pp. 3002–3023, 2006.

[7] ——, “How sensitive are online gamers to network quality?” *Communications of the ACM*, vol. 49, no. 11, pp. 34–38, Nov 2006.

[8] K.-T. Chen, P. Huang, G.-S. Wang, C.-Y. Huang, and C.-L. Lei, “On the sensitivity of online game playing time to network QoS,” in *Proceedings of IEEE INFOCOM’06*, Barcelona, Spain, Apr. 2006.

[9] W. S. Cleveland, “LOWESS: a program for smoothing scatterplots by robust locally weighted regression,” *The American Statistician*, vol. 35, no. 54, 1981.

[10] D. R. Cox and E. J. Snell, *Analysis of Binary Data*, 2nd ed. London: Chapman and Hall, 1989.

[11] A. DeLeán, P. Munson, and D. Rodbard, “Simultaneous analysis of families of sigmoidal curves: application to bioassay, radioligand assay, and physiological dose-response curves,” *Am J Physiol Endocrinol Metab*, vol. 235, pp. 97–102, 1978.

[12] J. Egan, *Signal Detection Theory and ROC Analysis*. New York: Academic Press, 1975.

[13] W. C. Feng, F. Chang, W. C. Feng, and J. Walpole, “A traffic characterization of popular on-line games,” *IEEE/ACM Transactions on Networking*, vol. 13, no. 3, pp. 488–500, June 2005.

[14] R. A. Fisher, “On the interpretation of χ^2 from contingency tables, and the calculation of p,” *Journal of the Royal Statistical Society*, vol. 85, pp. 87–94, 1922.

[15] T. J. Hastie and R. J. Tibshirani, *Generalized Additive Models*. London: Chapman and Hall, 1990.

[16] T. Henderson, “Latency and user behaviour on a multiplayer game server,” in *Proceedings of the Third International COST Workshop (NGC 2001)*. Springer-Verlag, 2001, pp. 1–13.

[17] T. Henderson and S. Bhatti, “Modelling user behaviour in networked games,” in *MULTIMEDIA ’01: Proceedings of the Ninth ACM International Conference on Multimedia*. ACM Press, 2001, pp. 212–220.

[18] ——, “Networked games: a QoS-sensitive application for QoS-insensitive users?” in *RIPQoS’03: Proceedings of the ACM SIGCOMM Workshop on Revisiting IP QoS*. ACM Press, 2003, pp. 141–147.

[19] Hosmer, S. L. Cessie, and Lemeshow, “A comparison of goodness of fit tests for the logistic regression model, statistics in medicine,” *Statistics in Medicine*, vol. 16, pp. 965–980, 1997.

[20] D. W. J. Hosmer and S. J. Lemeshow, *Applied logistic regression*, 2nd ed. New York: Wiley, 2000.

[21] S. Ila, D. Mizerski, and D. Lam, “Comparing the effect of habit in the online game play of australian and indonesian gamers,” in *Proceedings of the Australia and New Zealand Marketing Association Conference*, 2003.

[22] F. E. H. Jr., R. M. Califf, D. B. Pryor, K. L. Lee, , and R. A. Rosati, “Evaluating the yield of medical tests,” *Journal of the American Medical Association*, vol. 247, no. 18, pp. 2543–2546, 1982.

[23] E. L. Kaplan and P. Meier, “Nonparametric estimation from incomplete observations,” *Journal of the American Statistical Association*, vol. 53, pp. 437–481, 1958.

[24] M. Kendall, “A new measure of rank correlation,” *Biometrika*, vol. 30, pp. 81–93, 1938.

[25] J. Nichols and M. Claypool, “The effects of latency on online madden NFL football,” in *Proceedings of NOSSDAV’04*. ACM Press, 2004, pp. 146–151.

[26] T. P. Novak, D. L. Hoffman, and A. Duhachek, “The influence of goal-directed and experiential activities on online flow experiences,” *Journal of Consumer Psychology*, vol. 13, no. 1, pp. 3–16, 2003.

[27] M. Oliveira and T. Henderson, “What online gamers really think of the internet?” in *Proceedings of NetGames’03*. ACM Press, 2003, pp. 185–193.

[28] L. Pantel and L. C. Wolf, “On the impact of delay on real-time multiplayer games,” in *Proceedings of NOSSDAV’02*. ACM Press, 2002, pp. 23–29.

[29] ——, “On the suitability of dead reckoning schemes for games,” in *NETGAMES ’02: Proceedings of the 1st workshop on Network and system support for games*. ACM Press, 2002, pp. 79–84.

[30] V. Paxson and M. Allman, “Computing TCP’s Retransmission Timer,” RFC 2988 (Proposed Standard), Nov. 2000. [Online]. Available: http://www.ietf.org/rfc/rfc2988.txt

[31] V. Paxson and S. Floyd, “Wide area traffic: the failure of Poisson modeling,” *IEEE/ACM Transactions on Networking*, vol. 3, no. 3, pp. 226–244, 1995.

[32] P. Quax, P. Monsieurs, W. Lamotte, D. D. Vleeschauwer,

and N. Degrande, “Objective and subjective evaluation of the influence of small amounts of delay and jitter on a recent first person shooter game,” in *Proceedings of ACM SIGCOMM 2004 workshops on NetGames ’04*. ACM Press, 2004, pp. 152–156.

[33] N. Sheldon, E. Girard, S. Borg, M. Claypool, and E. Agu, “The effect of latency on user performance in Warcraft III,” in *Proceedings of NetGames’03*. ACM Press, 2003, pp. 3–14.

[34] J. Smed, T. Kaukoranta, and H. Hakonen, “A review on networking and multiplayer computer games,” Turku Centre for Computer Science, Tech. Rep. 454, Apr. 2002.

[35] UserJoy Technology Co., Ltd., “ShenZhou Online,” 2004. [Online]. Available: http://www.ewsoft.com.tw/

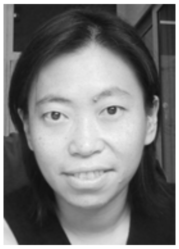
[36] B. S. Woodcock, “An analysis of MMOG subscription growth – version 23.0,” 2008. [Online]. Available: http://www.mmogchart.com/

[37] T. Yasui, Y. Ishibashi, and T. Ikedo, “Influences of network latency and packet loss on consistency in networked racing games,” in *Proceedings of NetGames’05*. ACM Press, 2005, pp. 1–8.

[38] N. Yee, “Motivations for play in online games,” *CyberPsychology & Behavior*, vol. 9, no. 6, pp. 772–775, Dec 2006.

[39] S. Zander and G. Armitage, “Empirically measuring the QoS sensitivity of interactive online game players,” in *Australian Telecommunications Networks and Applications Conference (ATNAC)*, 2004.

Kuan-Ta Chen received his B.S. and M.S. in Computer Science from Na- tional Tsing-Hua University in 1998 and 2000, respectively. He received his Ph.D. in Electrical Engineering from National Taiwan University in 2006. He then joined the Institute of Information Science, Academia Sinica as an assistant research fellow. His research interests include Internet measurement, Internet QoS, network security, and online gaming. Much of his recent work focus on the human factors in network systems, especially user satisfaction measurement, user behavior modeling, and user-perception-based system design. He is a member of ACM and IEEE.



Polly Huang received her Ph.D. (1999) and M.S. (1997) in Computer Science from University of Southern California, and her B.S. (1993) in Mathematics from National Taiwan University.

In 2003, she joined the Department of Electrical Engineering of the National Taiwan University at which she currently holds an associate professor position. Prior to joining NTU, she worked as a post-doctoral research scientist at the Computer Engineering and Networks Laboratory (TIK) of the Swiss Federal Institute of Technology (ETH) Zurich and as a post-doctoral fellow at the Institute of Pure and Applied Mathematics of UCLA.

Dr. Huang’s research interest includes sensor networking, multimedia networking, and Internet characterization. She is a member of ACM and IEEE and serves currently on the editorial board of the Journal of Communications and Networks.

Chin-Laung Lei received his B.S. degree in Electrical Engineering from National Taiwan University in 1980, and his Ph.D. degree in Computer Science from the University of Texas at Austin in 1986. From 1986 to 1988, he was an assistant professor in the Computer and Information Science Department at the Ohio State University, Columbus, Ohio, U.S.A. In 1988 he joined the faculty of the Department of Electrical Engineering, National Taiwan University, where he is now a professor. His current research interests include computer and network security, cryptography, parallel and distributed processing, design and analysis of algorithms, and operating system design. Dr. Lei has published more than 150 technical articles in scientific journals and conference proceedings, and he is a co-winner of the first IEEE LICS test-of-time award. He is currently the vice president and a life member of the Chinese Cryptology and Information Security Association. He is also a member of ACM, IEEE, and IACR.