A Generalizable Methodology for Quantifying User Satisfaction*

Te-Yuan HUANG[†a)], **Kuan-Ta CHEN**[††b)], **Polly HUANG**[†††,††††c)],
and **Chin-Laung LEI**[†††,†††††d)], *Nonmembers*

SUMMARY   Quantifying user satisfaction is essential, because the results can help service providers deliver better services. In this work, we propose a generalizable methodology, based on *survival analysis*, to quantify user satisfaction in terms of session times, i.e., the length of time users stay with an application. Unlike subjective human surveys, our methodology is based solely on passive measurement, which is more cost-efficient and better able to capture *subconscious reactions*. Furthermore, by using session times, rather than a specific performance indicator, such as the level of distortion of voice signals, the effects of other factors like loudness and sidetone, can also be captured by the developed models. Like survival analysis, our methodology is characterized by low complexity and a simple model-developing process. The feasibility of our methodology is demonstrated through case studies of *ShenZhou Online*, a commercial MMORPG in Taiwan, and the most prevalent VoIP application in the world, namely Skype. Through the model development process, we can also identify the most significant performance factors and their impacts on user satisfaction and discuss how they can be exploited to *improve user experience and optimize resource allocation*.
key words: *Human Perception, Internet Measurement, Quality of Service, Survival Analysis, Network Games, VoIP*

1. INTRODUCTION

User satisfaction is a key measure of an application's success; hence, service providers need accurate measurement results to help them improve their systems and technology. However, it is very difficult to quantify the degree of user satisfaction efficiently, especially for interactive real-time applications. Subjective surveys are expensive and can only capture conscious reactions; while most objective methods evaluate user

[†]The author is with the Department of Electrical Engineering, National Taiwan University, Taipei, Taiwan.
[††]The author is with the Institute of Information Science, Academia Sinica, Taipei, Taiwan.
[†††]The author is with the Faculty of Department of Electrical Engineering, National Taiwan University, Taipei, Taiwan.
[††††]The author is with the Faculty of Graduate Institute of Networking and Multimedia, National Taiwan University, Taipei, Taiwan.
  a) E-mail: r95921037@ntu.edu.tw
  b) E-mail: ktchen@iis.sinica.edu.tw
  c) E-mail: phuang@cc.ee.ntu.edu.tw
  d) E-mail: lei@cc.ee.ntu.edu.tw

satisfaction in terms of specific performance indicators, such as game scores or voice signals, which are limited and not portable. In this work, we propose a generalizable methodology, based on statistical analysis, to investigate the real reasons that users decide to stay with or leave an application. We then quantify user satisfaction according to the duration of a user's stay with the application, i.e., the *session time*. Although the session time cannot represent user satisfaction directly, we believe it is a good and intuitive indicator of users' perceptions of system performance. Besides, in our case studies, which are excerpted from our previous works [1], [2], the observed correlation between session times and performance factors, such as network QoS, strongly supports our assumption that *premature departures from an application are partially caused by unfavorable experiences.*

We base our methodology on *survival analysis* because of its ability to handle censored data, i.e., incomplete observations. Together with the Cox Proportional Hazards model [3], we are able to identify the most significant performance factors and quantify their impact on user satisfaction. The feasibility of our methodology is demonstrated by our case studies. By modeling user satisfaction in *ShenZhou Online*, a commercial MMORPG in Taiwan, and Skype, the most popular VoIP application in the world, we found that system performance factors, such as network QoS, significantly affect users' willingness to continue with an application. In *ShenZhou Online*, for example, both network delay and network loss have a significant impact on user satisfaction; while in Skype, the source rate and its jitter are the key factors. The proposed model not only quantifies user satisfaction, but also *provides useful hints about optimizing system performance and resource allocation.*

Our contribution in this work is fourfold. 1) Our methodology provides an *objective* method for quantifying user satisfaction, and it is generalizable to various applications. 2) Since the method is based solely on passive measurements, rather than subjective surveys, *subconscious reactions* can also be captured. 3) Unlike other objective methods, our method is based on session times, rather than a specific performance indicator, such as game scores or voice signals; factors other than the designated indicator can also be captured. 4)

Because it is based on survival analysis, our methodology is characterized by low complexity and a simple model-development process.

The remainder of this paper is organized as follows. We first summarize related works in Section 2, and then describe our methodology in Section 3. In Section 4, we provide two case studies that quantify user satisfaction in *ShenZhou Online* and Skype respectively. Section 5 considers other applications using the derived model. We then present our conclusions in Section 6.

2. RELATED WORK

The numerous methods proposed to gain a better understanding of user satisfaction can be divided into two categories: subjective and objective. In subjective evaluations, the assessment of user satisfaction is based on subjective human surveys. On the other hand, by definition, objective methods are based on impartial measurements.

2.1 Subjective Evaluation

Subjective evaluation is the most straightforward way to establish the relationship between user satisfaction and system performance factors, such as game design or network QoS. Quality of service is evaluated by groups of naive subjects under controlled environments. Combinations of various features, such as different user interfaces, designs, or color collocations, are applied in the environment, and subjects are asked to grade each setting based on the perceived quality. Although this is the most direct way to understand the impact of system performance factors on user satisfaction, it is very expensive to conduct such surveys. For example, as mentioned in ITU-T Recommendation P.800 [4], which describes methods and procedures for conducting subjective evaluations of transmission quality, sound-proof cabinets must be prepared and the reverberation time in the cabinets must be controlled within a certain range. Moreover, gifts or cash need to be given to elicit information from survey participants.

Another drawback with subjective methods is that they are *intrusive*, since subjects have to make explicit judgements about service quality using predefined scales, which do not reflect the real situation when using the service. Therefore, subjective methods can only capture conscious perceptions; however, subconscious reactions may be just as important, but they are not considered. The results may also suffer from errors of attrition, caused by the respondents becoming increasingly careless as they fill in a long questionnaire. Responses may also be affected by recency effect, the user's expectations, his/her experience relative to the service, as well as emotional and cultural factors. Since every type of reactive human behavior should be critically analyzed, all of the above factors should be taken

into consideration. Consequently, the subjective quality assessment model and the interpretation of the results are always in danger of either oversimplifying the perception process in establishing a user satisfaction model, or getting lost in individual behavioral characteristics and thereby over complicating the model [5].

2.2 Objective Evaluation

Most objective evaluation methods assess user satisfaction in terms of a specific performance indicator, such as the number of kills in a shooting game or the level of distortion in a received voice signal. Thus, the results can only reflect the impact of the designated performance factor and do not provide a comprehensive evaluation of the system's performance quality. For example, PESQ [6] is an objective method for evaluating end-to-end speech quality, but it only measures the effects of one-way speech distortion and noise on speech quality. The effects of loudness, loss, delay, sidetone, echo, and other impairments related to two-way interaction (e.g., centre clipper) are not reflected in the PESQ scores [7]. Meanwhile, some objective indicators based on user performance, such as game scores. However, user performance is highly dependent on user skills; thus, the results are not comparable and generalizable across different applications. Furthermore, some objective metrics, such as the received voice signal and its original voice signal, are difficult to retrieve under the prevalent peer-to-peer network structure.

The proposed objective evaluation method is based solely on passive measurements; thus, *subconscious reactions* that users are even unaware of can also be captured. Furthermore, unlike other objective methods, our approach assesses system performance based on session times, i.e., how long a user stays with an application, rather than on a specific performance indicator. Hence, our method can capture the effects of other factors, such as the content of a service and UI design, on user satisfaction in terms of session times.

3. METHODOLOGY

We now present our methodology for analyzing the relationship between session times and performance factors from the perspective of survival analysis. Instead of using straightforward multiple regression techniques, we adopt this statistical methodology for three reasons. First, and foremost, survival analysis can handle *censored data*, i.e., incomplete observations. Second, the session times, i.e., the length of time a user stays with an application, usually follow an exponential or long-tailed distribution, not a normal distribution assumed in ordinary multiple regression. Third, the relationship between session times and performance factors can be properly assessed by transforming it into a multiple regression problem, which corresponds to the well known
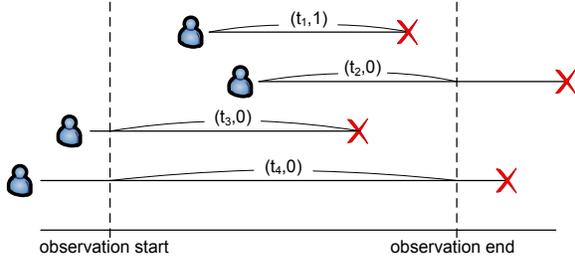
Fig. 1 Measurement setups often lead to explicit censoring of sessions. The four possible censoring scenarios are denoted by (t, s), where t is the observed duration and s is the censoring status.

Cox Proportional Hazards model [3] in survival analysis.

3.1 Survival Analysis

Survival analysis is often used in the medical field to describe the relationship between patients' survival time and the treatment they receive; and *survival time* often refers to the development of a particular symptom. We found that this approach also provides a good fit for describing the reactions of users to a system's performance.

Following the conventions of survival analysis, we denote a user's departure as an *event* or a *failure*. However, at the end of the observation period, some subjects will not have left the system. On the other hand, some subjects may start their session before the beginning of the observation period. Thus, the survival times of those subjects are not completely observed. Such survival times are termed *censored*, to indicate that the session is not completely observed. Censored observations should be also used because longer sessions are more likely to be censored than shorter sessions. Simply disregarding them would lead to underestimation. An indicator variable, s_i, which is the censoring status, is used to indicate whether a session, i, has been censored: thus $s_i = 1$ means an event has occurred (the observation is not censored) and $s_i = 0$ represents a censored observation, as illustrated in Fig. 1.

A *survival function* is commonly used to describe the lifetime pattern of an object or a set of observations. In our context, the survival function is defined as follows:

$$S(t) = \Pr(\text{a session that survives longer than time } t)$$
$$= 1 - \Pr(\text{a session that fails before,}$$
$$\text{or is equal to, time } t)$$
$$= 1 - F(t),$$

where $F(t)$ is the cumulative distribution function (CDF) of session times. A standard estimator of the survival function, proposed by Kaplan and Meier [8], is called the Product-Limit estimator or the Kaplan-Meier estimator. Suppose there are n distinct ses-

sion times $t_1, t_2, \ldots, t_n$ in ascending order such that $t_1 < t_2 < \ldots < t_n$, and that at time t_i there are d_i events and Y_i active sessions. The estimator is then defined as follows for all values of $t \leq t_n$:

$$\hat{S}(t) = \prod_{t_i \leq t} \Pr[T > t_i | T \geq t_i]$$
$$= \begin{cases} 1 & \text{if } t < t_1, \\ \prod_{t_i \leq t} [1 - \frac{d_i}{Y_i}] & \text{if } t_1 \leq t. \end{cases}$$

Note that, for time intervals contain censored data, the probability of surviving is one, since the exact surviving time is longer than the observed one. Thus, if the session i is censored, i.e., s_i is zero; d_i is zero. On the other hand, d_i is one when s_i is one, i.e., the session i is not censored. For observations with ties, if the session times are continuous in essence and later discretized by measurement, a practical solution is to add a small amount of "noise" so that all times are unique. After estimating the survival function, the pth quantile of the lifetime, t_p, can be obtained by

$$t_p = \inf\{t : \hat{S}(t) \leq 1 - p\}. \tag{1}$$

This equation can be used repeatedly to estimate the median session time as $t_{0.5}$ for a group of sessions.

In addition to the survival function, *hazard function*, or *hazard rate*, is a frequently used quantity in survival analysis. It is also known as the conditional failure rate in reliability engineering, or the intensity function in stochastic processes. The hazard rate is defined by

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

A related quantity is the cumulative hazard function $H(t)$, which is defined by

$$H(t) = \int_0^t h(u)du = -\ln[S(t)].$$

The hazard function gives the *instantaneous rate* at which failures occur for observations that have survived at time t. The quantity $h(t)\Delta t$ may 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. The hazard function plays an important role in the Cox regression model because the hazard rate of session times $h(t)$ is taken as the response variable of network QoS factors. We discuss this point in the next section.

3.2 Regression Modeling

By combining survival analysis with correlation analysis, we may be able to derive the relationship between a specific QoS factor and session times. However, the true impact of individual factors remains hidden because of *the collinearity of factors*; that is ,two or more

factors correlate significantly with session times. As a result, it is not clear which factor has the greatest effect on user satisfaction. Users may be particularly affected by one factor, or they may be sensitive to all of them.

To distinguish the impact of individual factors, we use regression analysis to model session times as the response to QoS factors.

3.2.1 The Cox Regression Model

The Cox proportional hazards model [3] has long been the most widely used method for modeling the relationship between factors and *censored* outcomes. The model treats performance factors, such as the bit rate, as risk factors or covariates; in other words, as variables that can cause failures. The hazard function of each session is *decided completely by a baseline hazard function and the risk factors related to that session.* We define the risk factors of a session as a risk vector $\mathbf{Z}$. The regression equation is as follows:

$$h(t|\mathbf{Z}) = h_0(t)\exp(\boldsymbol{\beta}^t\mathbf{Z}) = h_0(t)\exp(\sum_{k=1}^{p}\beta_k Z_k), (2)$$

where $h(t|\mathbf{Z})$ is the hazard rate at time t for a session with risk vector $\mathbf{Z}$; $h_0(t)$ is the baseline hazard function computed during the regression process; and $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_p)^t$ is the coefficient vector that corresponds to the impact of risk factors. Dividing both sides of Equation 2 by $h_0(t)$ and taking the logarithm, we obtain

$$\log\frac{h(t|\mathbf{Z})}{h_0(t)} = \beta_1 Z_1 + \cdots + \beta_k Z_k = \sum_{k=1}^{p}\beta_k Z_k = \boldsymbol{\beta}^t\mathbf{Z}, (3)$$

where Z_p is the pth factor of the session. The right side of Equation 3 is a linear combination of covariates with weights set to the respective regression coefficients, i.e., it is transformed into a *linear regression equation.*

Under the Cox model, if we look at two sessions with risk vectors $\mathbf{Z}$ and $\mathbf{Z}'$, the hazard ratio (the ratio of their hazard rates) is

$$\frac{h(t|\mathbf{Z})}{h(t|\mathbf{Z}')} = \frac{h_0(t)\exp[\sum_{k=1}^{p}\beta_k Z_k]}{h_0(t)\exp[\sum_{k=1}^{p}\beta_k Z_k']}$$

$$= \exp[\sum_{k=1}^{p}\beta_k(Z_k - Z_k')], \qquad (4)$$

which is a time-independent constant, i.e., the hazard ratio of the two sessions is independent of time. For this reason the Cox model is often called the *proportional hazards model.* Note that Equation 4 imposes very strict conditions for applying the Cox model, because the model's validity relies on the assumption that the *hazard rates for any two sessions must be in proportion all the time.*

3.2.2 Proportional Hazards Check and Adjustment

Because of the above restriction, we should begin the model development by checking whether the data set satisfies the proportional hazards assumption. We first present our methodology for checking categorical variables, and then consider continuous factors.

It is not unusual to observe temporal dependence, such as the day of the week effect, in network traffic traces. Thus, categorical variables are often used to categorize the data set. The check could be performed by first grouping sessions by the categorical variables and then plotting the cumulative hazard function $H_i(t)$ versus t for each group i in a log-log scale. If the proportional hazards assumption is satisfied, the log survival curves should steadily drift apart. If non-proportional categorical variables are present, the *stratified Cox model* can accommodate them. The model augments the basic Cox model by incorporating the support of strata, where each stratum has its own baseline hazard function. For a Cox model with m strata, Equation 3 is generalized to

$$h_i(t|\mathbf{Z}) = h_{0i}(t)\exp(\boldsymbol{\beta}^t\mathbf{Z}), i = 1, \ldots, m.$$

Note that, although the baseline hazard function for each stratum can be different, the coefficient vector $\boldsymbol{\beta}$ is shared by all strata.

For continuous variables, the Cox model assumes a linear relationship between the covariates and the hazard function, i.e., it implies that the ratio of risks between a 20ms- and a 30ms-average RTT session is the same as that between a 90ms- and a 100ms-average RTT session. Thus, to proceed with the Cox model, we must ensure that our predictors have a linear influence on the hazard functions. We investigate the impact of the covariates on the hazard functions with the following equation:

$$E[s_i] = \exp(\boldsymbol{\beta}^t f(\mathbf{Z})) \int_0^\infty I(t_i \geqslant s)h_0(s)ds, \qquad (5)$$

where s_i is the censoring status of session i, and $f(z)$ is the estimated functional form of the covariate z. This corresponds to a Poisson regression model if $h_0(s)$ is known, where the value of $h_0(s)$ can be approximated by simply fitting the Cox model with unadjusted covariates. We can then fit the Poisson model with smoothing spline terms for each covariate [9]. If the covariate has a linear impact on the hazard functions, the smoothed terms will approximate a straight line. A solution for non-proportional variables is *scale transformation*, which checks if the failure rate is proportional to the *scale of the variable*, rather than its magnitude.

Despite the non-strict-linearity of our covariates, we should further examine whether the proportional hazard assumption holds. One way is to fit the same

data to a more generalized Cox model that allows time-dependent coefficients [9]. In this model, Equation 3 is extended to

$$\log \frac{h(t|\mathbf{Z})}{h_0(t)} = \sum_{k=1}^{p} \beta(t)_k Z_k = \sum_{k=1}^{p} (\beta_k + \theta_k \ln(t)) Z_k,$$

where the coefficient vector $\boldsymbol{\beta}(t)$ is time-dependent. The null hypothesis, which indicates the conformance of the proportional hazards assumption, corresponds to $\theta_k \equiv 0, k = 1, \ldots, p$. In this case, $\boldsymbol{\beta}(t)$ in the extended model reduces to $\boldsymbol{\beta}$ in the standard model. The test is similar to a standard linear trend test in that it determines whether a significant non-zero slope exists by an ordinary least square regression test.

3.2.3   Model Validation

To assess the overall goodness-of-fit of our model, we can use the Cox and Snell residuals [10]. If the model is correctly fitted, the random variable $r_i = \hat{H}(t_i, \mathbf{Z_i})$ will have an exponential distribution with a hazard rate of 1, where $\hat{H}(t_i, \mathbf{Z_i})$ is the estimated cumulative hazard rate for session i with risk vector $\mathbf{Z_i}$. Accordingly, the plot of r_i and its Kaplan-Meier estimate of the survival function $\hat{S}(r)$ will be a straight line through the origin with a slope of 1. We can further validate our model by *prediction*; that is, given a performance vector $\mathbf{Z}$, we can predict the most probable session time as the median time of the estimated survival curve, i.e., $\inf\{t : S(t|\mathbf{Z}) \le 0.5\}$; while $S(t|\mathbf{Z}) = exp(-H(t|\mathbf{Z}))$ is the computed survival function for the session with risk vector $\mathbf{Z}$. By the relation, one can sort and group all sessions by their risk scores, $\boldsymbol{\beta}^t \mathbf{Z}$, and predict session times based on the median risk score in each group. Then, the model can be validated by comparing the predicted session times with the actual median times to determine if they are within a certain predicted confidence band.

4.   CASE STUDY

In this section, we consider two case studies to demonstrate the complete procedure for quantifying user satisfaction. We first investigate online gaming, since it is one of the most profitable businesses on the Internet. Then, we explore user satisfaction on VoIP, which is one of the most popular services on the Internet. By determining user satisfaction, service providers can deliver better service quality to users and thereby boost service subscriptions.
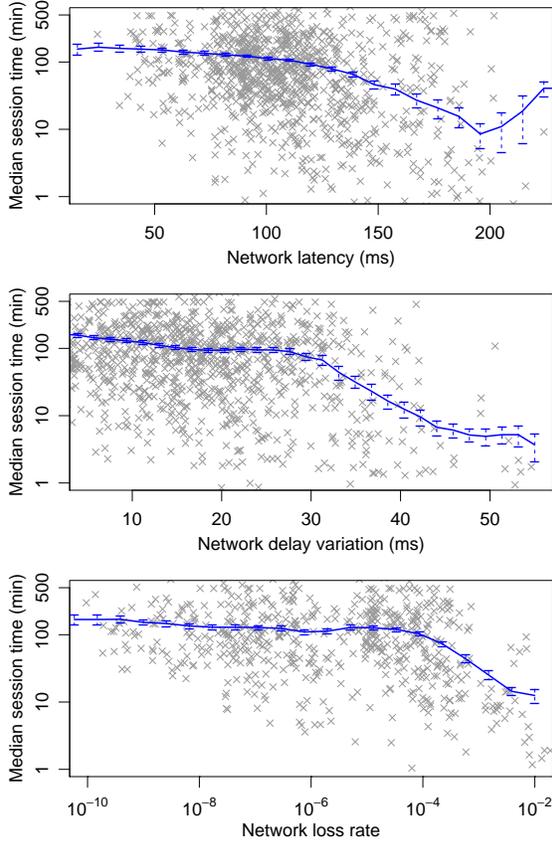
4.1   Online Game

The popularity of online gaming has increased rapidly in recent years; however, users still experience unfavorable network conditions. Numerous complaints about long or frequent lags are made in game-player forums. Thus, to understand online gamers' QoS-sensitivity, we investigate the relationship between gamers' playing times and network QoS factors.

We collected traces from *ShenZhou Online* [11], a commercial massively multiplayer online role-playing game (MMORPG). To play *ShenZhou Online*, thousands of players pay a monthly subscription fee at a convenience store or online. As a typical MMORPG, players can engage in fights with random creatures, train themselves to acquire particular skills, partake in commerce or take on a quest. Compared to other game genres, such as FPS (First-Person Shooting) games, MMORPGs are relatively slow-paced and have less-stringent service requirements. Therefore, they could be viewed as a baseline for real-time interactive online games. In other words, if network QoS frustrates MMORPG players, it should also frustrate players of other game genres. With the help of the *ShenZhou Online* staff, we recorded all inbound/outbound game traffic of the game servers located in Taipei; a total of 15,140 game sessions over two days. The observed players were spread over 13 countries, including China, India, Hong Kong, and Malaysia, and hundreds of autonomous systems, thus manifesting the heterogeneity of network-path characteristics and the generality of the trace.

4.1.1   Performance Factor Identification

According to the theory proposed in [12], when users are playing a game, if the feeling of involvement in the virtual world is diminished by network lags, they become more consicous of the real world, which reduces their sense of time distortion. Therefore, we expect players' staying times in MMORPGs will be affected, to some extent, by the network's QoS. From the collected trace, we extracted the network performance for each session based on the sequence number and flags in the TCP packet header. On average, players stayed for 100 minutes after joining a game. However, the difference in individual game-playing times was quite large; for example, the shortest 20% of sessions spanned less than 40 minutes, but the top 20% of players spent more than eight hours continuously in the game. Fig. 2 illustrates the difference in the game-playing times of sessions experiencing different levels of network quality. The figure depicts the association of game playing times with network latency, network delay variation, i.e., the standard deviation of network latency, and the network loss rate, respectively. All three plots indicate that the more serious the network impairment experienced by players, the earlier they were likely to leave the game. The changes in game-playing time are significant. For instance, gamers who experienced 150 msec latency played four hours on average, but those experiencing 250 msec latency played for only one hour an

Fig. 2 Relationship between game session times and network QoS factors

Fig. 3 Actual vs. model-predicted game-playing time for session groups sorted by their risk scores.

average, a ratio of 4:1. Moreover, variations in network delay and network loss induce more variation in game playing times.
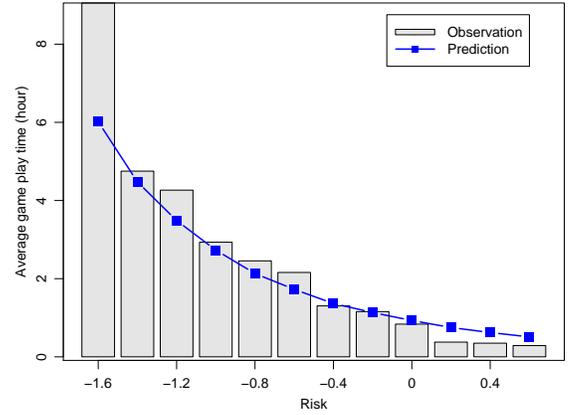
4.1.2 Impact of Individual Factors

Having demonstrated that players are not only sensitive to network quality, but also reactive to it, we assess how each individual QoS factor *influences* players' willingness to continue with a game or leave it.

Based on the methodology mentioned in Section 3.2, we propose a model that describes the changes in game-playing time due to network QoS. The model grades the overall quality of game playing based on specific network performance metrics, such as latency and loss, in terms of user satisfaction. The derived model takes network QoS factors as the input and computes the departure rate of online players as the output. The regression equation is derived as follows:

$$
\begin{aligned}
departure\ rate \propto \exp(&1.6 \times network\ latency \\
&+ 9.2 \times network\ delay\ variation \\
&+ 0.2 \times \log(network\ loss\ rate)).
\end{aligned} \quad (6)
$$

Note that, the logarithm of network loss rate in the

model is the result of *scale transformation*, mentioned in Section 3.2.2.

This illustrates that the player departure rate is roughly proportional to the exponent of the weighted sum of certain network performance metrics, where the weights reflect the effect of each type of network impairment. The coefficient can be interpreted by the ratio of the departure rates of two sessions. For example, suppose two players – A and B – join a game at the same time and experience similar levels of network loss and delay variations, except that their network latency is 100 msec and 200 msec, respectively. The ratio of their respective departure rates can then be computed by $\exp(1.6 \times (0.2 - 0.1)) \approx 1.2$, where 1.6 is the coefficient of network latency. That is, at every moment during the game, the probability that A will leave the game is 1.2 times greater than that of B.

Given the strong relationship between game-playing times and network QoS factors, we can "predict" the former if we know the later. Forecasting when a player will leave a game could provide useful hints for optimizing system performance and resource allocation. To validate our model, we compared the actual time and the model-predicted time for players in *Shen-Zhou Online*. In Fig. 3, we sort and group sessions by their risks scores, $\boldsymbol{\beta}^t \mathbf{Z}$, and predict game play time by the method described in Section 3.2.3. Then, the observed average game play times and the predicted times of each group are plotted in the figure. Note that, the departure rate in Equation 6 is proportional to the risk score, $\boldsymbol{\beta}^t \mathbf{Z}$. From the figure, we observe that, at the macro level, the prediction is rather close to the actual time, suggesting that a service provider can predict how long a given individual player will stay in a game and optimize resource allocation accordingly.

Note that, while this methodology can be applied to all kinds of online games, the exact equation for players' QoS sensitivity may depend on individual game design characteristics, such as the transport protocol and

client-prediction techniques used.

4.1.3 Findings and Discussion

The above results highlight the fact that *network delay variations are less tolerable than absolute delay.* Therefore, while current network games rely primarily on a "ping time" to select a server for a smooth game playing, delay jitters should also be considered in the server selection process. We also find that *players are more sensitive to network loss rates than network latency.* However, a study on *Unreal Tournament 2003* [13] reported that a typical network loss rate ($< 6\%$) has *no impact* on user performance. We believe the difference is caused by the choice of underlying transport protocol. That is, while most FPS games transmit messages via UDP, many MMORPGs, including *ShenZhou Online*, use TCP. Since TCP provides in-order delivery and congestion control, a lost packet will cause the subsequent packets to be buffered until it is delivered successfully, thereby reducing TCP's congestion window. In contrast, packet loss incurs no overhead in UDP. In short, for TCP-based online games, packet loss incurs *additional packet delay and delay jitters*, and therefore causes further annoyance to players. For this reason, and because of TCP's high communication overhead [14], we consider that more lightweight protocols would be more appropriate for real-time interactive network games.

4.2 VoIP

Among the various VoIP services, Skype is by far the most successful. There are over 200 million Skype downloads and approximately 85 million users worldwide. However, fundamental questions, such as whether VoIP services like Skype are good enough in terms of user satisfaction, have not been formally addressed. In this subsection, we quantify Skype user satisfaction based on the call duration measured from actual Skype traces, and propose an objective and perceptual index called the User Satisfaction Index (USI).

To collect Skype traffic traces, we set up a packet sniffer to monitor all traffic entering and leaving a campus network. In addition, to capture more Skype traces, a powerful Linux machine was set up to elicit more relay traffic passing through it during the course of the trace collection. However, given the huge amount of monitored traffic and the low proportion of Skype traffic, we used two-phase filtering to identify Skype VoIP sessions. In the first stage, we filtered and stored possible Skype traffic on a disk. Then, in the second stage, we applied an off-line identification algorithm to the captured packet traces to extract actual Skype sessions. Since we could not deduce round-trip times (RTT) and their jitter simply from packet traces, we sent out probe packets for each active flow while cap-
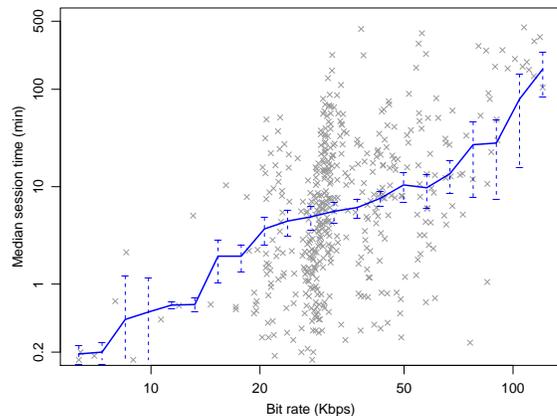

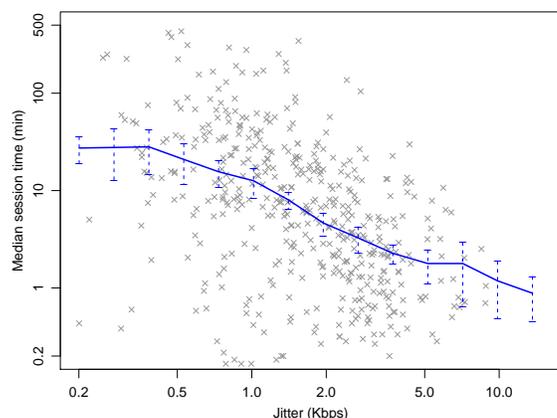**Fig. 4**   Correlation of bit rate with session time

Fig. 5   Correlation of jitter with session time

turing Skype traffic. The trace was collected over two months in late 2005. We obtained 634 VoIP sessions, of which 462 sessions were usable because they had more than five RTT samples. Among the 462 sessions, 253 were directly-established and 209 were relayed.

4.2.1 Performance Factor Identification

Skype uses a wideband codec that adapts to the network environment by adjusting the bandwidth used. Thus, when we explore the relationship between call duration and network conditions, we must also consider the source rate, along with network delay and loss. However, we do not have exact information about the source rate of remote Skype hosts. Thus, we use the received data rate as an approximation of the source rate. For brevity, we use the *bit rate* to denote the received data rate. We illustrate the correlation of the *bit rate* and call duration in Fig. 4, where the median time and their standard errors are plotted. The effect of the *bit rate* is clear, as we find that users tend to have longer conversations when the *bit rate* is higher. In fact, the median duration of the top 40% of calls is
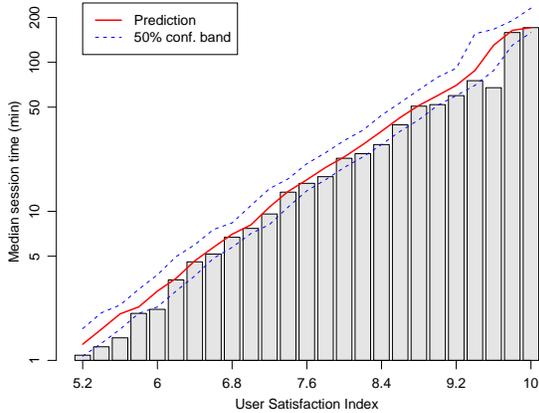
Fig. 6   Predicted vs. actual median duration of session groups sorted by their User Satisfaction Indexes.

ten times longer than the shortest 15%.

We also consider the jitter and round-trip time (RTT) variables, where jitter is the standard deviation of the bit rate sampled every second. It can capture the level of network delay variations and packet loss. We observe that when network impairment is more serious, users are more likely to terminate a call. For instance, as shown in Fig. 5, users who experienced jitter of less than 1 Kbps would make a call for 21 minutes in median; while users who experienced jitter of more than 2 Kbps would only talked for 3 minutes, which gives a high ratio of 7:1.

4.2.2   Impact of Individual Factors

To understand the impact of individual factors, we use regression analysis to model call duration as a response to QoS factors. Although we could simply put all potential QoS factors into the regression model, the result would be ambiguous if the predictors were strongly interrelated [15]. In [2], we analyze the level of correlation between QoS factors and classify them into three collinear groups. Then, we pick the *bit rate*, jitter, and RTT from each group and incorporate into the model, since they are the most significant predictors compared with their interrelated variables. For simplicity and parsimoniousness of the model, we omit the interaction terms of these three factors, although correlations between them have been observed. The developed User Satisfaction Index (USI) model is then used to evaluate the satisfaction levels of Skype users. As mentioned in Section 3.2.3, the risk score $\boldsymbol{\beta}^t\mathbf{Z}$ is used to represent the levels of instantaneous hang up probability, as it can be taken as a measure of user intolerance. Accordingly, we define the User Satisfaction Index of a session as its minus risk score:

$$
\begin{aligned}
USI &= -\boldsymbol{\beta}^t\mathbf{Z} \\
&= 2.15 \times \log(\text{bit rate}) - 1.55 \times \log(\text{jitter}) \\
&\quad - 0.36 \times \text{RTT}.
\end{aligned}
$$

We can further verify the proposed model by comparing the predicted call duration based on the proposed USI with the actual call duration. In Fig. 6, we group sessions by their USI, and plot the actual median duration, predicted duration, and 50% confidence bands of the latter for each group. The results show that the predicted duration is rather close to the actual median time; moreover, for most groups the actual median time is within the 50% of the predicted confidence band.

Although not shown, we use a set of independent metrics derived from patterns of user interactivity to validate USI. A strong correlation between the call duration and user interactivity suggests that *our model based on call duration is significantly representative of Skype user satisfaction.*

4.2.3   Findings and Discussion

By deriving the objective perceptual index, we can quantify *the relative impact* of the bit rate, the compound of delay jitter and packet loss, and network latency on the duration of Skype calls. Also, in [2], we have derived the importance of these three factors is approximately 46:53:1 respectively. The delay jitter and loss rate are known to be critical to the perception of real-time applications. To our surprise, the above results show that network latency has relatively little effect; however, the source rate is almost as critical as jitter, which is the compound of the delay jitter and packet loss. We believe these discoveries indicate that *adaptations for a stable, higher bandwidth channel would probably be the most effective way to increase user satisfaction with Skype.* The selection of relay nodes based on network delay optimization, a technique often used to find a quality detour by peer-to-peer overlay multimedia applications, is less likely to make a significant difference to Skype in terms of user satisfaction.

5.   APPLICATION

By understanding the most significant performance factors and their impacts on user satisfaction, we can further *improve user experience* and *optimize resource allocation.*

Given the quantified risk score of users leaving an application due to unsatisfactory service, systems can be modified accordingly. For example, for network applications, systems can be designed to automatically adapt to network quality in real time in order to improve user satisfaction. On the other hand, we might enhance the smoothness of usage in high-risk sessions

by increasing the packet rate or the degree of data redundancy; thus, users would have better experiences and be less likely to leave an application prematurely. Resource allocation could be deliberately biased toward high-risk sessions. For example, scarce resources, such as processing power or network bandwidth, could be allocated more effectively based on session risk scores.

The developed model could also provide useful hints to *resolve design trade-offs.* For instance, as the results in Section 4.1 indicate, players in *ShenZhou Online* are less tolerant of large delay variations than high latency. Thus, providing a smoothing buffer at the client side, though incurring additional delay, would improve overall user experience. Also, the concept of session time can be used to design an alarm system for abnormal system conditions. As we know, to provide continuous high-quality services, providers must monitor system performance around the clock and detect problems in real time, i.e., before customer complaints flood the customer service center. However, monitoring a large-scale system in this way would be prohibitively expensive or even impractical. Instead, operators can track user session times, which is much more cost-effective. Since users are more sensitive to certain system performance factors, a series of unusual departures over a short period might indicate abnormal system conditions and thus automatically trigger appropriate remedial action.

6. CONCLUSION

Unlike system-level performance, user satisfaction is intangible and unmeasurable. The key to addressing this problem is our ability to measure user opinions objectively and efficiently. In this work, we have proposed a generalizable methodology, based on survival analysis, to quantify user satisfaction from session times, i.e., the length of time users stay with an application. The results of two case studies show that session time is strongly related to system performance factors, such as network QoS, and is thus a potential indicator of user satisfaction. With the derived model, service providers can further improve user experience and optimize resource allocation.

References

[1] 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," Proceedings of IEEE INFOCOM'06, Barcelona, Spain, pp.1–12, April 2006.

[2] K.T. Chen, C.Y. Huang, P. Huang, and C.L. Lei, "Quantifying skype user satisfaction," Proceedings of ACM SIGCOMM'06, Pisa, Italy, pp.399–410, Sept. 2006.

[3] D.R. Cox and D. Oakes, Analysis of Survival Data, Chapman & Hall/CRC, June 1984.

[4] ITU-T Recommendation P.800, "Methods for subjective determination of transmission quality," 1996.

[5] U. Jekosch, Voice and Speech Quality Perception Assessment and Evaluation, Springer, 2005.

[6] A. Rix, J. Beerends, M. Hollier, and A. Hekstra, "Perceptual evaluation of speech quality (PESQ) - a new method for speech quality assessment of telephone networks and codecs," Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing, pp.73–76, 2001.

[7] ITU-T Recommendation P.862, "Perceptual evaluation of speech quality (PESQ), an objective method for end-to-end speech quality assessment of narrow-band telephone networks and speech codecs," Feb 2001.

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

[9] T.M. Therneau and P.M. Grambsch, Modeling Survival Data: Extending the Cox Model, 1 ed., Springer, August 2001.

[10] D.R. Cox and E.J. Snell, "A general definition of residuals (with discussion)," Journal of the Royal Statistical Society, vol.B 30, pp.248–275, 1968.

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

[12] D.M. S. Ila and D. Lam, "Comparing the effect of habit in the online game play of australian and indonesian gamers.," Proceedings of the Australia and New Zealand Marketing Association Conference, Adelaide, Australia, Dec. 2003.

[13] 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," NetGames '04: Proceedings of the 3nd Workshop on Network and System Support for Games, pp.144–151, ACM Press, 2004.

[14] 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.

[15] F.E. Harrell, Regression Modeling Strategies, with Applications to Linear Models, Survival Analysis and Logistic Regression, Springer, 2001.

Te-Yuan Huang received her B.S. in Computer Science from National Chiao Tung University, in 2006. She is currently a Master's student in the Department of Electrical Engineering, National Taiwan University. Her research interests include computer networking with a focus on network traffic measurement, analysis and modeling, as well as quality of service and performance evaluation.

Kuan-Ta Chen received his B.S. and M.S. in Computer Science from National 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, where he is currently an assistant research fellow. His research interests include multimedia networking, Internet measurement, network security, and entertainment networking. Much of his recent work

has focused on the analysis and design of networked multimedia systems, including traffic characterization, transport protocols, quality of service, human factors, and security issues. He is a member of ACM and IEEE.

Polly Huang received her B.S. in Mathematics from the National Taiwan University in 1993 and her Ph.D. in Computer Science from University of Southern California, Los Angeles, in 1999. She is currently an associate professor in the Department of Electrical Engineering and the Institute of Networking and Multimedia of National Taiwan University. Prior to joining NTU, she worked in the Computer Engineering and Networks Laboratory (TIK) of the Swiss Federal Institute of Technology (ETH Zurich) as a post-doctoral research scientist. Polly is interested in the design, analysis, and application of communication systems in general.

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 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, the design and analysis of algorithms, and operating system design. Dr. Lei is a member of the Institute of Electrical and Electronics Engineers and the Association for Computing Machinery.