A new user behaviour model and QoE determination on Short-Message-Service

Marcelo Fiori
IMERL - IIE
Facultad de Ingeniería, UdelaR
Montevideo, Uruguay
mfiori@fing.edu.uy

Julio Fitipaldo
Facultad de Ingeniería, UdelaR
Montevideo, Uruguay

ABSTRACT

This work proposes a method for end-to-end Quality of Experience determination for Short-Message-Service based on a new user behaviour modelling and end-to-end QoS measurements. The suggested model includes several relevant properties of SMS users behaviour that have not been considered in previous works, and improves Quality of Experience evaluation by distinguishing different behaviour states. The proposed method was implemented and has been tested for a period of time on a local operator network, proving that it is effective for QoE determination, providing consistent results, including reliable real time fault detection.

Categories and Subject Descriptors

C.4 [**Performance of Systems**]: [Measurement techniques, Modeling techniques, Performance attributes];
G.3 [**Probability and Statistics**]: [Markov Processes, Queuing theory]

General Terms

Performance, modeling, measurement.

Keywords

User behaviour modeling, Performance measurement, QoS, QoE, Short-Message-Service, SMS, GSM.

1. INTRODUCTION

Today, cellular networks are widely extended being the support for several telecommunications services. New mobile data communication services have been adopted massively, besides the mobile telephone service (which is still the main mobile service). Particularly, in Latin America the Short-Message-Service (SMS) has become the first communication media between people and the second source of revenue for mobile network operators. In other numbers, SMS is the most widely used data application on earth, with more than 3 billion active users ([2]).

Two different kinds of SMS can be distinguished: *customer to customer* and *application to customer*. Although the first one is the most important in terms of volume, the second one represents a considerable source of revenue.

Initially GSM[1] cellular networks were developed to basically offer mobile telephony service. SMS was supported from the beginning but it needed an external push to attract users. When e-mail turned into the most widespread peer to peer communication tool, SMS was found by users as its closest relative in the mobile world.

Nowadays, mobile network operators have some difficulties on Short-Message-Service performance measuring and fault detection. As SMS architecture is based on many different interconnected platforms, some failures can only be detected through customer complaints. Thus, these particular failures can only be detected through end-to-end usage monitoring; however, most of the maintenance and supervision systems are based on individual platform monitoring.

According to [11] around 82% of customer defections are due to frustration for unsuitable service availability, worsened by the operator's lack of effectiveness in dealing with such situations. Moreover, this leads to a chain reaction as, on average, one frustrated customer will tell 13 other people about his bad experiences. The growth of this service has posed a new challenge for measuring and managing the performance, in order to keep the customers satisfied. This paper addresses this challenge, developing a full measurement system capable of delivering SMS Quality of Service and Quality of Experience results.

Specifically, on the one hand we developed a new user behaviour model that describes the user SMS activity more precisely than the classical telephonic models. This model allows the administrator to configure the parameters by simply extracting and analyzing some customer usage data. Additionally, we mapped QoS KPI's[2] on QoE values by using this model. On the other hand, we implemented a software tool able to provide real time end-to-end QoS measurement and QoE determination, SNMP[3] fault alarms and detailed periodic reports, by sending SMS probes through USB/serial connected mobile phones and IP/SMPP[4] based connections.

The rest of the paper is organized as follows: Section 2

[1]Global System for Mobile Application, originally Groupe Spécial Mobile ([1]).
[2]Key Performance Indicators.
[3]Simple Network Management Protocol.
[4]SMPP: Short Message Peer to peer Protocol.

describes briefly the measurement system. In section 3 a behaviour model for Short-Message-Service users is presented. Section 4 describes how a Quality of Experience value is obtained taking into account the user model. In section 5 we give a brief introduction on how the system is implemented and the potential interested groups on using this tool. Finally, section 6 presents the results taken over a real local GSM network, and conclusions are given in section 7.

2. GENERAL SYSTEM DESCRIPTION

The measurement system is based on an underlying process that sends SMS probes across the hole network, for example, from a mobile station to another mobile station. In other words, the analysis is based on an end-to-end measurement. Specifically, the implemented system is capable of managing several interfaces in order to send and receive SMS, namely, mobile stations connections and IP/SMPP connections that, for example, emulate value added services. This allows the system to test the performance of both the *customer to customer* service (sending SMS between mobile stations) and the *customer to application* service (sending SMS from a mobile station to a SMPP interface). In figure 1 we illustrate a simplified version of the system set up. The measurement system is refered as *System* and four different connections to the GSM network are shown: two AT[5] connections (connected to two mobiles), and two SMPP connections. As an example, SMS probes generated by the system could be sent through one of the mobile (via AT connection) and received by the system via the other mobile. Another of the many possible paths is detailed in the figure (green): from a mobile to a SMPP connection.

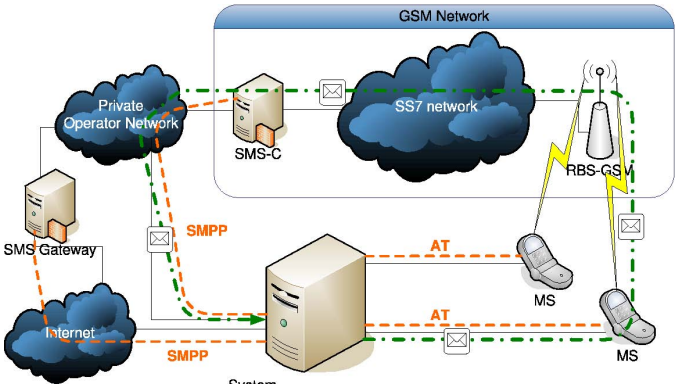

Figure 1: General set up.

For the purpose of knowing the experience of the users, the end-to-end approach provides a more realistic measurement (closer to real experience) than evaluating the performance indicators of the service involved equipment. This approach also lets us measure over the network without the consent of the operator. On the downside, there is a trade-off between the measurement accuracy and the probe traffic overload in the network.

The measurement scope is limited to the network core, access network is not considered in the process. This is why we should take special attention on possible access network introduced errors. In particular, the inter-probes period must be long enough in order to avoid signaling overload on the access network layer, and the physical system location should guarantee a congestion-less scenario. In figure 2 we show which is the surveyed network region.

[5]AT: Hayes command set, originally developed for the Hayes Smartmodem 300 baud modem. The ETSI GSM 07.07 (3GPP TS 27.007) specifies AT style commands for controlling a GSM phone or modem.

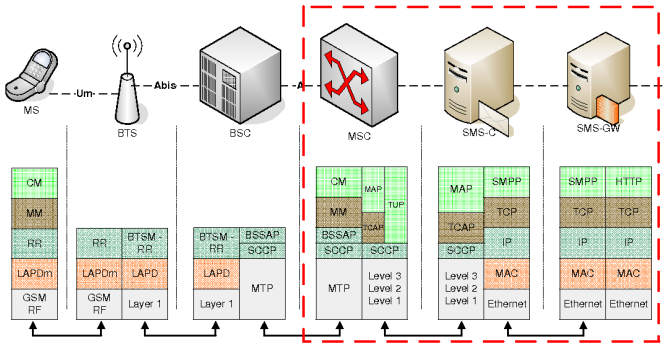

Figure 2: Protocols and scope. Measurement scope in red.

The system measures end-to-end delivery time[6] as the basic SMS KPI. This measurement is done periodically, filling a large grid (stored in a database), leading to direct real-time QoS values. Also, based on the behaviour model, a big number of users are simulated, generating date and time of each simulated message for each user. This messages times are located in the grid obtaining the respective end-to-end delivery time. This way, we gather performance information for each simulated user, which is used for QoE calculation along with specific message characteristics.

3. USER BEHAVIOUR MODEL

Traditionally, the application of the probability and statistics tools were focused mainly for tasks such as network dimensioning. In this regard, the aim was to model the users from the viewpoint of the network, but not to model each user behaviour individually. In this section we propose a model with this last goal, specifically to model the SMS users behaviour, which is still less studied.

The classical approach for this problem in telephony is the well known Erlang's model, i.e. the events (calls in this case) arrive into the system as a Poisson process with intensity λ_i and each one has a random exponential duration of parameter μ_i. Nevertheless, this approach is simplistic for the Short-Message-Service because of mainly two reasons: it does not describe the interaction between people via SMS, and it assumes a unique distribution (and importance) for all the events. Additionally, there is no chance of modeling the activity difference of the people at different times of the day. These weaknesses suggest an extension of the classical model.

On the one hand, the Poisson process is a particular case of pure-birth processes, which is the simplest example of birth-dead processes, which in turn are a special case of continuous time Markov processes (or CTMC for Continuous Time Markov Chains). On the other hand, another generalization

[6]Time between sending a Short Message to a Short Message Center and receiving the very same Short Message on another mobile equipment[3].

of the Poisson process is the Non-homogeneous Poisson Process (NHPP), in a few words, a Poisson Process with rate parameter $\lambda(t)$, a function of time. Both generalizations (CTMC and NHPP) will be used in order to solve the disadvantages. In the following, modifications and extensions are described to remove these shortcomings.

3.1 Two state model

The basic idea is to distinguish messages classes. For example, in this subsection we distinguish the messages that we call "isolated" from the messages with responses. The first ones correspond to messages the user sends without expecting responses[7]. The other ones correspond to interactions between users, questions and answers, messages for setting up a meeting, etc.

This separation partially solves some of the issues presented at the beginning of this section. Namely, since the model described below contains the classical model as a particular case, it describes more precisely the user behaviour. As another point of view, this separation allows the model to treat differently each kind of message, in terms of "importance"; while in an "isolated" message is hard for the user to notice a delay, on a conversation scenario it is not only easier to notice the delay, but also more disturbing for the user.

The two-state model is illustrated in figure 3, and detailed below.

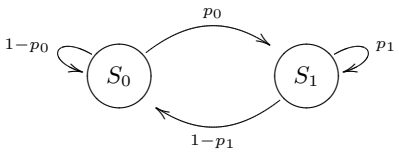

Figure 3: Two state model.

State S_0 represents the user while "waiting" to send a message, actually doing nothing within the Short-Message-Service (generally most part of the time), and the statistical properties are the same as the classical ones, i.e, the time period until the user sends a message is modeled by an exponential distribution. When the event occurs, two possibilities split: if the message is an "isolated" one, then the model stays at state S_0 (in particular, the next message will be sent at a time governed by the exponential distribution mentioned before); if the message starts a conversation the model moves to the state S_1, which represents an interaction scenario. The random variable that represents the time between messages at the state S_1 at first has a different distribution than the distribution at S_0 (it is natural to think that it will be "faster"); in fact it is not necessarily exponential.

In the graph at figure 3, the messages sent at the state S_0 have a probability p_0 of being the originators of a conversation, and a probability $1 - p_0$ of being "isolated". Similarly, the messages sent at the state S_1 have a probability p_1 of being the end message of a conversation, and a probability $1 - p_1$ of being a message at the middle of the conversation.

[7]Typically messages such as "When you get a minute call me", "On way home now" or "Please turn the oven on".

This informal description is considerably similar to a Continuous Time Markov Chain, but it is not possible to consider this model as a CTMC, mainly because of two reasons. Namely, there can be no loops at CTMC, and the distribution of the sojourn time at each state must be exponential.

Formalizing, the stochastic process has state space $S = \{S_0, S_1\}$. However, when looking at just the state through time, we are not able to distinguish messages sent without changing state (loops: "isolated" messages at S_0 or middle-conversation messages at S_1). This simple technicality can be solved in at least two ways: by adding additional states (avoiding loops), or by adding a message counter. Considering this last option, the stochastic process X is then $X : \Omega \times T \to S \times \mathbb{N}$, where T (time) is the parameter space $T = [0, \infty)$.

3.2 The n-state model

Although the process just presented improves the classical user model behaviour, in this section we introduce a natural generalization, which does not significantly increase complexity.

Inspired by the fact that the effect of delays depends on the conversation status (particularly in question and answers applications from an added value service provider), we added states in order to count the messages at each conversation, and treat the performance of each message differentially.

This model is the natural extension of the previous model, the state space is $S = \{S_0, S_1, \ldots, S_n\}$, and the process is $X : \Omega \times T \to S \times \mathbb{N}$. The state S_0 still represents the user while "waiting" to send a message, but the state S_1 now means that the user sent a "non-isolated" message (in this case, this SMS started a conversation) and will send the next message soon. From this state the user will go back to the state S_0 (in case this second message ends the chat), or move to the state S_2 (if the message is an inner-chat message).

As the model has a finite number of states, from the last one (S_n) the two possibilities are to go back to S_0 (as from the other states) or to stay at S_n with chat messages.

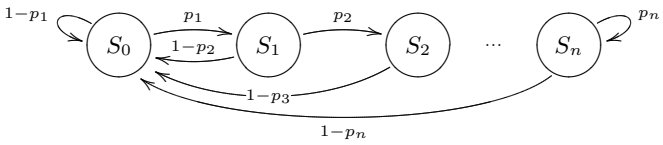

Figure 4: The n state model.

Unlike the previous model, this generalization allows us to differentiate the distinct messages of a conversation, specifically the importance of the message[8] as well as the probability distribution that governs the waiting time until the next message.

As mentioned above, the classical model presents mainly three shortcomings when used to represent SMS users (once again: it does not describe the interaction between people via SMS, it assumes a unique distribution and importance for all events and there is no chance of modelling the activity according to the time of the day). So far, the presented

[8]For example, the delivery time impact can be completely different in the start of a conversation than in the middle of one.

model solves the first two of them, but the time dependence is still a problem. In the following section we tackle this last disadvantage.

3.3 Time dependence

In this section we present three ways to extend the model in order to solve the mentioned problem.

The simplest one consists of taking m rates $\lambda_1, \lambda_2, \ldots \lambda_m$ for the distribution of the sojourn time at S_0. This way, the 24 hours of the day are partitioned in m intervals, during each one the distribution is exponential with parameter λ_i. Notice that due to the memoryless property of the exponential distribution the transition is straightforward.

The second one is similar to the first one, but with a "smooth" transition between the different rates. The state S_0 now splits in m states (with the same properties as S_0 except for the rate), and these processes are switched between by an underlying Markov process. This setting is commonly known as Markov-modulated Poisson process (MMPP).

The third approach is based on the non-homogeneous Poisson Process, which is basically a Poisson process with a variable intensity (rate) defined by the deterministic piecewise differentiable function $\lambda(t)$. As the goal is to have a distribution probability at S_0 (the states S_i, $i \geq 1$, as middle conversation states, do not depend much on the time of the day) that represents activity variations by time of day, we take the first arrival of the non-homogeneous Poisson Process as the waiting time distribution at S_0.

Elegance and flexibility of the last solution have led us to choose it and implement it. An analytical expression of the inter-arrival time distribution of non-homogeneous Poisson Process ([6]) and several indirect methods ([9],[12],[4]) allow efficient simulation of the waiting time distribution at S_0.

4. QUALITY OF EXPERIENCE DETERMINATION

In order to understand the user behaviour and its service perceived experience, we developed a user behaviour model, as explained in the last section. This section tries to tackle the measurement of the perceived experience based on the mentioned model, specifically, the mapping of QoS parameters and user behaviour onto QoE results.

There are several factors interacting to determine the users QoE, including cost, reliability, availability, usability, utility and fidelity ([11]). We focus on reliability and availability only, considering that operator network performance only depends on these two factors.

Figure 5: Quality of Experience levels [11].

The main Quality of Experience KPI considered is based on the SMS end-to-end delivery time, and it takes values on the set {excellent, very good, good, fair, poor}, also respectively referred as $\{5, 4, 3, 2, 1\}$. These Quality of Experience levels are standard.

We first assign a QoE value to each sent SMS. This assignment is based on the delivery time of the message and the state of the model at which the SMS was sent. Specifically, for each state a (configurable) five level thresholds table is used. In other words, the quality of experience value depends not only on the SMS delay but also on the nature of the message itself.

Based on the several QoE values, we then compute a single QoE value for each user, as a function of the latest messages parameters, namely, its QoE value and its time-stamp.

5. IMPLEMENTED MEASUREMENT SYSTEM

The implemented measurement system consists of a server running a Java Virtual Machine and a Web Server over GNU/Linux. The main software module runs on Java, and the GUI is a web application. The whole application was successfully tested on a *Pentium I* PC. The software is licensed under GPL, and it is available for downloading at `http://www.fing.edu.uy/~mfiori/ast.tar.gz`

Mobile phone to PC communication is based on *AT* protocol; implementations of this protocol differ from one manufacturer to another. However, a big effort was made on providing compatibility to a big number of mobile phones, including manufacturers as *Sony Ericcson*, *Ericcson*, *Nokia*, *Motorola*, *Telular* and *Huawei*. Some of these presented stability problems on receiving messages under stress conditions[9].

The content of SMS probes is a relatively small sequence number (about six digits), but we could not notice performance differences using distinct lengths of messages.

The implemented system allows the user to add as many AT and SMPP connections as desired. Each of these connections is handled by a core thread, which periodically generates SMS probes, sends them via one connection and receives them later via other connection. All this process is time-stamped in order to get the delay of each message.

The web interface allows the user to:

- Configure connections - type of connection, phone number, service provider, etc.
- Configure SMS paths - each path specifies the source and destination connections, for example, from mobile phone number 1 to SMPP connection number 3 (the number of paths is unlimited).
- Configure time period between SMS probes.
- Configure user behaviour model parameters - number of states and probabilities.
- Configure Quality of Experience thresholds - QoE thresholds for each state.
- Get measurement reports - Get statistics of QoS and QoE results in *pdf* format.

The application was developed in order to comply with operator requirements. Pursuing this goal, several characteristics and features were added, particularly snmp-capability both for measurement data collection and fault reports (traps), remote web management, stability, etc.

[9]We refer to stress conditions when more than one message per minute is received, during at least one day.

Nevertheless, its application is not restricted to network operators only, we found several potential interested groups including:

- Mobile network operators, in order to measure their own network performance or even the competition one.
- Regulatory bodies, in order to assure complying QoS and QoE normatives, if any.
- Large customers (value-added service providers), in order to assure operator complying of SLA (Service Level Agreement).

6. RESULTS

This section introduces results from measurements over a real GSM network. In particular, these results were taken from one Uruguayan operator, the system worked non-stop through the whole trial period, and the location was fixed on a congestion-less cell in Malvin neighborhood (Montevideo).

Two mobiles were used in the experiment, namely, a *Huawei E220* and a *Ericcson BVF221m*, an IP/SMPP connection was established directly to the SMS-C through the operator private network, and another IP/SMPP connection was established through an Internet connection to the SMS-Gateway. This allowed us to detect failures and different performances through the different network access points.

During the trial period, all faults were detected by our system, generating *snmp* alarms, and no false positives were given.

In the following subsections we present results and their analysis divided by normal and fault situations, including also a division by QoS and QoE values in each situation.

The QoS analysis is included not only because of the analysis itself, but also for providing a vision of the network state over which the Quality of Experience is studied.

For the purpose of evaluating QoE, we have simulated 800 users divided in two groups. These two groups represent "light" and "heavy" users. The former sends a mean of three messages per day (most of them "isolated" ones), setting a two-state model. The latter represents users with a bigger level of activity, with a mean of ten chats per day (with one or more messages each) and modeled by three states.

This section aims to show how network performance variations are actually perceived by the service users. Quality of Experience gives us the possibility to inform operators managers in a way that they can perceive the real impact of network performance over the users, much more effectively and directly than QoS.

This way of showing network performance (in a user oriented perspective) allows managers to take decisions according to different users types experiences. For example, in some cases "chatty" users may be much more affected than other users, moreover, they may be the only ones that realize the service degradation, therefore it might be good to compensate them in accordance.

6.1 Normal situation scenario

We refer as Normal situation scenario where there is not a specific fault on the system, and the behaviour of the service is statistically the same as most of the time. The following analysis is based on a sample taken between August 23 and August 29, 2008.

6.1.1 QoS values

We found that under normal conditions the mean end-to-end SMS delivery time between mobiles was about 6.8s. We must take into consideration that each mobile equipment introduces a systematic error, which is due to different mobile software and hardware implementations. Nevertheless, the maximum error value found was about one second. According to [8] the delay must be less than 10 seconds to be acceptable, considering this value, the measured network complies with this recommendations. As expected, delay of SMS sent and received from SMS-C through SMPP connection, was less than mobile-mobile delivery times. Namely, mobile to SMS-C delay mean value was 3.8s, and SMS-C to mobile delay mean value was 5.0s. The standard deviation in all cases was about 0.5s. This value is basically affected by RBS buffers and depends on the channel control congestion; given the obtained values we assume that this behaviour depends on the RBS, and does not represent QoS globally.

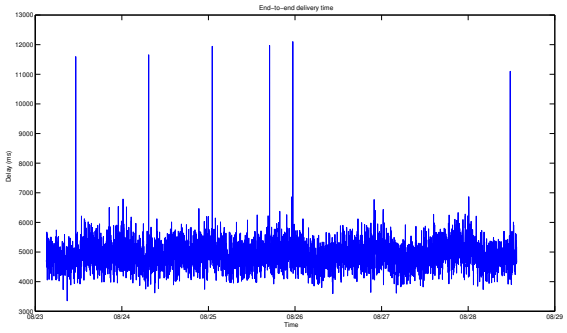

Figure 6: End-to-end delivery time.

Figure 6 shows five days end-to-end delivery time behaviour for messages sent from SMS-C to the mobile equipment. Figure 7 shows a histogram of the same data. Some peaks were found on the sample; these can be clearly seen in both figures and may be explained as follows: most of the messages are routed by the FDA (First Delivery Attempt) entity, and a very little portion of them are routed by a Store-and-Forward entity.

6.1.2 QoE values

For this section analysis, we used the same time period that has been used for QoS analysis on section 6.1.1.

Figure 8 shows percentage of "light" users who experienced each level of QoE versus time. Taking into consideration that the best level of experience is most of the time experienced by closely 100% of the users, this level was not included in this figure, in order to appreciate variations on other levels[10]. In particular, just a 0.25% of the sample had

[10]Anyway, this value can be computed as the complement of

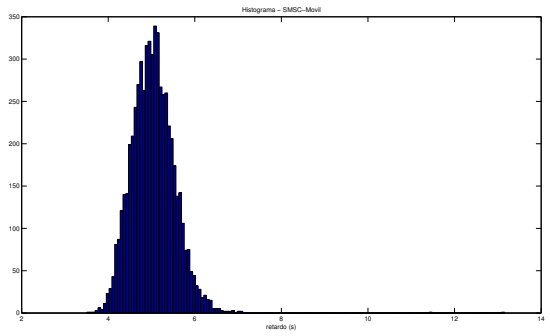

Figure 7: End-to-end delivery time histogram.

reached in one moment a level 4 (*very good*) experience (one user of the 400 simulated), and the rest of them had a level 5 (*excellent*) experience.

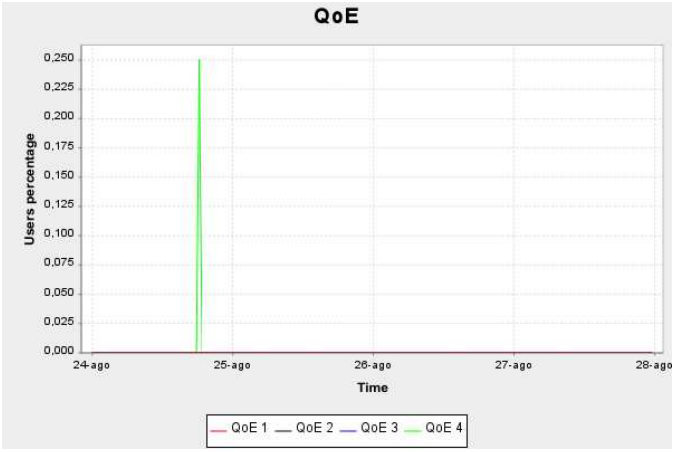

Figure 8: Quality of Experience values for "light" users.

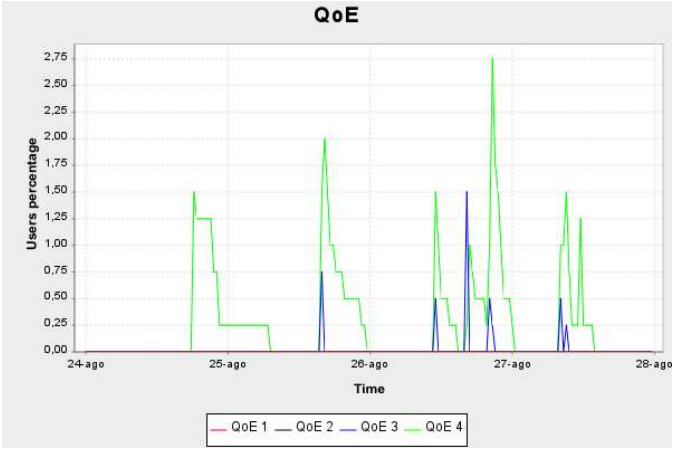

Figure 9: Quality of Experience values for "heavy"users.

the other four levels.

On the other hand, figure 9 shows similar representation but for "heavy" users, during the same time period. This users perceived the small performance fluctuations shown in figure 6. In fact, the percentage of users that experienced some level under 5 is one order of magnitude greater than the last analysis ("light" users). There had been also some occurrences of level 3 (*good*) experience in the time sample, and events are more frequent.

Even at normal conditions, where QoS parameters are under the recommended threshold, we found that some type of users appreciated small performance variations, but others did not.

6.2 Fault scenario

We refer to fault scenario, when the performance of the network is clearly affected by a problem, and the behaviour differs from normal situations. We found two of these situations along the testing period, both with similar characteristics, and we present one of them below.

6.2.1 QoS values

We found that under this fault situation, SMS sent from one mobile to the SMS-C SMPP connection, were not affected. Because of this behaviour, we can conclude that the problem was not local on the RBS, or on the Backhaul network, but on the SMS core.

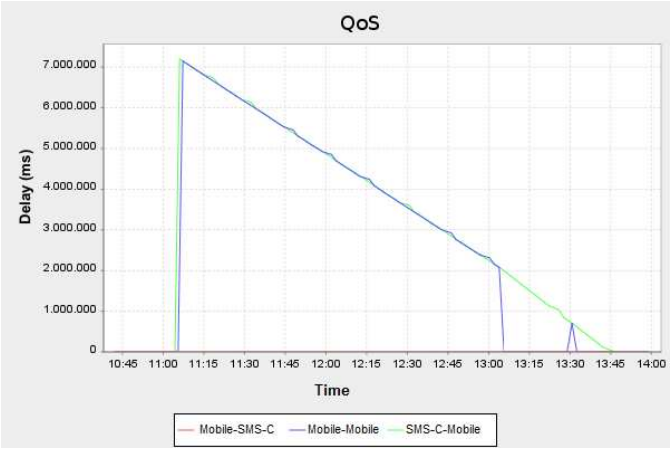

Figure 10: End-to-end delivery time. Fault scenario: August 22, 2008.

The problem was solved roughly at 1:05 PM, this can be justified in at least two ways: on the one hand, mobile-mobile SMS delays began to be normal at that time. On the other hand, the first message affected by the problem was sent at 11:05 AM, and arrived 7200s later (2 hours).

Although the problem was solved at that time, the messages sent from the SMS-C SMPP connection kept having greater delays than normal, because they were queued in a FIFO queue. The slope of the line is related to the FIFO queue dispatch speed. Messages sent from the mobile phone, after the problem was fixed, were dispatched through FDA, so they did not get into the queue (except one single message around 1:30PM).

6.2.2 QoE values

Quality of Experience values were taken over a five day

period as in section 6.1. This time period includes August 22, day in which the problem just described at section 6.2.1 took place. The analysis is based on the simulations of the 800 users, divided in "light" and "heavy", same as before.

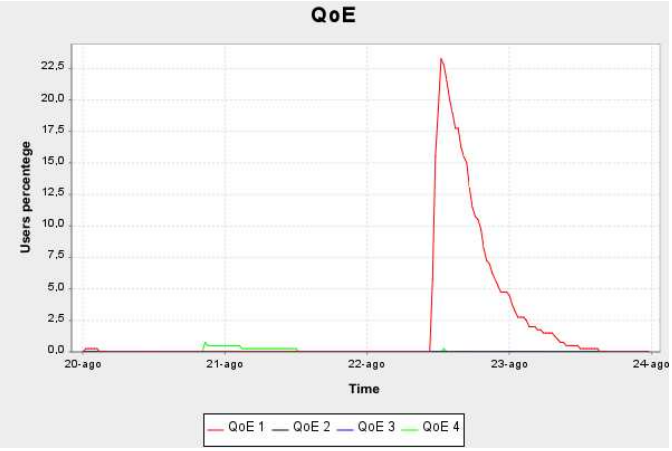

Figure 11: Fault scenario. Quality of experience for "light" users.

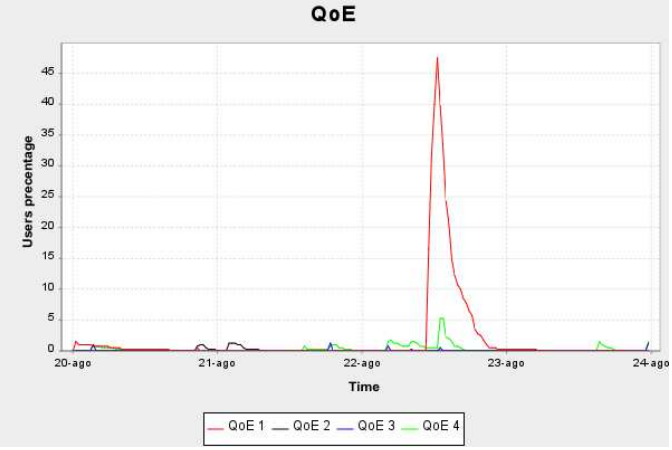

Figure 12: Fault scenario. Quality of experience for "heavy" users.

Figure 11 shows QoE values of "light" users, while figure 12 shows "heavy" users results. In both of them, the previous described problem stands out.

We first compare the percentage of users who realize the fault: around 20% of the first type noticed it (with a *poor* QoE), while almost half of the second type suffered the problem. In addition, around 3 percent of "heavy" users had several experiences of level 1 and 2 (*poor* and *fair*), out of the fault period.

We should point out that in QoS delay graphs (figure 10), small events go unnoticed and it is difficult to estimate their impact on users, while in QoE graphs these events can be appreciated, and their impact can be directly estimated.

7. CONCLUSIONS

In this work we have presented an end-to-end measurement system for SMS performance and a SMS user behaviour model. This model includes several relevant properties of SMS users behaviour that have not been considered in previous works, and improves Quality of Experience evaluation by distinguishing different behaviour states.

Several experiments have been done in order to test the robustness of the measurement system and the model over a local operator network, obtaining very good results. During this testing period the system was capable of detecting all faults with no false-positives, and describing the users experience over these faults periods and also under normal operation.

The developed system can be used over any GSM network without the need of a permission from the operator. This allows the use of this system not only by the operator (network owner) but also by regulatory bodies, large customers and competition network operators. The only requirement is a low-resources PC with GNU/Linux and a Java Virtual Machine.

Due to privacy policies we had no access to real individually identified users behaviour data. Given this data the estimation of model parameters still remains open for further research.

8. REFERENCES

[1] 3rd generation partnership project. `http://www.3gpp.org`.

[2] Tomi T Ahonen. *Tomi Ahonen Almanac 2009: Mobile Telecoms Industry Annual Review*. 2009.

[3] GSM Association. Permanent reference document ir 42 definition of qos parameters and their computation. Technical report, GSM Association, Abril 2007. Version 3.3.

[4] Luc Devroye. *Non-Uniform Random Variate Generation*. Springer, April 1986.

[5] P. Griffiths G. Peersman, S. Cvetkovic and H. Spear. Global system for mobile communications short message service. *IEEE Personal Communications IEEE Wireless Communications*, 7:15–23, 2000.

[6] R. Shcherbakov D. Turcotte G. Yakovlev, J.B. Rundle. Inter-arrival time distribution for the non-homogeneous poisson process. *arXiv:cond-mat/0507657v1*.

[7] Architecture & Transport Working Group. Triple-play services quality of experience (qoe) requirements. Technical report, DSL Forum, December 2006.

[8] ITU-T. G.1010 - end-user multimedia qos categories. Technical report, ITU-T, 2001.

[9] Patricia Kisbye. Generación de eventos en procesos de poisson. Facultad de Matemática, Astronomía y Física, Universidad Nacional de Córdoba, 2008.

[10] L. Kleinrock. *Queueing Systems, Volume 1: Theory*. John Wiley & Sons, 1975.

[11] Nokia. Quality of experience (qoe) of mobile services: Can it be measured and improved?, 2004.

[12] Sheldon M. Ross. *Introduction to Probability Models, Eighth Edition*. Academic Press, January 2003.

[13] J. Virtamo. Queueing theory / probability theory. `http://www.netlab.tkk.fi/opetus/s38143/luennot/english.shtml`.