Online Game Quality Assessment

Functional Specification

Luca Venturelli C00164522

Contents

Chapter 1

What is the project supposed to be?

The project consists in developing a metric for online games. This metric will estimate the effective user perception of the game quality by taking as parameter not only the estimation of the network quality, but also the players experience and the I/O parameters. In order to collect the data from the users and to test the metric, an environment for testing will be implemented and a plug-in will be integrated in the game Doom 3. The language we are going to use to implement the plug-in is C++. The plug-in will be also integrated into Torque 3D engine and Ogre3D. Torque 3D was one of the best commercial game engine platforms. A lot of games have been developed using this platform and the possibility to have full access to the code enables us to fully integrate our plug-in. This engine has very good documentation, it is compatible with all the most important operating systems and has been released under the MIT license that guarantees the reuse of the code in proprietary software. Ogre has been released under the same license. It is one of the most important open source platform for 3D development and thanks to his modularity it enables the developer to easily develop plug-ins. It doesnt provide a network framework but there are a lot of projects that implement this function in Ogre. Its community is the largest and most active between the open source game engines At the end of the project a research paper will be proposed describing the results of the tests made and the new metric developed.

Chapter 2

What is the application supposed to do

The development of the application will follow four main phases. In the first phase we will develop a plug-in to automatically evaluate the game quality based on the metric developed. In the second phase a test environment will be build and a first session of tests with a group of users will take place. In the third phase the data collected will be analysed and new parameters for the metric will be proposed. The fourth phase will be a new test phase in which the experience of the user will be compared with the result of the metric evaluation.

2.1 Plug-in implementation

In this part of the project we will develop a plug-in able to give us an opinion score. The plugin will analyze the quality of the network and will give a score to the hardware where it is installed. It will collect all this data and put it into a database, and calculate the MOS. We will analyze the engine in which we are going to integrate our software and then we will start developing the plug-in. The plugin will have the following functionalities:

1. Hardware analysis: analysis of the hardware of the pc using C++ functions such as GetSystemInfo for the CPU or GlobalMemoryStatusEx for the RAM.
2. Network analysis: analysis of the network quality using the libraries provided by the game engine. This analysis will be performed interacting directly with the server responsible for the game management. In this way we can have a more precise estimation of the network quality.
3. User experience: we will ask the user to rank his game experience. In this way we will have a subjective opinion on the quality of the player.
4. Metric implementation: it will also implement the metric developed and will save the MOS calculated into the database.
5. Database interaction: the plug-in will store all the data in the database installed in our main server.

2.2 Test 1

This phase consists in the set-up of the environment. The model proposed will take into consideration the following parameters:

1. End user experience: the experience of the user will be based on the average of the players score in the game played.
2. Distortions introduced by game client equipment (graphic card, memory, processor) and I/O devices (mouse/joystick,screen, keyboard): each component will be rated according to a ranking of the hardware used. The values obtained will be added together with a different weight (that will be estimated during the tests) and then it will be divided by 6 (number of components considered).
3. Distortions introduced by the network
4. Distortions introduced by the game server

To test the network quality all the traffic generated by the two players will pass through an Ubuntu server. In order to do that the server will be set-up as the gateway of the two client machines and every client will have a different LAN in order to force all the traffic to pass through our server. In this way we can monitor all the traffic generated by the clients and we will also be able to modify the quality of the network of the different clients independently. We have decided to emulate the network parameters delay, jitter and packet loss. The network emulator Netem will be used in order to vary ping, jitter and loss. Netem is by default enabled in the kernel of the Linux distributions. Netem is used in conjunction with the Traffic Control application, TC. TC allows us to specify the queuing discipline used for sending outbound packets on an Interface. The server we are going to set up will have a Debian-like distribution. The reason of this choice is basically because we are more familiar with it and the set-up of the environment will be easier and faster. We will use 3 different kind of machines with different characteristics to analyse the impact of the hardware on the quality of the game. Finally we will also collect informations about the level of experience of the player asking him about his experience on playing fps games. A selected group of players will play the game Doom 3. Every 5 minutes the parameters of the network will be changed and the testers will be asked to give their opinion of the gaming quality. They can select one of the following five opinion scores according to the ITU-T ACR scale [3] :

- 5:Excellent gaming quality;
- 4: Good gaming quality;
- 3: Fair gaming quality;
- 2: Poor gaming quality;
- 1: Bad gaming quality.

The players have to assess different kinds of scenarios. We will analyse impairments caused by a single factor, as well as impairment consisting of combinations of ping, jitter, and packet loss. Our metrics will be developed first basing our values on the past studies reported on the literature. For the network part we will try to focus our attention most on the jitter as we know from our research that is one of the most important factor for network evaluation. The players will be asked also to play on three different machines and to give their opinion of the gaming quality.

2.3 Data analysis

In this part of the project we will analyze the data collected and we will propose a weight for each factor. This procedure will allow us to create our model. Network impairments can then be passively or actively measured on any given network and the likely subjective effect on various games calculated. The completion of these tasks will allow us to create our model. Network impairments can then be passively or actively measured on any given network and the likely subjective effect on various games calculated. Using the data collected a Game Rating Factor (GRF) is proposed. The Game Rating Factor is inspired from an ITU-T recommended computational model (E-model)[6]. The model is used to assess the combined effects of variation in several parameters that may affect end-user perception of speech quality. The GRF can be described as follows: a maximum value that reflects the highest level of game quality will be reduced in proportion with the distortion caused by various impairment parameters. As follow the formula to calculate the GRF:

$$GRF = GRFMAXIGCDINIGS + A \tag{2.1}$$

where GRFMAX is the maximum GameRatingFactor(90), IGCD is the impairment factor representing all impairments due to Game Client and I/O device, IN represents all the impairments due to network connection between the game server and game client, IGS is the impairment factor representing all impairments due to Game Server and A represents the end-user experience with online games. The GRF is then related with the Mean Opinion Score that is a value between 1 and 5 where 1 is bad and 5 is excellent. The programming language we are going to use for the plug-in will be C++ and for the database we will use Mysql.

2.4 Test 2

In this part of the project a new test will be arranged to compare the user perception with the metric developed. We will start new tests taking different testers and we will propose new values of jitter, packet loss and ping, and a different hardware will be used for this test. The players opinions will be compared with the evaluation made by our metric generated by the plug-in and a value R given by the difference between the user evaluation and the metric developed will be calculated. We will be happy if R has a value between 0.9 and 1. In this way we will have proved the consistency of our metric.

Chapter 3

Who is going to be using this application?

The proposed metric, which represents and includes an objective alternative, based on network impairments, to the subjective game assessment, could have several applications:

a) Game service providers: the impact caused by network impairments that may arise during a multi-player game session will be quantified by the MOS. Using the MOS, the game service providers could enhance the online games quality and the overall Quality of Service (QoS).

b) Game players: the MOS can be used to inform the game player about the game quality at a point of time before starting and for the duration of playing the game. If the player's choice is to start/join an online game, the MOS will provide the game player with a continuous feedback about the network state and game quality.

c) Game Developers: game developers could take into account the MOS in developing and or adapting an aspect of the online game which may be affected by network impairments.

d) Network Service Providers: those providers who aspire to the provision of large-scale distributed interactive applications as part of their network service could use the MOS to assist the policing of traffic and control congestion particularly at the access points or in a wireless environment.

Chapter 4

Metrics

Two are the main goals of this project:

1. Consistency of our metric
2. Research Paper: at the end of the project we will write a research paper to describe the results of our research.

We can say that these goals are achieved if the following conditions are satisfied:

1. The measure that we are going to use to express how well a model explains subjective data is the linear correlation coefficient R, also known as the Pearson correlation. The correlation coefficient is related to the coefficient of determination R^2, which is a measure for how well the model explains the variation in the subjective data and can be 1 for maximum correlation. A correlation coefficient $R > 0.90$ is considered high while a correlation coefficient $R > 0.95$ is considered very high. We expect a value of R between 0.90 and 1.
2. The research paper with the new metric developed will be submitted to the Annual Workshop on Network and Systems Support for Games 2013. We will consider our work well done if we will receive a positive feedback from this submission.

Chapter 5

Is there a precedent for this application?

The metric we will develop considers some aspect of the game quality that never before have been considered. Most of the existing metrics consider just the network quality. The network layer is one of the most important aspect to consider while approaching a quality metric for online games. In terms of network awareness user perception in terms of player tolerance is crucial. In [1] the author reveals that there are two possible approaches for discovering player tolerance to network disruption. The first is to build a controlled lab environment in which to test small groups of players under selected conditions and secondly to monitor player behaviour on public servers over thousands of games. As follow an analysis of the metrics proposed to evaluate the game quality and the aspects they cover:

5.1 Quake3 G-model

Quake3 G-model is a model proposed by Ubicom [4] that introduces a new benchmark, OPScore, or the Online Playability Score, to describe the effects of network impairments on the playability of online games. The authors focus their attention on latency, the average amount of time necessary to transmit information about a players actions, and jitter, the variance of latency. This technique uses measurements of traffic in a realistic home network environment to forecast the playability of online games.

The model defines an impairment factor R given by:

$$R = (WLL + WJJ)(1 + E) \tag{5.1}$$

where WL is set to 1 for the test , K is the Average frags per minute, WL is the Latency weighting factor, equal to 1 by definition, E is the Packet loss, as a percentage of bytes lost and R is the Impairment factor in ms.

5.2 Quake IV G-Model

A successive approach proposed by Wattimena [5] uses a similar model named the Quake IV G-Model to predict the perceived quality of a First-Person Shooter. The authors conducted a number of subjective experiments to quantify the impact of network parameters on the perceived quality of a FPS game. The model proposed enables to predict a gamers quality rating based on measured ping and jitter values, and it shows a very high correlation with the subjective data. The final score is given by the following mapping function:

$$MOS_model = -0.00000587X^3 + 0.00139X^2 - 0.114X + 4.37 \tag{5.2}$$

where: X the network impairment is defined as:

$$X = 0.104 * pingAverage + jitterAverage. \tag{5.3}$$

WJ that is the Jitter weighting factor is calculated with the following formula

$$WJ = (\Delta K/\Delta J)/(\Delta K/\Delta L) \tag{5.4}$$

The experiments done demonstrate that ping and jitter have a significant negative effect on both the subjective (MOS) and objective gaming quality (kills), while packet loss goes unnoticed for values up to 40%. Especially the introduction of jitter in the network has a large negative effect on the perceived quality of the players. The model has been developed for the game Quake IV and was not tested on other games and platforms.

5.3 QoE

A recent paper [2] presents a new metric to measure the quality of experience (QoE). In this paper is presented an analysis of the causal relationship among network delay, system inconsistency and QoE. The QoE is divided into responsiveness, precision and fairness. They create a metric to qualify the view inconsistency. A function is built to map from the objective system inconsistency to the subjective QoE property score. The input of this mapping function considers only the relative inconsistency between views that takes into account any built in time compensation algorithms, because the same network delay can produce different inconsistencies in different games and the same absolute inconsistency value can lead to different impacts in different scenarios. The approaches described above have at least one of the following weaknesses:

- Although they claim to estimate a user perception, the parameters used are based on network components which are limited in fully predicting the user satisfaction.

- Implementing limited subjective testing for a short period of time with a small user pool

- While a high level of correlation between the subjective and the proposed models is shown, this could be justified only for one defined game in restricted testing conditions.

- None of the model presented presents a parameter to measure the game client and I/O devices except from the MOS model. This model, by the way, does not provide any information on how to measure the parameters and it is not based on the tests made.

The new proposed model overcomes the weaknesses of the existing models by:

- Combining different parameters such as the user-based experience/knowledge factors and distortions introduced by user equipment.
- Moving from a game-specific model to a wide range of existing online games (including console based games).
- Proposing objective tests and values to estimate the single parameters based on the empiric tests done.
- Using a number of users for subjective testing in line with accepted test models used in telecommunications based scenarios recommended by ITU-T.
- Expecting a level of correlation in excess of 95

Bibliography

[1] G. Armitage and L. Stewart. Thoughts on emulating jitter for user experience trials. *Proceedings of NetGames 2004 Workshop, ACM SIGCOMM, Portland, Oregon, USA*, August 2004.

[2] P. Chen and M. El Zarki. Perceptual view inconsistency: an objective evaluation framework for online game quality of experience (qoe). *Proceedings of NetGames 2011, ACM SIGCOMM, Ottawa, Canada*, 2011.

[3] ITU-T Recommendation P.800. Methods for subjective determination of transmission quality. *http://www.itu.int/rec/T-REC-P.800-199608-I/en*, 1996.

[4] Ubicom White Paper. Opscore or online playability score: A metric for playability of online games with network impairments. *http://www.ubicom.com/news/whitepapers/whitepapers.html*, 2005.

[5] A.F. Wattimena, R.E. Kooij, J.M. Van Vugt, and O.K. Ahmed. Predicting the perceived quality of a first person shooter: the quake iv g-model. *Proceedings of NetGames 2006, ACM SIGCOMM, Singapore*, October 2006.