

*Faculty of Informatics*

*Faculty of Informatics - Papers*

---

*University of Wollongong*

*Year 2006*

---

Fairness and playability in online  
multiplayer games

J. Brun\*

F. Safaei†

P. Boustead‡

\*University of Wollongong

†University of Wollongong, farzad@uow.edu.au

‡University of Wollongong, boustead@uow.edu.au

This article was originally published as: Brun, J, Safaei, F & Boustead, P, Fairness and playability in online multiplayer games, 3rd IEEE Consumer Communications and Networking Conference (CCNC 2006), 8-10 January 2006, 2, 1199-1203. Copyright IEEE 2005.

This paper is posted at Research Online.

<http://ro.uow.edu.au/infopapers/232>

# Fairness and Playability in online multiplayer games

Jeremy Brun, Farzad Safaei, Paul Boustead  
Telecommunications and Information Technology Research Institute,  
University of Wollongong, Wollongong, Australia  
Email: {jeremy,paul}@itr.uow.edu.au, farzad@uow.edu.au

**Abstract**—Playability and fairness are crucial elements of an enjoyable online game experience. The unavoidable propagation time introduced by the geographical distance between participants can deteriorate the game responsiveness and playability. Also, differences in these delays can provide unfair advantages and/or disadvantages to different players.

This paper explores the causal relationships between propagation time, inconsistencies, playability and fairness in online multiplayer games. It clearly defines causes of unfairness and introduces a theoretical quantification supported by an experimental technique to evaluate it in simulations. It also demonstrates through simulations the impact of changes in network architecture and topology on playability and fairness.

## I. INTRODUCTION

Aside from the artistic design and originality of an online game, the network aspect of playability and fairness is of crucial importance to give the participants an enjoyable game experience. Customers will not want to spend time in a game where actions are delayed or other players seem to have inexplicable and unfair advantages.

While bandwidth and processing scarcity can technically be overcome by deploying enough resources, propagation time due to geographical distances is impossible to avoid. Without proper design, the differences in game responsiveness to user inputs can introduce unfair advantages to some players even when there are abundant processing and network resources available to the game. In order to fully explore the impact of this inevitable delay, this paper always assumes uncongested networks and zero processing time on both users' machines and network servers.

Other published works have been discussing the issues of fairness and playability in online games. While playability is commonly understood as being the trade off between *response time* and *consistency* [1] [2] there is no globally accepted definition of fairness for online games and different research groups usually discuss fairness concepts in specific ways that suit a particular situation or model.

Lin, Gua and Paul [3] have defined 'state update' and 'player action' fairness which they maintain using Sync-MS, a synchronisation protocol for distributed games. Fairness concepts have also been associated to game security and cheating. Under this aspect Baughman [4] first introduced the Lockstep synchronisation. Later, improvements and variations were proposed for distributed and peer to peer games such as the New Event Ordering protocol from GauthierDickey [5] and the Fair Synchronisation Protocol from Chen and Maheswaran [6] among others.

Playability in online games has been studied under different forms for both client-servers [7] and distributed or mirrored [2] architectures. Playability and inconsistency in games is also related to research in the area of Distributed Virtual Environments (DVE) which is exploring similar concepts.

The aim of this paper is to provide insights about the causal relations between network propagation time, inconsistencies, playability and fairness in order to discuss strategies to maximise the game experience of players. The main contribution of this work is a clear definition of fairness in network games along with a theoretical quantification and simulation procedures for its evaluation. Depending on the topology, architecture and synchronisation parameters, different types of inconsistencies are introduced due to the propagation delay. We show how different inconsistencies have different impact on users depending on their *utility function* and how it can be beneficial to trade one inconsistency type for another. This prompts us to introduce the concept of *playability space* which leads to a theoretical definition of fairness.

The remainder of this paper is organised as follows: Section II introduces important concepts in online games and basic architectures. The notion of playability related to inconsistencies is explored with examples and illustrations in Section III. Section IV proposes a definition for fairness in online games along with a procedure to quantify it in simulations. Section V presents and discusses playability and fairness in simulations of different network topologies. Finally Section VI will conclude this work.

## II. NETWORKED GAME ARCHITECTURES

In a networked game participants interact in a common virtual world. At any time, the virtual world is fully described by a set of parameters called the *game state*. Such parameters include, but are not limited to, the position and states of avatars and other in-game objects in the virtual world. Each *terminal* renders a projection of the game environment to its player based on game state updates from its associated *decision point(s)*. Decision points are processing locations (either in servers or client machines) that can authoritatively determine changes in the game state. Participants react according to their perception of the virtual world displayed through their terminal. Each action's consequences on the game state are evaluated by a decision point.

There are two main network architectures possible to make decisions and disseminate the game state: central or distributed decisions points.

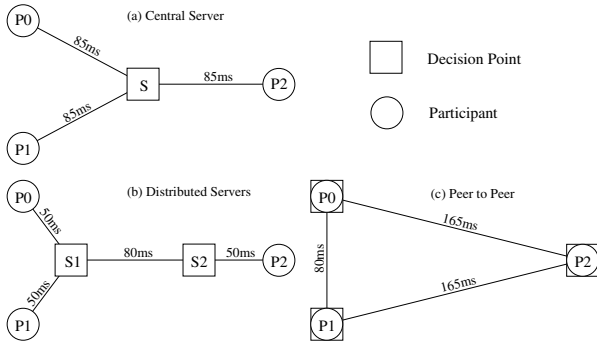


Fig. 1. Network games architectures

Most current online games use the central server approach (Figure 1a). Participants connect to a unique server which is the authoritative decision point for all actions in the game. This server is also responsible for the dissemination of the game state data to the terminals used for the game rendering. It can either be a dedicated server or hosted on one of the participant’s machine. It is technically the easiest architecture to implement since the consistency of the game state is guaranteed by the presence of one unique decision point for all actions.

In a distributed architecture, more than one machine act as decision points and therefore are authoritative on the game state. Distributed servers (Figure 1b) and peer to peer (Figure 1c) are examples of such architectures. While this can help reducing response time by bringing decision points closer to players, discrepancies between the decision points with respect to the ‘current’ game state can create inconsistencies.

### III. PLAYABILITY AND INCONSISTENCIES

In a network game with all participants in the same physical location, ie. with no appreciable network propagation delays (and infinite processing power and bandwidth as we assume throughout this paper), the virtual environment can perfectly obey the ideal laws of the virtual world. In particular, all players should perceive the same game state at the same time, actions should be executed in time and order and the physics of the world respected. Regardless of the type of network architecture used, the addition of appreciable propagation delay between participants, introduces inconsistencies: digressions and sometimes direct violations of the ideal laws, which can negatively affect the playability of the game.

#### A. Types of inconsistencies

The propagation time between a terminal and its associated decision point may introduce a *response time*, which is the delay between the time of the issuance of an action order by a player and the display of the action results on the terminal. A non-zero response time is a digression of the time consistency of the game.

In distributed models, multiple decision points can influence the game state and need to synchronise by exchanging their decisions. If a conservative synchronisation mechanism such as local lag [8] [9] or lock-step [6] is used, consistency between decision point is maintained at the cost of an increased

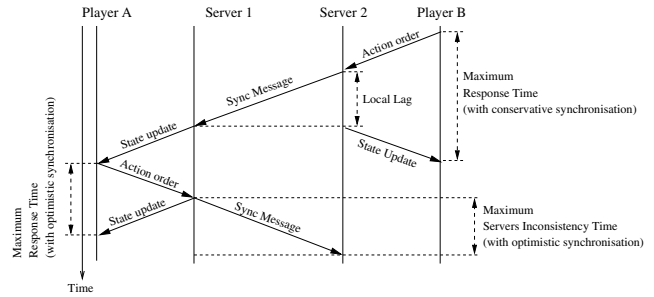


Fig. 2. Inconsistencies in Distributed Server Architecture

response time as shown in Figure 2. However, without the use of conservative synchronisation there may be a discrepancy between different decision points about the game state because of propagation delay between them.

A game state inconsistency is a divergence in the value of one or more game state parameters between two decision points due to the propagation time of a decision. The game state of an inconsistent decision point shall be corrected once the game state synchronisation message reaches it. The inconsistency may cause these decision points to evaluate actions out of order, possibly violating causality and taking incompatible decisions. In order to heal the game state from a paradox, a decision point might have to roll back in time [9], this operation is also referred to as a Timewarp [8].

In optimistic synchronisation, the amount of inter-decision point inconsistency influences the ‘freshness’ of the state updates sent to the terminals. The age of different game state parameters perceived by a given player may not be uniform. For example, in Figure 1b, if  $S1$  takes decisions for  $P1$  positions and  $S2$  is authoritative on  $P2$  positions, and assuming there is no conservative synchronisation involved between the servers, then the perceived position of  $P1$  by  $P0$  is always  $50ms$  old while the perceived position of  $P2$  is  $50 + 80 = 130ms$  old.

#### B. Perceptual impact of inconsistencies

Different inconsistencies, such as long response time or inter-decision point inconsistency leading to a large number of rollbacks, have different perceptual impacts on users. For example in the Unreal Tournament first person shooter game, Quax et al. [10] conclude that a round trip delay above  $60ms$  seriously disturbs players. To capture the results of this empirical study, one could imagine a *utility function* for the response time of such a game being like Figure 3a. A response time from 0 to  $30ms$  may have no impact on players’ comfort (utility of 1) since this would stay below the perception threshold of most people. Discomfort would start appearing from  $40ms$ , growing stronger up to  $100ms$  limit where the average player would be too annoyed to keep playing.

In general, we may derive a different utility function for each type of inconsistency (and for a given genre of game) to describe the relationships between measurable values of an inconsistency and its perceptual impact. However, empirical estimation of these functions is outside the scope of this paper and depends on many parameters including game type,

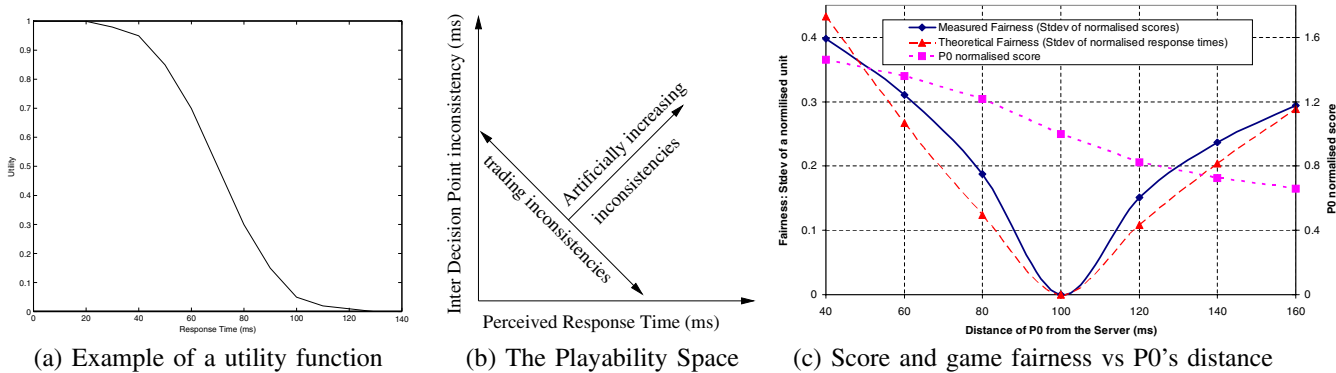


Fig. 3.

individuals and terminal capabilities. For the remainder of this paper, we focus on the objective and measurable values associated with duration of inconsistencies.

The knowledge of perceptual impact of different inconsistency types can be used to improve user comfort level when it is possible to trade one type of inconsistency for another. For example, Figure 2 illustrates how inter-decision point inconsistency can be traded for an increase in response time by adding local lag. Since any amount of inter-decision point inconsistency increase the probability of a Timewarp (see Section III-A), and if a given amount of response time increase is not noticeable by the player, it might be worth it to add as much local lag as the player can handle without discomfort in order to reduce the probability of Timewarp. This operation would maximise the sum of the response time and the inter-decision point inconsistency utility functions.

### C. The Playability Space

As discussed above, the propagation delay introduces different types of inconsistencies depending on network topology and the synchronisation parameters. Modifying these parameters allows trading different types of inconsistencies. Let us consider a multidimensional space created by these inconsistency types, hereafter referred to as *playability space*. If the amount of each type of inconsistency could be quantified per participant, then each player could be positioned as a point in this space.

Figure 3b represents a simple two dimensional playability space taking into account only two type of inconsistencies: response time and inter-decision point inconsistency. The origin point is the ideal case of perfect game playability where there is no inconsistencies whatsoever. Therefore, the closer the user to the origin point, the more playable the game is for this player.

Depending on the perceptual impact of the different inconsistencies, a *playable zone* of this game could be defined as the zone where players would be considered as content to play the game. Players outside the zone would consider the game unplayable. Playable and unplayable areas do not have a simple line boundary but are separated by a transition zone. The shape and boundaries of these zones would depend on the utility functions of the inconsistencies.

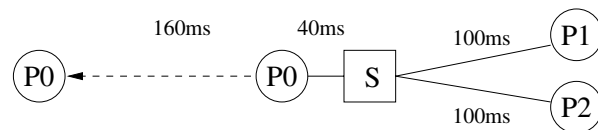


Fig. 4. Measured fairness and - Simulated Topology

### D. Moving a participant in the Playability Space

Trading inconsistencies, as explained in Section III-B, would be like moving in the playability space along a line with a negative slope (see Figure 3b) : increasing one type of inconsistency (say by 10 ms) to decrease another may improve the playability depending on the differences in the perceived impact of these.

It is always possible to move a user away from the origin (decreasing the user's playability) for example by artificially adding delay before sending update messages. One cannot however pull a player towards the origin at a constant angle without modifying the underlying topology and distribution of decision points.

## IV. FAIRNESS

### A. The source of unfairness

As discussed above, we believe that playability is an attribute that is influenced by the perceptual impact of different inconsistencies for a *given player* and stems from the position of that player in the playability space. We would like to propose that fairness is a notion concerned with *relative playability among* the players. Our view is that variations in playability between players, that is, their 'distances' from origin in the playability space, are the source of unfairness. In other words, for a game to be fair, all players should be clustered together in the playability space and their *spread* would be a suitable metric to measure fairness. The only caveat is to note that some positions in the playability space are *Similar* to each other. We define two points as 'similar' if one can be transformed to another by a combination of inconsistency trading without changing the topology of the decision points. It would be important, therefore, to use a norm for measuring this spread that is insensitive to such transformations. In our example, the taxicab (also called Manhattan) norm applied in the playability plane would return a distance of players from

the origin independent from their trade-off between response time and inter-decision point inconsistency.

Once a adequate norm is defined, the theoretical unfairness can be quantified by calculating the standard deviation of the players' distances to origin in the playability space.

A fair game is not necessarily playable. For example any game where all players are at the same distance (in terms of network delay) from a central server is perfectly fair. In general, it is easy to enforce fairness by artificially raising the level of inconsistencies of all players to the one worse off. However, if the worse off player is outside or at the edge of playable zone, the quality is degraded for everyone.

For an enjoyable game experience, a game should be fair and playable, ie all players should have similar locations in the playability space and all in the playable zone.

### B. Measuring fairness

The above definition of fairness was from a technical point of view that can be evaluated objectively by measuring the separation of players in the playability space which depends on the decision point topology and synchronisation. One of our aims is to investigate whether it is possible to relate this objective viewpoint to the experiential views of users. In other words, we are interested to demonstrate if the players' perception of fairness/unfairness can indeed be the result of their spread in the playability space. To this end, we need to identify what would constitute fairness from a user's perspective and see if these two viewpoints are related.

From a user's perspective, a game may be considered fair if it provides the same opportunity to all players in similar conditions. However, the evolution of the game state depends on many variables, and in some cases a player might just be lucky while the game is fair.

We have developed a technique to measure the fairness of a network topology in our simulator by neutralising any influencing variables other than the network topology and synchronisation parameters. It involves a scoring system and a method for cancelling the influence of initial conditions. Our method is not dissimilar to scoring techniques used in sport competitions like soccer. The main difference is that in sport competitions, the aim is to determine which team has the best skills given fair playing conditions while in our case it is the opposite: using synthetic players with the exact same skill level, we would like to evaluate the fairness of a game for a given network delay and decision point topology by looking at the outcome of specific combinations of the game.

In a soccer match, for example, each game is divided into two half sessions for the teams to switch their initial conditions (side and kick off). We also cancel the impact of initial conditions for an arbitrary number of players  $N$  by running all the different possible permutation of the initial conditions (that is,  $N!$  simulated games).

After each game session, avatars are ranked in the increasing order of their lifetime. The first player to have died gains 0 point, the second 1 point, the third 2 points and so on. Scores are added individually over all the permutations. To obtain

meaningful data multiple permutation groups, or matches, are simulated and averaged on the same network topology. To be fully comparable, players scores are normalised by the average number of points distributed. Therefore, if a player has an unfair advantage due to the topology, its normalised score will be above 1, and less than 1 if it is being handicapped. We define our experimental measure of fairness as the *standard deviation of the normalised player's scores*: the closer to 0, the fairer the game.

To validate the causal relationship between this experiential definition of fairness and the objective definition based on spread in the playability space, Figure 3c presents simulation results showing how the score of a player, along with the measured and theoretical fairness, vary when its distance from the central server increases. The simulated topology is shown in Figure 4. Players  $P1$  and  $P2$  delays are fixed at  $100ms$  propagation time from the central server while  $P0$  is gradually moved away from it. As predicted, there is a connection between the variation in playability (response time), the normalised scores, the theoretical and measured fairness in the game. When all players are equidistant from the server, the game is fair because the scores off all players are equal. The closer  $P0$  is to the server, the higher its relative score, the further away from it, the lower its score.

## V. SIMULATIONS

### A. Simulator and topology

In order to study the fairness of different architectures we simulated a simple network game on variations of a basic topology. Four participants are playing a game in three different countries connecting to three potential servers in each country as shown in Figure 4: two in the USA, one in Australia and the last one in the UK. Delays are rough estimation of the propagation time between the servers locations.

The number of players is chosen to be small to limit the number of necessary initial conditions permutations (hence simulation runs). The imbalance in distribution of players is on purpose to mimic the non-uniform distribution of players due to time zone or popularity differences in different countries.

For simplicity and readability of data, the playability space is collapsed onto a single dimension, generated by the response time, by trading all other inconsistencies for response time. To that aim, fully conservative local lag is used when using distributed servers.

The simulator used is an upgraded version of the Distributed Game Simulator (DGS) we presented in [9] running on different topologies. In the simulated games, avatars can either move or try to shoot another avatar in their visual range. All figures were generated from the average of 100 matches of 24 rounds necessary to permute the initial conditions of the four players. In total each topology has required 2400 simulations using an OpenMosix cluster of 48 Pentium4 machines.

### B. Discussion

Figure 6 shows the standard deviation of the players' normalised score for all the combinations of central and

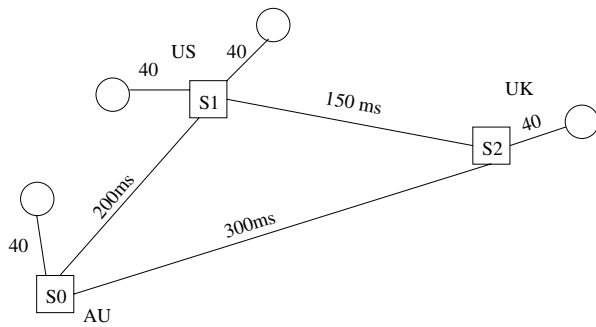


Fig. 5. Fairness Comparison: Topology

distributed servers possible. Players always connect to their closest activated server.

Topology labels have the following meanings:  $S_i$  means only the server  $S_i$  is active and all players connect to it.  $S_{eq}$  is the imaginary server which situated on the surface of the Earth and would provide equal propagation delay to all three locations: it is the ideally located central server for optimal fairness. In  $S_i S_j$  topology, servers  $S_i$  and  $S_j$  act as distributed servers.  $SS_{sequ}$  stands for the three server being distributed and all using the same local lag (300ms in this case).  $SS_{diff}$  shows the case when the three servers are being distributed but not forced to bear identical local lag: this configuration enables  $S_1$  in the US to apply only 200ms of lag to its players since no other decision point from it is further than this distance.

Not surprisingly, over these long distances, central server architectures perform very poorly in terms of fairness. The best average playability is found when the server is placed near the group of two players in  $S_1$ . It is the same configuration which offers the best, but still very bad, fairness of all possible central server sites (apart from the imaginary  $S_{eq}$ ). This is due to the fact that the  $S_1$  is also the server which is the closest to the two others.

When using two mirrored servers, fairness is greatly improved since two out of the three sites have the same amount of consistency (ie the same response time in our simplified case). Again, the topologies providing better response time and fairness are the ones which include  $S_1$  as one of the servers for the same reasons as the central servers case.

Finally the fully distributed architectures are the ones performing the best in terms of fairness. The "optimal local lag" version, where each server only applies the amount of local lag required to maintain perfect consistency, provides a better average response time because the server  $S_1$  doesn't require as much local lag as the two other servers thanks to its more central location with respect to the other decision points. The "equal local lag" version, forcing all replicated servers to use a 300ms local lag would result in perfect fairness as expected. It is however, very interesting to compare it to the imaginary  $S_{eq}$  central server: while both ensure the perfect fairness, the distributed architecture still provides a better response time. This is an important point because it shows that in some topologies a set of distributed decision points will always outperform a central decision point, whatever its location.

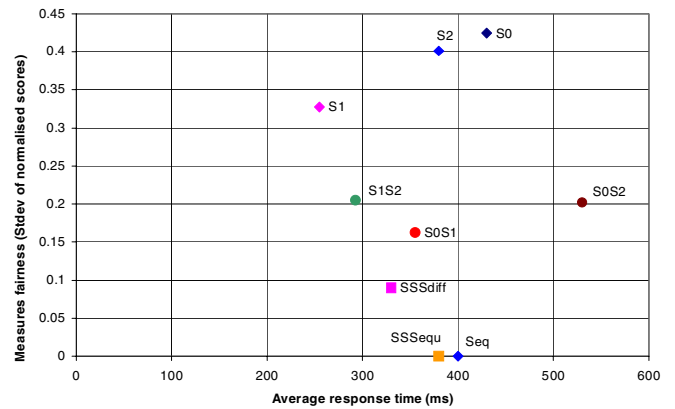


Fig. 6. Fairness Comparison: Fairness vs Average Response Time

## VI. CONCLUSION

This paper studied the relationship between propagation delay, inconsistencies, playability and fairness in online multiplayer games through analysis of the different architectures. The introduction of the playability space is used to clarify how trading inconsistency can improve the player's comfort. The playability space also helped to arrive at a theoretical definition for fairness. We also proposed an experimental procedure to measure fairness from the user's perspective in simulations. Simulations show consistent results with our theoretical predictions. Finally the study of a few topology variation shows fairness and playability can be improved using server distribution.

## VII. ACKNOWLEDGEMENTS

This work was partially funded by the CRC for Smart Internet Technology, Australia.

## REFERENCES

- [1] F. Safaei, P. Boustead, C. D. Nguyen, J. Brun, and M. Dowlatshahi, "Latency-driven distribution: infrastructure needs of participatory entertainment applications," *IEEE Communications Magazine*, vol. 43, pp. 106–112, May 2005.
- [2] C. Palazzi, S. Ferretti, S. Cacciaguerra, and M. Rocchetti, "On maintaining interactivity in event delivery synchronization for mirrored game architectures," in *IEEE GlobeCom*, Dec 2004, pp. 157–165.
- [3] Y.-J. Lin, K. Guo, and S. Paul, "Sync-ms: synchronized messaging service for real-time multi-player distributed games," in *IEEE ICNP*, Nov 2002, pp. 155–164.
- [4] N. Baughman and B. Levine, "Cheat-proof payout for centralized and distributed online games," in *IEEE InfoCom*, Apr 2002, pp. 104–113.
- [5] C. GauthierDickey, D. Zappala, V. Lo, and J. Marr, "Low latency and cheat-proof event ordering for peer-to-peer games," in *ACM NOSDAV*, 2004, pp. 134–139.
- [6] B. D. Chen and M. Maheswaran, "A fair synchronization protocol with cheat proofing for decentralized online multiplayer games," in *IEEE NCA*, 2004, pp. 372–375.
- [7] W. Bernier, "Latency compensating methods in client/server in-game protocol design and optimization," in *Game Developer Conference (GDC)*, Mar 2001.
- [8] M. Mauve, J. Vogel, V. Hilt, and W. Effelsberg, "Local-lag and timewarp: Providing consistency for replicated continuous applications," *IEEE Transactions on Multimedia*, vol. 6, pp. 47–57, Feb 2004.
- [9] J. Brun, F. Safaei, and P. Boustead, "Distributing network games servers for improved geographical scalability," *Telecommunication Journal of Australia (TJA)*, vol. 55, pp. 23–32, 2005.
- [10] P. Quax, P. Monsieurs, L. Wim, D. De Vleeschauwer, and N. Degrande, "User experience: Objective and subjective evaluation of the influence of small amounts of delay and jitter on a recent first person shooter game," in *ACM Netgame*, Aug 2004.