Implementation and Evaluation of a Multiplayer Pong Game

Marco Begolo
University of Padua,
Department of Mathematics
Via Trieste 63, 35123 Padua,
Italy
marco.begolo@studenti.unipd.it

Sebastiano Valle
University of Padua,
Department of Mathematics
Via Trieste 63, 35123 Padua,
Italy
sabastiano.valle@studenti.unipd.it

Marco Zanella
University of Padua,
Department of Mathematics
Via Trieste 63, 35123 Padua,
Italy
marco.zanella@studenti.unipd.it

Armir Bujari
University of Padua,
Department of Mathematics
Via Trieste 63, 35123 Padua,
Italy
abujari@math.unipd.it

Ombretta Gaggi
University of Padua,
Department of Mathematics
Via Trieste 63, 35123 Padua,
Italy
gaggi@math.unipd.it

Claudio E. Palazzi
University of Padua,
Department of Mathematics
Via Trieste 63, 35123 Padua,
Italy
cpalazzi@math.unipd.it

ABSTRACT
Mobile gaming is a new frontier of the gaming industry with an ever increasing market share. Augmenting processing capabilities coupled with mobile sensing technology are some key ingredients contributing to a steady growth of innovative gaming solutions. In these settings, players could interact with each other in the context they are immersed in and exploit proximity, local information through wireless communication(s). Pursuing this goal, we discuss Multipong, a multiplayer version of the classic Pong game whereby players, in addition to the infrastructure mode, are offered the possibility to interact through ad-hoc communications. Along with the game description, we present a user-study involving 168 subjects undertaken at the University of Padua coupled with energy measurements of the proposed solution.

ACM Classification Keywords
K.8. Personal Computing: Games

Author Keywords
Games, Multiplayer, Wi-Fi Direct, Proximity.

INTRODUCTION
Mobile gaming is gaining momentum in the gaming industry. The ease of use, the social dimension it embodies coupled with the vast penetration of mobile hand-held devices have proven to be a gold mine for small and medium enterprises (SMEs) approaching the gaming market. It is a belief that this trend will continue in the future and mobile gaming will become a forefront which will include players from console and PC vendors too [1].

The ever increasing graphic processing power of mobile devices coupled with advances in sensing and communication technology has brought many innovative tools at the developers disposal, lowering the barrier of entry into the gaming market [2]. These features provide the building blocks for an innovative gaming experience departing from traditional gaming consoles [3].

Nevertheless, mobile game development is challenged by the resource-constrained nature of the targeted appliance [4, 5, 6]. Battery is a major issue demanding for intelligent duty cycling techniques now considered even at the application layer. Moreover, mobile cellular communication(s) are cost-attributed, posing a burden on the player side; intelligent networking techniques making a parsimonious use of data exchange are needed. Yet, the input mechanism mainly relies on the touch sensor (soft keypad) which, if not properly considered, might hinder the gaming experience [7, 8].

Among the various categories, multiplayer mobile games represent the top grossing category where many of them require real-time user interaction involving Internet access. In this paper, we present a study of Multipong, a multiplayer ad-hoc version of the classic arcade game Pong. A preliminary version of the game was previously presented in [9]. In this context, we present a complementary analysis concerning energy consumptions considerations of our adopted solution along with a user-study involving 168 participants from the University of Padua, Italy.

The remainder of the paper is organized as follows: in Section “Background” we provide a brief background on ad-hoc networking support in the Android platform. Related works are discussed in Section “Related Work”. Section “Game Description” describes the game and some technical details regarding the development process, outlining key features of the network-

GHITALY17: 1st Workshop on Games-Human Interaction, April 18th, 2017, Cagliari, Italy.

ing and gaming module. The experimental testbed and the outcome are discussed in sections “User Study” and “Energy Consumption”. Finally, we conclude in Section Conclusion.

BACKGROUND

The Wi-Fi Direct standard, also known as Wi-Fi P2P, enables devices to connect with each other without requiring the presence a physical wireless access point [10]. Wi-Fi P2P implements a software access point module, capable of host configuration and management. In Wi-Fi Direct terminology a network unit is referred to as a *group* and each group has a *group owner (GO)* whose role is analogous to the one of an access point in infrastructure-mode.

Usually, a device supporting Wi-Fi Direct, in order to create or join a group, starts a discovery session in which it may find other unconnected Wi-Fi Direct devices or GOs. A device can autonomously decide to start the formation of a group, or may ask to join one. During group formation, devices need to negotiate their roles in order to find a peer that assumes the role of a *logical* access point. While the GO negotiation protocol is specified by the standard, applications can implement their own logic of electing a suitable one. Legacy devices on the other side, those that do not support Wi-Fi Direct, may later on decide to connect to the GO and join the group.

More in detail, the standard outlines three different group formation techniques, namely standard, persistent and autonomous. The *standard* technique is the most generic group formation technique while the others shortcut some of the phases involved. The procedure starts with nodes first becoming aware of each other either by passive or active scanning of Wi-Fi channels. Once this phase is completed, the GO negotiation phase takes place, where each device states its own `GroupOwnerIntent`, consisting of a value ranging from 0 (not willing to become the GO) to 15 (highest inclination to become a Group Owner). Successively to the GO negotiation phase, the security and address configuration phases take place in sequence and, if successful, the group is considered as established and nodes can communicate without any infrastructure mediation.

Support for Wi-Fi Direct in Android devices has been rolled out since Android 4.0, enabling P2P connectivity amongst Wi-Fi Direct capable devices. In these settings, a GO is connected to multiple clients in a P2P fashion (hereafter *NGO*s). As discussed, the GO is decided after a negotiation phase between the devices; thus, the same hosts may create an ad-hoc network with different GOs from time to time.

However, the implementation of Wi-Fi Direct in Android presents some issues and limitations: first of all Android does not have native support for multi-group formation and devices must ask the user for the permission to join a group, hindering the automatic creation of Wi-Fi Direct networks [11].

RELATED WORK

According to [12, 13], main requirements for a gaming session are: (i) *good interactivity*, i.e. the delay between the user interaction and the game response should be as short as possible, (ii) *consistency*, i.e. different players should see coherent and admissible game states, (iii) *fairness*, i.e. it should be possible to win a match regardless of different network conditions, (iv) *scalability*, i.e. being able to support a large number of players, and (v) *continuity*, i.e. the present game session should not be interrupted because of disconnections, handoffs, or any other mobility-related issue. Fulfilling these requirements, a lot of research effort has been devoted, spanning from architectural solutions to efficient network-layer proposals [14, 15].

Mobile gaming further exacerbates the issues, also presenting its own challenges in the context of real-time applications: e.g., multiplayer gaming requires Internet connectivity which in the mobile world might be cost-attributed or at least not available anytime, anywhere [16]. As a remedy, one might resort to local gaming sessions whereby a coordinator node hosts the session becoming a potential bottleneck [17]. On the other side, pure P2P or hybrid solutions represent an attractive alternative but usually lack of protection from cheaters [18, 19].

Multipong belongs to the category of *casual games*, i.e, video games which present a simple gameplay and targets a mass audiences [20]. Casual games are designed to be played by users with no special skills and without requiring too much time for both understanding and playing them [21]. A well known example of casual game is the Candy Crush Saga.

Despite being very common among mobile games, casual games were originally played by users through a web browser and a large number of users still play using the web platform, e.g., through social networks: the idea of casual gaming has been indeed mashed up with this recent phenomenon, allowing casual gamers to play with their friends through different platforms and network architectures [22, 23, 24].

Casual games experiments were also undertaken in [25, 26] but, as these studies reported, this type of games has not break through either the academic or the commercial world yet.

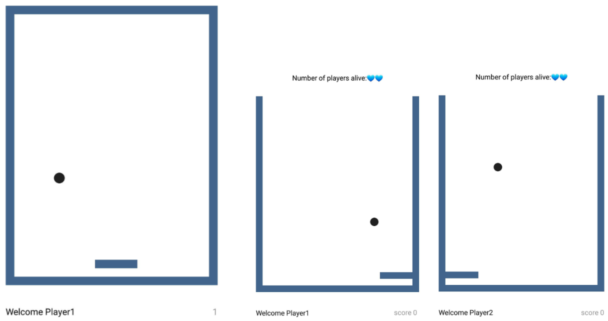
GAME DESCRIPTION

Multipong is a multiplayer version of the *Pong* game, one of the first arcade videogames, where the players need to prevent a ball from falling out of the screen with a paddle.

Multipong allows several players to connect their devices forming an ad-hoc network. The device who created the gaming session has to make the first move and when this happens, the ball is transferred to the next player’s screen as if their gameboards were joint (see Figure 1(b)). When a player misses the ball, the player loses a life and the ball is thrown out randomly to the next player’s screen. If a player runs out of lives, the player will not be able to play for the rest of the game and the game takes place between the remaining players. The last standing player is the winner of the gaming session.

The game also implements the single-player mode which is essentially the old Pong game (see Figure 1(a)) and, for this reason, will not be discussed further in this paper.

Multipong is based on a two layer architecture as depicted in Figure 2. The *networking-layer* handles group formation and communication among peers and the *game-layer* handles the



(a) (b)

Figure 1. A single-player session (left) and a multiplayer session with a shared, open board between the players (right)

game application logic. This loose coupling between layers allows the reuse of the network-layer for future potential scenarios e.g., in the context of crow-sourced video annotation and of geo-localized partecipatory sensing [27, 28, 29]. More in details, the `Discovery` component of the *network-layer* is responsible for understanding which is the group owner (GO) device and for retrieving the IP addresses of the other peers. The `NameResolution` component binds the IP addresses to logical, application-level identifiers, and the `Communication` component manages data exchanging amongst devices.

On the other side, the `GameLogic` component of the game layer is accountable for the application logic while the `GameView` component displays information to the user and manages player interactions. The two layers talk to each other through the `NetworkingInterface`, which is the component that provides a mid-level abstraction.

During the game, the participating devices are connected into a Wi-Fi Direct ad-hoc network. Since in the initialization (game formation) phase there are no strict requirements to meet in terms of real-time information delivery, we decided to use TCP as a transport protocol. Raw data sent over a socket are formatted as JSON objects, giving us the ability to distinguish more easily the requests from one peer to another.

Since any player can start a gaming session and all players' devices are capable of hosting a game, we can not *a priori* assume that the host coincides with the group owner (GO).

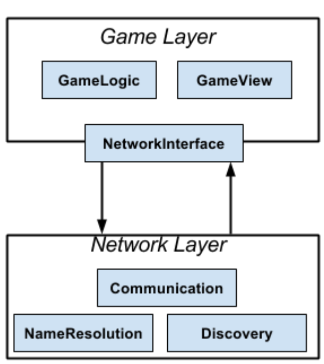


Figure 2. Multipong architecture

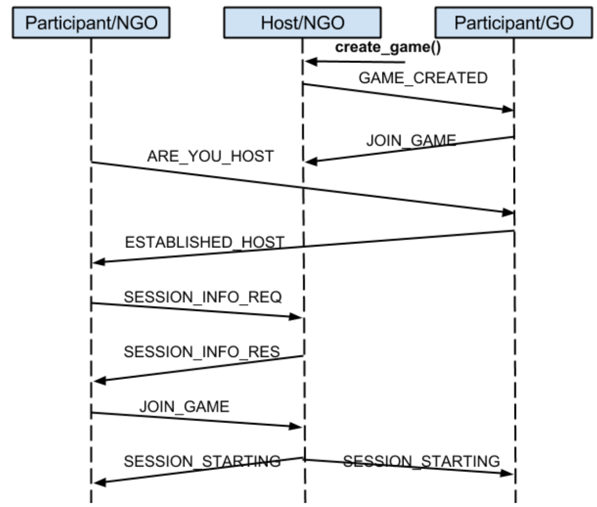


Figure 3. Message exchanged during game initialization phase

Moreover, since the Wi-Fi Direct protocol attributes to all NGOs an IP address and this information is not passed or stored on the upper layers of the GO, we have built a protocol enabling the GO to retrieve the IPs addresses attributed to other NGOs devices. A general scenario of the communication protocol and the steps involved are shown by means of a sequence diagram depicted in Figure 3. This represents a general case scenario whereby the host and GO device are separate entities and the players need to autonomously identify the host device after which each player contacts it to acquire additional information regarding the gaming session. These data comprise information such as the number of players that currently have joined the game session along with an application layer identifier used to denote the players turn. Also, all communication is unicast and performed in an asynchronous fashion, so that messages can be exchanged without blocking the device while waiting for a response and we guarantee a FIFO ordering of the data sent out of a device. Additional details regarding the protocol can be found in [9].

During the gameplay we want to ensure both consistency and low latency of gaming events, so messages have to be delivered very quickly. Therefore we employ the UDP protocol, in particular application-level acked UDP transmission in order to improve reliability: after sending an UDP packet, a peer waits for a short ACK packet within a short time frame; if it does not, the peer retries the transmission up to a number of times which is set to four but can be changed.

Moreover, the two layers of the architecture have different roles: one layer is concerned with coordinating the peers and the other one deals with the multiplayer game logic on top of the coordination layer. The multiplayer game layer has to manage the local state of the game and contribute to manage the global state. Moreover, when the player hits the ball with the paddle, it has to compute the ball exit point from the screen and send this information to the GO as soon as it is available to spread this message to all the active participants. We compute and send the ball exit position in advance to reduce the network latency perceived by the player.

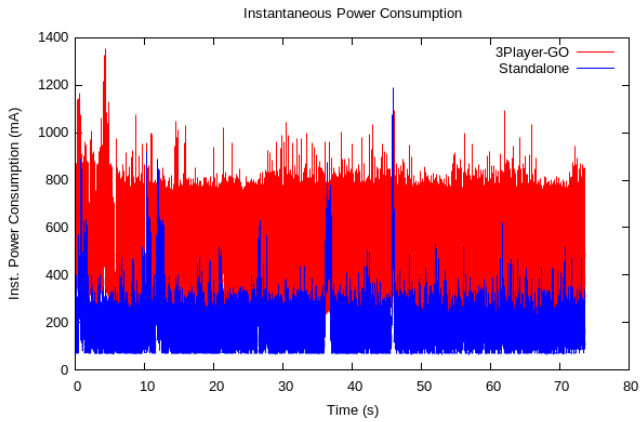


Figure 4. Instantaneous power consumption

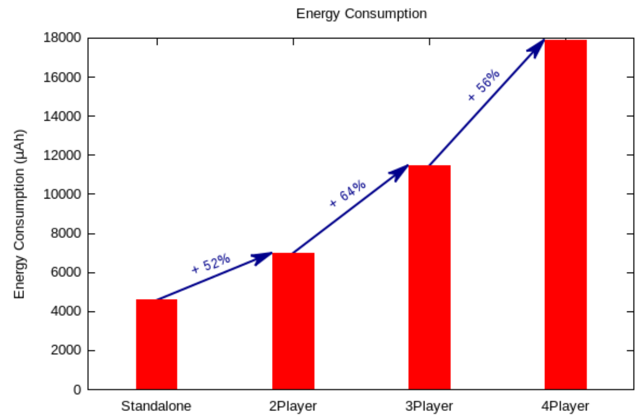


Figure 5. Power consumption of different gaming modes with a resulting standard deviation of ±11, ±31, ±35, ±30 µAh respectively.

The GO device acts as a coordinator of the gaming session, even if the game is hosted elsewhere. Hence, the GO represents a single point of failure during the game phase, and a network failure such as the crash or sudden disconnection of the coordinator would make the whole network collapse, and therefore end the game. A NGO failure, instead, causes only the end of the game for that particular player, and the game continues with the GO telling the other players that a that peer is not active anymore.

ENERGY CONSUMPTION

In this section we discuss the measurements regarding energy consumption of our proposed solution. To this end, we first introduce the testbed we employed and how the measurements were performed and, thereafter we discuss the outcome.

Measurement Testbed

The test devices employed for the measurements are four Android Galaxy Samsung S5 devices and all devices have identical settings, updated to Android version 6.0.1. To measure the amount of energy consumed by the application we chose to rely on an external hardware tool named Monsoon PowerMonitor. This components main function is to measure the energy requested by the smartphone (or other devices that use a single lithium battery) and it is the sole power sources for the device. We refer the reader to [30, 31] for more information regarding the usage of the measurement hardware.

In order to perform measurements not influenced by the user interaction, we coded an autonomous version of the Multipong game, whereby an artificial player plays the game for the 10 consecutive rounds. After those rounds are completed, the last standing player, corresponding to the last player joining the game, wins. In a multiplayer game session the GO device acts also as a host of the gaming session, hence the GO has the additional burden of coordination among the devices at the application layer. It is also noteworthy to point out, that a GO device acts as a layer 3 router in the network as by Wi-Fi Direct specifications. Therefore a GO device is a potential bottleneck and represents the major interest from an energy consumption viewpoint when compared to other devices.

Results

Figure 4 shows a comparison of the instantaneous power consumption of a standalone and a multiplayer game session. The multiplayer game session lasts longer when compared to the singleplayer one, showing regular behaviours at each round. In the standalone mode the instantaneous power consumed is lower when compared to the multiplayer one and this is to be attributed mainly to the absence of wireless communication. In fact, in the multiplayer version of the game, the Pong ball goes from one screen to another at quite regular intervals, and to these events are associated with communicatio costs. The number of transmissions increases with an increase in the number of players.

We must note here that in order to perform an accurate measurement of the power consumption, the smartphone should be usually put in "Airplane mode" [31] so as to avoid interference by external events or by normal operating system functionalities. However, these settings are not feasible in this case since the devices must communicate each other using the wireless inteface. In Figure 4, some of the spikes in power consumption correspond to these kind of events.

Another crucial piece of information is the amount of energy consumed by the GO device in every scenario. To compute it, for each game configuration we perform 5 runs in order to gain more confidence in the obtained results. The number of runs was sufficient, lowering the standard deviation of the measurements which are reported in Figure 5. When communication is employed, energy consumption is higher, increasing linearly with an increase in the number of players. We also report that for each gameply the average expected battery lifetime measured by the Monsoon tool is 6h, 3.13h and 2.5h for the standalone, 2 and 3 player respectively.

USER STUDY

In order to test the level of acceptability of the Multipong game among users, we asked to a set of users to answer an anonymous questionnaire. The questionnaire contains 13, 5-point Likert questions with possible answers ranging from "Strongly disagree" to "Strongly agree". 168 users completed the questionnaire, 23 females and 145 males, only 2,4% of

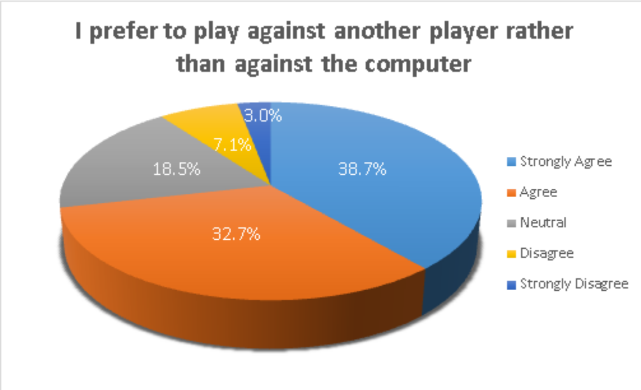


Figure 6. Answers to the question: "I prefer to play against another player rather than against the computer"

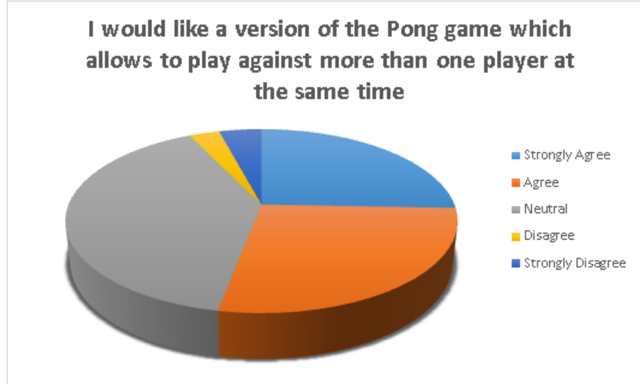


Figure 7. Answers to the question: "I would like a version of the Pong game which allows to play against more than one player at the same time"

them was over 30 years old, the rest was under 30. 96% of the participants were students (96% from the Bachelor and Master degrees in Computer Science at the University of Padua, Italy). No participants were involved in the development process, and they did not have any knowledge about the project before the questionnaire, but 54% already knew the traditional Pong game.

We asked the users a set of questions to understand their preferences about games. Most of the participants usually play with videogames, and 31% defined themselves as *frequent players*, 37% plays with videogames every days, 24% weekly, 25% plays sometimes and only 13% seldom plays with videogames. The preferred gaming platform are online video games (67 participants declared to use online videogames, 35 to use console ones like Play Station, Wii or Xbox).

According to the given answers, Multipong collects some features that users usually enjoy: 71.4% of the users prefer to play against another user instead of a computer simulated player (see Figure 6), 81 users like to play videogames which involve other players in the same room and 53% of the users would like a new version of the Pong game which allows to challenge more than one player at the same time, which is exactly what Multipong is. Even if 53% may look like a very low percentage, actually, only 8% of the participants declared to do not like this version of the game: as depicted in Figure 7 the rest declared themselves as neutral. Therefore, we can say that the level of acceptability for this game is high, i. e., the game like to the participants.

After explaining the implemented version of Multipong, 56% the participants defined as one of the most important features of the game the possibility to challenge more players which are in the same room, and only 30% of the them considered the requirement that all the players must be in the same room a strong constraint.

CONCLUSION

Mobile gaming has seen an increase in popularity and this trend is expected to grow in the future. Sensing and communication capabilities along with a rich set of development libraries provide the basic building blocks for an innovative gaming experience. In this article, we discussed *Multipong*, a multiplayer ad-hoc version of the old arcade game Pong, whereby players could engage in a gaming session when in proximity with each other. A user study, undertaken at the University of Padua involving 168 subjects, showed that Multipong collects some features that users usually enjoy. Along with the user study, we presented realistic measurements of energy consumption showing that it increases linearly with the number of participating players, exhibiting no strange behavior.

As an extension to the current solution we plan to address the network reformation process whereby peer devices are able to autonomously and transparently recover from the loss of a GO device. Also, we plan to exploit multicast/broadcast capabilities addressing the overhead of employing solely unicast communications among devices. To this end, we plan to exploit the feasibility of multicast/broadcast communications between nearby devices relying on the GO's device routing capabilities only when necessary.

Acknowledgment

This work has been partially funded by the University of Padua, through the projects PRAT CPDA137314 and PRAT CPDA151221.

REFERENCES

1. Newzoo Games, The Global Games Market Share. April 2016.

2. Marco Furini, "Mobile Games: What to Expect in the Near Future", in Proc. of *GAMEON*, 2007.

3. Catia Prandi, Valentina Nisi, Paola Salomoni, and Nuno J. Nunes, "From Gamification to Pervasive Game in Mapping Urban Accessibility", in Proc. of *ACM Italian SIGCHI Chapter*, 2015, pp. 126-129.

4. Matteo Ciman, Ombretta Gaggi, and Nicola Gonzo, "Cross-platform mobile development: A Study on Apps with Animations", in Proc. of *ACM SAC*, 2014.

5. Matthias Dick, Oliver Wellnitz, and Lars Wolf, "Analysis of Factors affecting Players' Performance and Perception in Multiplayer Games", in Proc. of *ACM SIGCOMM*

Workshop on Network and System Support for Games, 2005, pp. 1-7.

6. Sebastian Möller, Steven Schmidt, and Justus Beyer, "Gaming Taxonomy: an Overview of Concepts And Evaluation Methods for Computer Gaming QoE", in Proc. of *IEEE QoMEX Workshop*, 2013, pp. 236-241.
7. Kimberly Chu, and Chui Y. Wong, "Mobile Input Devices for Gaming Experience", in Proc. of *IEEE i-USEr*, Nov. 2011, pp. 83-88.
8. Stefano Cacciaguerra, Silvia Mirri, Paola Salomoni, and Marco Pracucci, "Wandering About the City, Multi-Playing a Game", in Proc. of *IEEE CCNC*, 2006, pp. 1214-1218.
9. Marco Begolo, Sebastiano Valle, Marco Zanella, Armir Bujari, Ombretta Gaggi, and Claudio E. Palazzi, "Multipong: a Multiplayer Ad-Hoc Version of Pong", in Proc. of *DENVECT*, 2017
10. Wi-Fi Alliance®, *Wi-Fi Peer-to-Peer (P2P) Technical Specification v1.7*. July 2016.
11. Claudio Casetti, Carla F. Chiasserini, Luciano C. Pelle, Carolina D. Valle, Yufeng Duan, and Paolo Giaccone, "Content-centric Routing in Wi-Fi Direct Multi-group Networks", in Proc. of *IEEE WoWMoM*, 2015, pp. 1-9.
12. Arnaud Kaiser, Dario Maggiorini, Nadjib Achir, and Khaled Boussetta, "On the Objective Evaluation of Real-Time Networked Games", in Proc. of *IEEE GLOBECOM*, 2009, pp. 1-5.
13. Claudio E. Palazzi, "Interactive Mobile Gaming over Heterogeneous Networks", in Proc. of *IEEE/ITI International Conference on Information and Communications Technology*, Cairo, Egypt, Dec. 2007.
14. Lothar Pantel, and Laars C. Wolf, "On the Impact of Delay on Real-time Multiplayer Games", in Proc. of *ACM International Workshop on Network and Operating Systems Support for Digital Audio and Video*, May 2002, pp. 23-29.
15. Armir Bujari, Michele Massaro, and Claudio E. Palazzi, "Vegas over Access Point: Making Room for Thin Client Game Systems in a Wireless Home", *IEEE Transactions on Circuits and Systems for Video Technology*, 25(12), 2015.
16. Claudio E. Palazzi, Armir Bujari, Gustavo Marfia, and Marco Roccetti, "An Overview of Opportunistic Ad hoc Communication in Urban Scenarios", in Proc. of *IFIP MedHocNet*, 2014.
17. Claudio E. Palazzi, Stefano Ferretti, Stefano Cacciaguerra, and Marco Roccetti, "On Maintaining Interactivity in Event Delivery Synchronization for Mirrored Game Architectures", in Proc. of *GLOBECOM Workshops*, 2004, pp. 157-165.
18. Tobias Fritsch, Hartmut Ritter, and Jochen Schiller, "CAN Mobile Gaming be Improved?", in Proc. of *ACM SIGCOMM Workshop on Network and System Support for Games*, 2006.
19. Tobias Fritsch, Hartmut Ritter, and Jochen Schiller, "User Case Study and Network Evolution in the Mobile Phone Sector (A Study on Current Mobile Phone Applications)", in Proc. of *ACM SIGCHI ACE*, 2006.
20. Marco Furini, "An Architecture to Easily Produce Adventure and Movie Games for the Mobile Scenario", *ACM Computers in Entertainment*, 6(2), 2008.
21. Andrea Grimes, Vashudara Kantroo, and Rebecca E. Grinter, "Let's play!: Mobile Health Games for Adults", in Proc. of *ACM UbiComp*, Sep. 2010, pp. 241-250.
22. Dario Maggiorini, Christian Quadri, and Laura A. Ripamonti, "On the Feasibility of Opportunistic Collaborative Mixed Reality Games in a Real Urban Scenario", in Proc. of *ICCCN*, 2012.
23. Dario Maggiorini, Antonio Nigro, Laura A. Ripamonti, and Marco Trubian, "Loot Distribution in Massive Online Games: Foreseeing impacts on the players base", in Proc. of *ICCCN*, 2012.
24. Mario Gerla, Dario Maggiorini, Claudio E. Palazzi, and Armir Bujari, "A Survey on Interactive Games Over Mobile Networks", *Wireless Communications and Mobile Computing*, 13(3), 2013.
25. Janne Paavilainen, Annakaisa Kultima, Jussi Kuittinen, Frans Mäyrä, Hannamari Saarenpää, and Johannes Niemelä, GameSpace: Methods and Evaluation for Casual Mobile Multiplayer Games. Available at: https://tampub.uta.fi/handle/10024/65773, 2009.
26. Kevin A. Li, and Scott Counts, "Exploring Social Interactions and Attributes of Casual Multiplayer Mobile Gaming", in Proc. of *ACM Mobility*, 2007, pp. 696-703.
27. Catia Prandi, Stefano Ferretti, Silvia Mirri, and Paola Salomoni, "Trustworthiness in Crowd-sensed and Sourced Georeferenced Data", in Proc. of *IEEE PerCom Workshops*, 2015, pp. 402-407.
28. Stefano Ferretti and Silvia Mirri and Marco Roccetti, and Paola Salomoni, "Notes for a Collaboration: On the Design of a Wiki-type Educational Video Lecture Annotation System", in Proc. of *IEEE ICSC*, 2007.
29. Claudio E. Palazzi, Lorenzo Teodori, and Marco Roccetti, "Path 2.0: A Participatory System for the Generation of Accessible Routes", in Proc. of *IEEE ICME*, 2010.
30. Matteo Ciman, and Ombretta Gaggi, "Evaluating the Impact of Cross-platform Frameworks in Energy Consumption of Mobile Applications", in Proc. of *WEBIST*, 2014, pp. 423-431
31. Matteo Ciman, and Ombretta Gaggi, "An Empirical Analysis of Energy Consumption of Cross-platform Frameworks for Mobile Development", Pervasive and Mobile Computing, Volume 39, 2017, pp. 214-230