Massivizing Multi-Player Online Games on Clouds

Siqi Shen
Parallel an Distributed Systems Group
Delft University of Technology
Delft, Netherlands
S.Shen@tudelft.nl

Alexandru Iosup
Parallel an Distributed Systems Group
Delft University of Technology
Delft, Netherlands
A.Iosup@tudelft.nl

Dick Epema
Parallel an Distributed Systems Group
Delft University of Technology
Delft, Netherlands
D.H.J.Epema@tudelft.nl

Abstract—**Massively Multiplayer Online Games (MMOGs) are an important type of distributed applications and have millions of users. Traditionally, MMOGs are hosted on dedicated clusters, distributed globally. With the advent of cloud computing, MMOGs such as Zynga's are increasingly run on cloud resources, through the use of cloud technology and innovation. Massivizing MMOGs on clouds is the focus of my PhD research. My main contributions are: 1) analyzing and modeling various MMOG workloads, including those of social and traditional real-time games, 2) designing and implementing a cost-efficient and reliable cloud-based MMOG platform, 3) designing and implementing a scalable MMOG system which employs domain-specific scaling techniques to support the real-time strategy games of the future, 4) experimental prototypes and tools to evaluate our proposed research via real-world experimentation and simulation, and applying our proposed research to a popular real-world application. In this article, I introduce my research progress and my future plans.**

I. Introduction

Massively Multiplayer Online Games (MMOGs) are a new type of large-scale distributed application. For example, the social game FarmVille and similar games attract monthly over one hundred million users. Since the early 2000s, MMOGs are traditional HPC users; for example, World of Warcraft (WoW) deploys globally hundreds of thousands of cores in 9 datacenters. Invented by industry, MMOGs have recently attracted the interest of distributed computing researchers [1]–[6] as well. With the advent of cloud computing, there is a trend of MMOG operators using cloud resources as their infrastructure, to serve their customers due to the elastic, flexible, and pay-as-you-go features of clouds. For example, Zynga is entertaining players with the help of both public and private clouds. MMOG operators need to design scalable, reliable, and possibly cloud-based MMOG architectures to ensure the users' growing quality-of-service (QoS) expectations. A good gaming experience is needed to keep gamers playing, and has an important impact on the income of the industry. Such cloud-based MMOG architectures constitute the focus of my PhD thesis.

Challenges of MMOGs such as scalability, heterogeneity, security, consistency, reliability, and QoS have been identified by distributed computing community [5], including CCGrid community [6]. As a type of application that runs in cluster, cloud, and grid environments, these MMOGs challenges fit well the interest of the CCGrid community.

Understanding the workload of different games is important for the design and tuning of efficient systems to support them. Real-time MMOGs, such as WoW, and online social gaming, such as Facebook's and Zynga's games, are two types of MMOGs my thesis focuses on. Different genres of games have different type of QoS requirements [1]. Thus, understanding the workload of MMOGs is the *first challenge* we are facing.

To serve millions of users world-wide, MMOG operators need to use massive amounts of machines distributed globally. To maximize the profit of MMOG operators yet deliver the QoS required by players, resource provisioning and allocation algorithms that are fast, efficient in both cost and performance, and can withstand flashcrowds of user activity, are needed. Moreover, as cloud computing resources may not be reliable, it is necessary to develop fault-tolerant techniques to ensure QoS. The *second challenge* we are facing is to develop a cost-efficient and reliable cloud platform to support MMOGs.

The industry adopts the client/server model as the main MMOG architecture: the game server simulates a world, and receives and processes commands from clients. Based on client input, the game server computes a global state and transfers in-game state such as objects' positions back to the clients. Although the industry is serving millions of players, the number of players that can interact with each other is limited by the capacity of a single server [5] which supports less than one hundred moving objects in close encounters. The scalability limitation of MMOG architecture is the *third challenge* we are facing.

Although many methods and architectures have been proposed to overcome the challenges of designing MMOG systems, only few of them have been evaluated comprehensively, especially through real-world experimentation. The *fourth challenge* we are facing is to show that our proposed approaches work in practice.

To summarize, my PhD thesis focus on massivizing multiplayer online games using cloud computing technology. Addressing the above challenges, I plan to answer the following research questions:

1) What is the workload of MMOGs?
2) What is a cost-efficient and reliable cloud-based platform that is suitable for MMOGs?
3) What mechanisms can make MMOG architectures scalable on clouds?
4) How to prove that the proposed research results are valid in practice?

The rest of this paper is structured as follows. Section II describes our current and ongoing research, methodology, and results. Section III presents the remaining time line of my PhD research. Section IV summaries my research.

II. PROPOSED RESEARCH

This section describes the four main components of my PhD thesis, in turn: analyzing and modeling MMOG workloads, designing a cloud-based MMOG platform, designing domain-specific method for a scalable MMOG system, and experimental research using real application and clouds. For each topic, we explain our main research results and emphasize the novelty of our contributions.

A. Workload of MMOGs systems

Analyzing the workload of MMOGs systems helps system designers to understand user behavior, and to gain insight about system operation, and thus to design better systems. The novel workload workload aspects we focus on are: online meta-gaming networks, players' mobility behavior, and match-based games.

1) MetaGaming network: Online Meta-Gaming Networks are Internet-based communities of (computer) game players that extend in-game functionality by focusing on the relationship between game sessions, on what happens in the meantime between game sessions, and on the relationship between games. Motivated by the popularity of these game-related networks and by the growth of their importancełmany major game developers have started to operate their own online meta-gaming network. We have analyzed [7] a long-term observation of XFire. Using long-term, large-scale data that we have collected, we have presented a high-level, marginal distribution- and time-based analysis of XFire: its global network, player activity, user-generated content, and social structure.

Ours is one of the first characterizations of a MetaGaming network. We have found that XFire is a slowly growing network whose players spend collectively over in-game 100 years, every hour. We have quantified the hardcore-ness of XFire players, and found that a significant fraction of them have played over 10,000 in-game hours each. We have also found that XFire community members are routinely engaged in the creation and consumption of game-related media, such as screenshots and videos.

2) Mobility model: The performance of distributed systems that support common human activities, such as mobile communication and participation in online games, depends on the characteristics of human mobility. Human mobility models are crucial in the design and tuning of these systems, but depend on detailed and long-term human mobility traces, which, in turn, are difficult to obtain from real environments. Thus, although many human mobility models have already been developed, few have been validated against adequate traces. In contrast, we have investigated the use of human mobility traces collected from realistic, humanlike virtual worlds for the purpose of understanding human mobility and developing new mobility models [8].

We have collected traces for over 30,000 virtual citizens of a popular virtual world, and compared these traces with other real- and virtual-world traces. Our collected traces are more precise and accurate than public datasets, thus reveal more evidences of mobility of human in virtual world. Our analysis reveals that the mobility traces of humans in virtual and real worlds have many similar characteristics. We have further proposed SAMOVAR, a novel model for human mobility in virtual and real worlds that takes into account mobility characteristics such as the population-wide and individual area popularity. Based on synthetic traces generated from our and three existing models, we have conduced a simulation-based analysis of the impact of human mobility on the performance of virtual and real networked environments. We have found that SAMOVAR leads to useful insights, and discriminates better than the other models among the performance of different networked environments.

3) Match-based game: Online match-based games, such as the online versions of the board game of chess, have already captured a global audience of tens of millions of players. Through a unique combination of characteristics, a short duration (coffee-break minutes), weak correlation between matches, and clear emphasis on winning, match-based games may serve a unique segment of the global player population.

We have collected five online match-based datasets [9] [10] and published two of these datasets in Game Trace Archive [11]. We have analyzed the number of matches per player, the inter-arrival time and duration of matches, geographical distribution and win-ratio of players. We have analyzed the social network of two match-based games by extracting social graph from the matches. Further, we have analyzed how matches are formed by players and found that various matchmaking systems lead to different social network structures.

B. Cost-efficient and reliable cloud-based MMOG platform

Our initial work on this topic focuses on designing a cloud-based platform that can meet the computing demands of match-based MMOGs. We assume that a match-based

MMOG (see Sec. II-A3) is using exclusively cloud resources to serve its players. Once several players want to play together, the cloud platform needs to allocate enough computing resources to support the new match. The cloud platform needs to allocate all the matches to its rented cloud resources, as soon as possible. The first step of this research is to find a scheduling policy that minimizes the operational cost for the cloud-based platform, while providing service to players in a timely manner.

Cloud providers adopt pay-as-you-go as the billing model, under which users are charged by their usage of resource. Reducing the rental cost of cloud-platform while keeping the wait-time of players' match low is non-trivial. The number of players concurrently online fluctuates over time and is subject to various human factors (see Sec. II-A3). Thus, have varying computing demands. Although the cloud provides on-demand scaling, MMOG operators need to develop scaling policies by specifying types and numbers of VM to be used to lower rental cost. Finding the best provisioning and allocation policies to map user requests to provisioned VMs is key to reduce the operational cost of MMOG operators.

Currently, we are designing a scheduling strategy which optimizes rental cost and wait-time. We have formulated the resource provisioning and scheduling problems as Integer Programming problems. As obtaining the optimal result is computationally intensive, we limit the time to explore the optimal solution, and compare the obtained feasible solution with various heuristic methods; then pick the best result. Further, we are devising a method which makes use of reserved instances to reduce rental cost. Our preliminary results, obtained via trace-based simulation, indicate that the proposed method can obtain significant lower cost than the cost-obvious method while keeping the wait-time low. We plan to conclude this research during 2013 (Section III). Regarding reliability, we plan to investigate fault-tolerant mechanisms based on VM/server replication techniques.

C. Domain-specific match-based scalable MMOG system

This is work in progress. Currently, we are focusing on one specific type of game: Real Time Strategy games (RTS), a type of match-based games. Today, RTS games are one of the most popular genres of computer games. Although RTS games already entertain tens of millions of online players, the current games do not scale. For example, even popular RTS games such as the StarCraft series support in the same game match only up to 16 players and up to several hundreds of moving units for players to control.

Towards scaling RTS games, we are currently designing a new scalability mechanism, Area of Simulation (AoS), which combines and extends two mechanisms commonly used in the design Networked Virtual Environments (NVEs), area of interest (AoI) and event-based lockstep simulation (EBLS). Unlike traditional AoI approaches, which employ update-based consistency models, our AoS mechanism uses event-based consistency to manage the important areas of interest. Unlike EBLS, which is traditionally used to synchronize the entire virtual world, our AoS mechanism synchronizes only selected areas of the virtual world, that is, the areas of interest. We are designing an AoS-based architecture, which is able to use both our AoS and traditional AoI mechanisms simultaneously, dynamically trading-off consistency guarantees for game scalability. Our workload analysis results (Section II-A) indicate that a single AoI, which is used for other game genres, cannot support the demands of RTS games; consequently, we are designing our architecture to support the dynamic creation and management of multiple areas of interest. We are implementing and deploying this architecture as a real-world working system. Our initial results show that it can operate with an order of magnitude more players while fulfilling the tight operational requirements of RTS games, such as an update rate of over 25 updates per second.

D. Experimental research

In line with the CCGrid community, we see evaluating, validating and reproducing the proposed approaches in practice as very important. In my PhD research, I use both simulation and experimentation to evaluate our approaches. We have built an experimental tool, `RTSenv` [12], to help us evaluate our proposed scalable MMOG system in two clouds: EC2 and Azure. We are currently building a tool to evaluate the proposed cloud-based platform using real-world gaming traces. Further, we plan to apply our proposed research to a real-world application.

1) Experimental tool: The growing population of RTS games expects new game designs and more scalable games every year. However, few tools and environments exist for game designers and implementers; of these, even fewer are available to researchers and game communities. We have developed and released `RTSenv`, an environment and associated set of tools for RTS games. Our environment can configure and manage the main aspects of RTS games, such as maps, computer-controlled units, and game scenarios.

`RTSenv` leverages multi-cluster systems and reactive fault tolerance mechanisms to perform robust, multi-machine, and multi-instance game experiments. Using `RTSenv` in DAS-4, a multi-cluster system, EC2, and Azure, we have shown that `RTSenv` can be used in a variety of scenarios. Validating our hypothesis that real-world experimentations is useful for distributed system research, our results give evidence that several common assumptions made by researchers about game workloads do not hold in general for RTS games and thus warrant a more detailed investigation: a linear dependence between network traffic and the number of players, independence of resource consumption from the complexity of the game scenario, etc. Further, we are currently building a simulation tool which enable MMOG

researchers to evaluate their proposed approaches in the same environment using real-world traces.

2) Real world application: The selected real world application should be representative, and with large number of players. Moreover, the application should be open-source, otherwise it is impossible adapt it to the needs of our platform. We have chosen `OpenTTD` as the target application, which is an open-source, popular RTS game. `OpenTTD` has been developed since 2004 by a community of developers as an extension to the commercial game Transport Tycoon Deluxe. We are also redesigning the game to make it able to support a massive number of players to play on the same map.

III. 24-MONTH TIMELINE

This section presents a two-year timeline for my PhD (due end-2014). During months 1-2, I will design and evaluate more scheduling techniques to reduce the operational cost of proposed cloud-platform, by leveraging various scaling, migration, and billing methods (spot instances). During months 3-4, I plan to develop an RTS and RPG game simulator, and publish it as a open-source software. After the development of this simulator, I will compare during months 5-6 my proposed domain-specific scaling approach against the other approaches. In months 6-7, I plan to further analyze the workload of player mobility and design a more realistic model; further, I will evaluate the impact of player mobility on more MMOGs architectures. During months 8-12, I plan to design a fault-tolerant method which can ensure that, should a small part of the proposed cloud-platform fail, the whole system will still work and the players' QoS will not be severely affected. Towards the end of my PhD thesis, I plan to investigate the application of my conceptual contributions to more game genres and even more distributed application domains. I will release my developed software and traces I collected for the benefit of the CCGrid community.

IV. SUMMARY

This section describes my PhD research: the contributions and remaining work.

1. Characterizing of MetaGaming networks. We have collected and analyzed large-scale data from XFire.
2. Analysis and modeling of mobility traces collected from virtual world, and comparing them with real-world GPS traces. We have proposed a mobility model to generate synthesize trace.
3. Characterizing of match-based game. We have analyzed various aspects of matches, geographical distribution and social network of players.
4. Designing and implementing of a cloud-based MMOG platform. Currently, we are developing a scheduling policy which significant lower rental cost while keeping wait-time low. We will design a fault-tolerant mechanism for the cloud platform.
5. Designing and implementing a scalable Real Time strategy game. We are designing a new scalability mechanism, Area of Simulation. We will show that our proposed method can support an order of magnitude more players than the other approaches.
6. Designing and implementing the experimental tool `RTSenv` which can test the machine and in-game performance of RTS games under varying configurations. We plan to integrate our proposed methods into real world game `OpenTTD` to valid them in practice.

REFERENCES

[1] M. Claypool and K. Claypool, "Latency and player actions in online games," *Commun. ACM*, Nov 2006.

[2] A. Bharambe, J. Douceur, J. Lorch, T. Moscibroda, J. Pang, S. Seshan, and X. Zhuang, "Donnybrook: Enabling large-scale, high-speed, peer-to-peer games," in *SIGCOMM*, vol. 38, no. 4. ACM, 2008, pp. 389–400.

[3] B. Knutsson, H. Lu, W. Xu, and B. Hopkins, "Peer-to-peer support for massively multiplayer games," in *INFOCOM*, vol. 1, no. C. IEEE, 2004, pp. 96–107.

[4] J. Müller and S. Gorlatch, "GSM: a game scalability model for multiplayer real-time games," in *INFOCOM 2005*.

[5] V. Nae, A. Iosup, and R. Prodan, "Dynamic resource provisioning in massively multiplayer online games," *IEEE TPDS*, no. 3, pp. 380–395, 2010.

[6] G. Schiele, R. Suselbeck, A. Wacker, J. Hahner, C. Becker, and T. Weis, "Requirements of peer-to-peer-based massively multiplayer online gaming," ser. CCGRID 2007, pp. 773–782.

[7] S. Shen and A. Iosup, "The xfire online meta-gaming network: Observation and high-level analysis," in *MMVE 2011*.

[8] S. Shen, N. Brouwers, and A. Iosup, "Human mobility in virtual and real worlds: Characterization, modeling, and implications," TU Delft, Tech. Rep. PDS-2011-007, Nov. 2011.

[9] Y. Guo, S. Shen, O. Visser, and A. Iosup, "An analysis of online match-based games," in *MMVE 2012*, pp. 1 –6.

[10] R. Van de Bovenkamp, S. Shen, A. Iosup, and F. Kuipers, "Understanding and recommending play relationships in online social gaming," in *COMSNETS 2013*.

[11] Y. Guo and A. Iosup, "The game trace archive," in *NETGAMES 2012*.

[12] S. Shen, O. Visser, and A. Iosup, "Rtsenv: An experimental environment for real-time strategy games," in *NETGAMES 2011*, pp. 1–6.