國立交通大學

網路工程研究所

碩 士 論 文

雲 端 大 型 多 人 線 上 遊 戲 下 考 量 暫 離 玩 家 之 資源配置方法

Resource provisioning for MMOG clouds considering AFK players

研 究 生：陳治豪

指導教授：王國禎　教授

中 華 民 國 101 年 6 月

雲端大型多人線上遊戲下考量暫離玩家之資源配置方法

Resource provisioning for MMOG clouds considering AFK players

研 究 生：陳治豪　　Student：Chihhao Chen

指導教授：王國禎　　Advisor：Kuochen Wang

國 立 交 通 大 學

網 路 工 程 研 究 所

碩 士 論 文

A Thesis

Submitted to Institute of Network Engineering

College of Computer Science

National Chiao Tung University

in Partial Fulfillment of the Requirements

for the Degree of

Master

in

Computer Science

June 2012

Hsinchu, Taiwan, Republic of China

中華民國101年6月

雲端大型多人線上遊戲下考量暫離玩家之
資源配置方法

學生：陳治豪　　指導教授：王國禎 博士

國立交通大學網路工程研究所

摘 要

近年來大型多人線上遊戲(MMOGs)吸引全世界成千上萬的玩家上線，且成為個人娛樂中非常重要的一部份。為了能夠支援更多的線上玩家，許多遊戲提供商(如OnLive)利用雲端計算的優點發展雲端遊戲。雖然雲端遊戲提供商從雲端計算上獲得強大的計算能力，如何有效的分配雲端資源(如虛擬機器)來滿足玩家的需求和有效的雲端資源利用是一項重要的研究議題。在此篇論文中，我們對大型多人線上遊戲提出一個考量暫離玩家(AFK players)的有效資源配置方法，利用暫離玩家僅需較低服務品質(QoS)的特性，降低配置給予他們的資源(如幀率),從而支援更多的線上玩家。我們以魔獸世界做為個案研究，調查實際線上玩家的人數和暫離玩家的人數。我們利用類神經網路來預測玩家和暫離玩家的人數，並評估可額外支援的線上玩家的人數。

然後，我們預測需使用的虛擬機器數量。依據真實遊戲數據之評估結果顯示，我們的資源配置方法能夠額外支援11.04%的線上玩家，且在46.05%的觀察時間內能在每一個魔獸世界的王國的一個遊戲區減少一台虛擬機器的使用，以支援更多玩家。因此，我們的方法能有效降低雲端資源的浪費，並提高資源的使用率。

關鍵詞：暫離玩家、雲端計算、大型多人線上遊戲、資源分配。

Resource provisioning for MMOG clouds considering AFK players

Student: Chihhao Chen **Advisor: Dr. Kuochen Wang**

Department of Computer Science

National Chiao Tung University

Abstract

Massively Multiplayer Online Games (MMOGs) are attracting millions of players from all over the world and become an important part of personal entertainments. To support more online players, many game providers (such as OnLive) take advantage of cloud computing to develop cloud gaming services. Although cloud gaming providers obtain powerful computing power from clouds, how to allocate cloud resources (e.g., virtual machines, VM) so as to meet player requirements and to efficiently utilize cloud resources is a key issue. In this paper, we propose an efficient resource provisioning scheme for MMOGs considering AFK (away from keyboard) players, RP-AFK, in game clouds. We leverage the characteristic of AFK players, which only need low QoS for gaming, to decrease the provision of resources (e.g., frame rates) to them and to support active online players. We used World of Warcraft (WoW), which is currently the world's most-subscribed MMORPG (Massively Multiplayer Online Role Player Game), as a case study to measure the total numbers of players and AFK players. We predicted the total numbers of players and AFK players by using neural networks, which can provide high prediction accuracy, and evaluated the number of extra online players supported. Then we predicted the number of VMs required. Evaluation results based on real game data show that the proposed RP-AFK can support 11.04% extra online players. One less VM can be used in a game zone of a WoW realm for 46.05% of the observed time. Therefore, the proposed scheme can reduce the waste of cloud resources and increase the resource utilization to support more players.

Keywords: AFK, cloud computing, MMOGs, resource provisioning.

Acknowledgements

Many people have helped me with this thesis. I deeply appreciate my thesis advisor, Dr. Kuochen Wang, for his intensive advice and guidance. I would also like to show my appreciation for all the classmates in the *Mobile Computing and Broadband Networking Laboratory* for their invaluable assistance and suggestions. The support by the National Science Council under Grants NSC99-2221-E-009-081-MY3 and NSC101-2219-E-009-001 is also gratefully acknowledged. Finally, I thank my father, my mother and my friends for their endless love and support.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Over the past decade, Massively Multiplayer Online Games (MMOGs) have become popular and attract millions of players worldwide. Nowadays, more than half of Internet users also are MMOGs players [1]. To support more players, game providers rack their brains trying to think of new architectures for MMOGs. Today, many game providers take advantage of cloud computing to develop cloud gaming services. Onlive [2], Ubitus [3] and Gaikai [4] launched cloud gaming services during 2010 to 2011.

Cloud gaming is a type of online gaming that directly transmits on-demand streaming of game screens to players' devices [5]. In other words, players can play games without storing all game contents on their own devices. Game screens are streamed directly to players' devices after being processed by cloud gaming providers' servers. Although cloud gaming providers have powerful capability of computing, they may waste cloud resources without utilizing resources efficiently. Therefore, how to efficiently utilize cloud resources is an important issue.

An MMOG is a multiplayer video game which many players participate at the same time over the Internet. There are many different types of MMOGs in the market, including MMORPG (Massively Multiplayer Online Role Player Game, e.g., World of Warcraft (WoW) [6]), MMOFPS (Massively Multiplayer Online First Person Shooter, e.g., Counter Strike [7]), MMORTS (Massively Multiplayer Online Real Time Strategy, e.g., StarCraft II [8]) and so on. However, MMORPGs are the most popular games. Hundreds or thousands of players interact with other players worldwide via avatars, which are virtual roles, in a persistent virtual world. According to the number of players, MMORPGs share up to 95% in the MMOG market [1].

Although there are a large number of online players in MMOGs concurrently, we observed that some players didn't play games all the time and they are sometimes away from their computers (or away from keyboard, AFK) temporarily. Some of the online players even leave their computers doing something else for a long time, such as eating dinner, watching TV, taking a bath, going to sleep and so on. In this paper, we propose an efficient resource provisioning scheme considering AFK players. The proposed resource provisioning scheme can reduce the waste of cloud resources and increase the resource utilization by supporting more players.

The remainder of this paper is organized as follows. We introduce the background and related work in Chapter 2. Chapter 3 presents the proposed resource provisioning scheme considering AFK players and depict our game data collection method. Chapter 4 shows evaluation results. Finally, concluding remarks and future work are given in Chapter 5.

Chapter 2

Background

2.1 WoW

World of Warcraft (WoW) [6] is currently the world's most-subscribed MMORPG with 10.2 million subscribers as of February 2012 [9]. WoW is a multi-server architecture to support a lot of realms, which are instances of the game world [10], to support millions of players. There are 763 realms worldwide and 39 realms in Taiwan [11]. Players can only interact with other players in the same realm.

Who List is a component of the default WoW User Interface for searching online players [12]. We can search online players by name, guild, zone, class (e.g., hunter, priest, rogue, warlock, warrior), race (e.g., human, dwarf, gnome) or level (1 - 85). Note that *Who List* only returns 50 players in one request. Besides, WoW also provides many APIs which is a set of functions for players. The list of API is updated and announced at [13].

2.2 Frame rate

Frame rate in MMOGs is the frequency of consecutive images, frames, refreshed on a monitor and expressed in frames per second (FPS) [14]. In multi-server MMOG architecture, frames are produced by players' computers. In cloud gaming, frames are produced and delivered to players' devices by servers. Frame rate involves players' experiences because low FPS causes avatar movement discontinuously and decreases players' experiences of games. The acceptable frame rate is between 30 to 60 FPS in modern MMOGs [14].

2.3 AFK

AFK is commonly used in the internet, not just in MMOGs, to denote that an online user is temporarily away from the computer, or is not actually paying attention to the computer [15]. For example, in the WOW, players are marked AFK automatically after 5 minutes without any interaction [16] and players' avatars could sit on the ground, as shown in Figure 1. In Figure 2, if WoW players are AFK for 30 minutes, they will be logged off and the game screen is switched to the character selection screen. And WoW disconnects players from servers if players are AFK for 60 minutes [16].

Figure 1: (a) An active player and (b) an AFK player in the WoW [6].

(a) After 30 minutes (b) After 60 minutes

Figure 2: A player is AFK for (a) 30 minutes and (b) 60 minutes in the WoW [6].

Because AFK players are not at their computers or are not playing games at that time, they only need low QoS for gaming. For example, AFK players don't care about game experience, such as frame rates and avatars' movements. If cloud gaming providers decrease the provision of resources to AFK players, they can save cloud resources and support more active players.

2.4 OnLive

OnLive [1] is one of the famous cloud gaming providers that provides the world's highest performance cloud gaming service [17]. OnLive stores the games on remote servers and delivers on-demand streaming of game screens (frames) via the Internet. OnLive has partnered with over 50 well-known games companies in the business and has provided nearly 300 games on the service [18] [19].

2.5 Game Data collection

For gaming designers, it is one of the most significant things that they take account of players' behavior when designing games. Lee et al. [1] provided a good starting point to exploring users' game play time and presented a WoW history dataset that includes players' game play time and a number of attributes, such as their races, current levels, and in-game locations, etc. They used the *who* command provided by WoW to collect the dataset between Jan. 2006 and Jan. 2009 in Taiwan. They investigated players' behavior, including the play time, session count, and session play time. Finally, they gave a sample use of the dataset to model and predict the subscription time of players and the usage time of avatars [1].

2.6 Resource provisioning

To have an efficient resource provisioning scheme for MMOGs, it needs a prediction algorithm with high accuracy to predict workloads. Nae et al. [20] proposed a dynamic resource provisioning method that can efficiently allocate resources in MMOG. They predicted MMOG workloads by using a neural network-based predictor which can provide high prediction accuracy. Their simulation results showed that the neural network-based predictor has lower prediction error than other prediction methods, such as last value and moving average.

2.7 Comparison of different approaches qualitatively

In Table I, we compare the proposed RP-AFK with other approaches. Although WoW logs off and disconnects players when players are AFK for 30 minutes and 60 minutes, respectively, it causes AFK players to re-login and still wastes resources for the first 30 minutes. Remind that related work did not take AFK players into consideration when allocating resources.

Table I: Comparison of different approaches qualitatively.

Approach	WoW [6]	Nae [20]	OnLive [2]	RP–AFK (proposed)
Description	MMORPG by Blizzard Entertainment	Resource provisioning for MMOG	Cloud gaming platform	Resource provisioning for MMOG clouds
Architecture	Multi-server	Multi-server	Cloud gaming	Cloud gaming
Resource provisioning	N/A	Based on predicted MMOG workloads	N/A	Based on the total numbers of players and AFK players
Considering AFK players	AFK players logged off after 30 minutes and disconnected from servers after 60 minutes	None	None	Decreases the provision of resources (e.g., frame rates) to AFK players

Chapter 3

Proposed Resource Provisioning Considering AFK Players

3.1 Resource provisioning considering AFK players

We propose an efficient resource provisioning scheme for MMOGs considering AFK players (RP-AFK) in cloud gaming. We leverage the characteristic of AFK players, which only need low QoS for gaming, to decrease the provision of resources (e.g., frame rates) to them and to support more players. The flowchart of the proposed RP-AFK is showed in Figure 3 and the details of the RP-AFK are described in the following.

1) Read historical game data that include total numbers of players and AFK players of a WoW realm in Taiwan.
2) Predict the total numbers of players and AFK players using a neural networks predictor based on historical game data.
3) Predict the number of VMs required at each time step during time t to time $t + VM\ start\ time$ (time period).
4) Find the maximum number of VMs required ($n_{max}^{(VM)}$) at the next time period. Note that we need to create new VMs in advance to prevent the existing VMs from overloading.
5) Compare $n_{max}^{(VM)}$ with the number of VMs used currently. If $n_{max}^{(VM)}$ is equal to the number of VMs used currently, go to step 8. If $n_{max}^{(VM)}$ is less than the

number of VMs used currently, go to step 6. Otherwise, go to step 7.

6) Close the VMs that aren't required. Then go to step 8.

7) Create extra VMs required. Then go to step 8.

8) Collect current game data and store in historical game database.

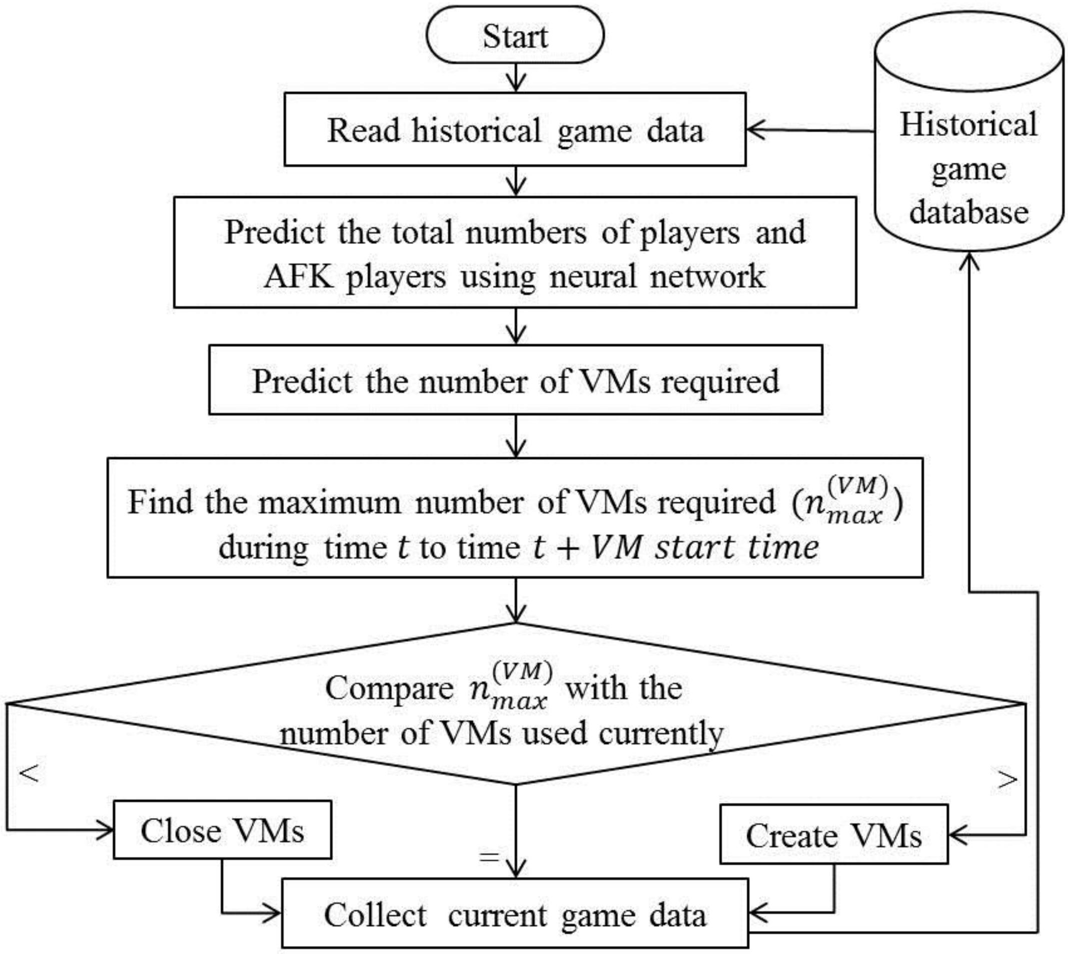

Figure 3: The flowchart of the proposed RP-AFK.

3.2 Game data collection

We use WoW as a case study and describe how we collected game data that include the number of AFK players. The details of our game data collection method are described in the following.

1) Create an AddOn for WoW.

2) Use *Who List* to search online players by class and zone because *Who List* only

returns 50 avatars in one request.

3) Add the online players searched to our friend list because API *UnitIsAFK* only works for friends.

4) Use API *UnitIsAFK* to verify whether the online players are AFK.

5) Output the total numbers of players and AFK players to the chat window.

6) Use API *LoggingChat* to log chat data to a file (historical game database).

In MMOGs, players are usually AFK in towns or cities because they will not be attacked by oppositional players or monsters. We collected the total numbers of players and AFK players in six cities of a WoW realm in Taiwan, as shown in Figure 4. We found that most of players are staying and AFK in a city, called Orgrimmar. Therefore, we just collected game data in this city.

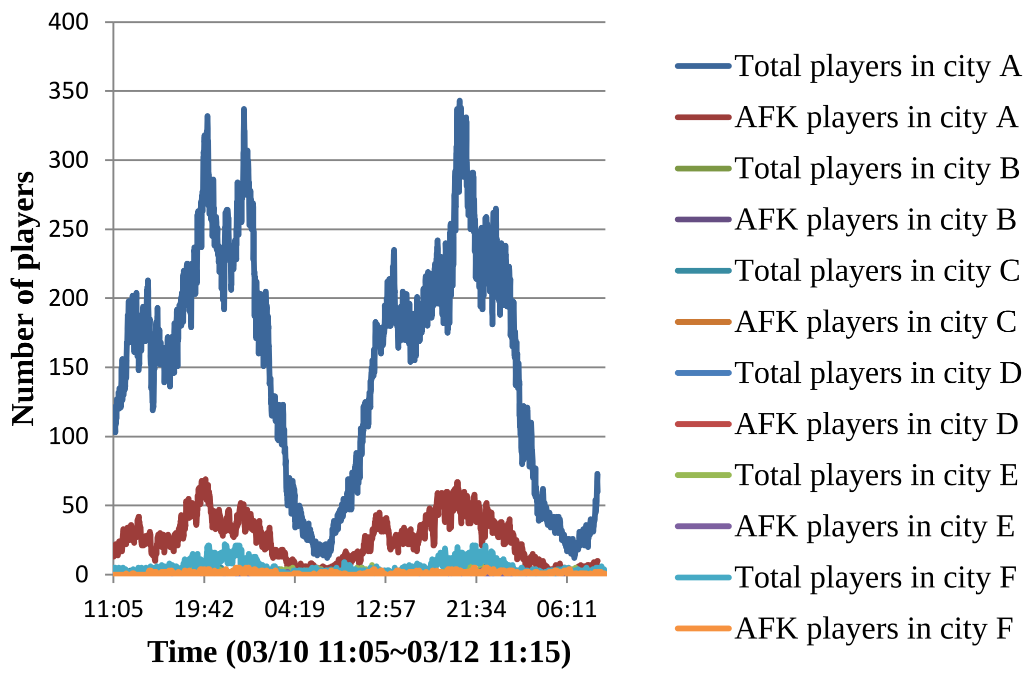

Figure 4: The total numbers of players and AFK players in six cities of a WoW realm in Taiwan.

Figure 5 shows the total numbers of players and AFK players in Orgrimmar. The number of AFK players is showed daily with variations and is about 25% of the total number of players. That is, if there is more number of total players, there will have more number of AFK players so as to save more cloud resources. We collected game data from 2012/03/10 to 2012/5/20 and predicted the total numbers of players and AFK players in the near future by using a neural network-based predictor.

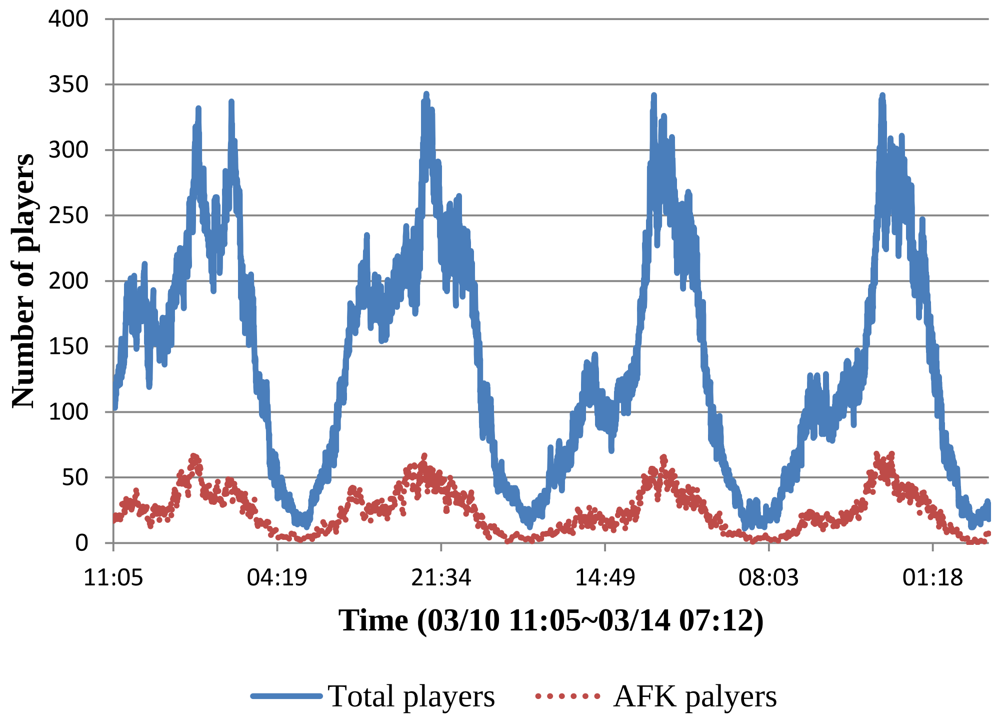

Figure 5: The total numbers of players and AFK players in a city, Orgrimmar, of a WoW realm in Taiwan.

3.3 Neural network-based predictor

The neural network-based predictor provides a robust and highly accurate approach to approximating a real target function [20]. A neural network-based predictor with a time series input is showed in Figure 6 [21]. The network has 20 network inputs and one network output. We input the number of players from time step 1 to time step 20 and predict the number of players at time step 21. The neural network has two hidden layers. Hidden layer A and hidden layer B have 20 neurons and 10 neurons, respectively. Each layer's neuron performs calculations based on the following equations:

$$x_j = f^{(1)}\left(\sum_{i=1}^{20} w_{i,j} n_i + b_j\right)$$

$$y_j = f^{(1)}\left(\sum_{i=1}^{20} w'_{i,j} x_i + b'_i\right)$$

$$n_{21} = f^{(2)}\left(\sum_{i=1}^{10} w''_i y_i + b''\right)$$

$$\text{where} \quad f^{(1)}(x) = \frac{1}{1+e^{-x}}$$

$$f^{(2)}(x) = x$$

Note that x_j is the output, n_i is an input, $w_{i,j}$ is a weight modifying n_i, and b_j is the bias of the hidden layer A. y_j is the output, x_i is an input, $w'_{i,j}$ is a weight modifying x_i, and b'_j is the bias of the hidden layer B. n_{21} is the output, y_i is an input, w''_i is a weight modifying y_i, and b'' is the bias of the output layer [21]. In addition, $f^{(1)}(x)$ is a log sigmoid transfer function for two hidden layers neurons and $f^{(2)}(x)$ is a linear transfer function for the output layer neuron. We used this neural network-based predictor to predict the total numbers players and AFK players.

Then, we predict the number of VMs required by the following equation:

$$n_i^{(vm)} = \frac{\left(n_i^{(total)} - n_i^{(AFK)}\right) \times R^{(active)} + n_i^{(AFK)} \times R^{(AFK)}}{VM\ capacity}$$

where $n_i^{(vm)}$ is the predicted number of VMs required at time step i, $n_i^{(total)}$ is the predicted total number of players at time step i, $n_i^{(AFK)}$ is the predicted number of AFK players at time step i, $R^{(active)}$ is average resources used (CPU utilization) by an active player, $R^{(AFK)}$ is average resources used (CPU utilization) by an AFK player, and $VM\ capacity$ is number of players supported in a VM.

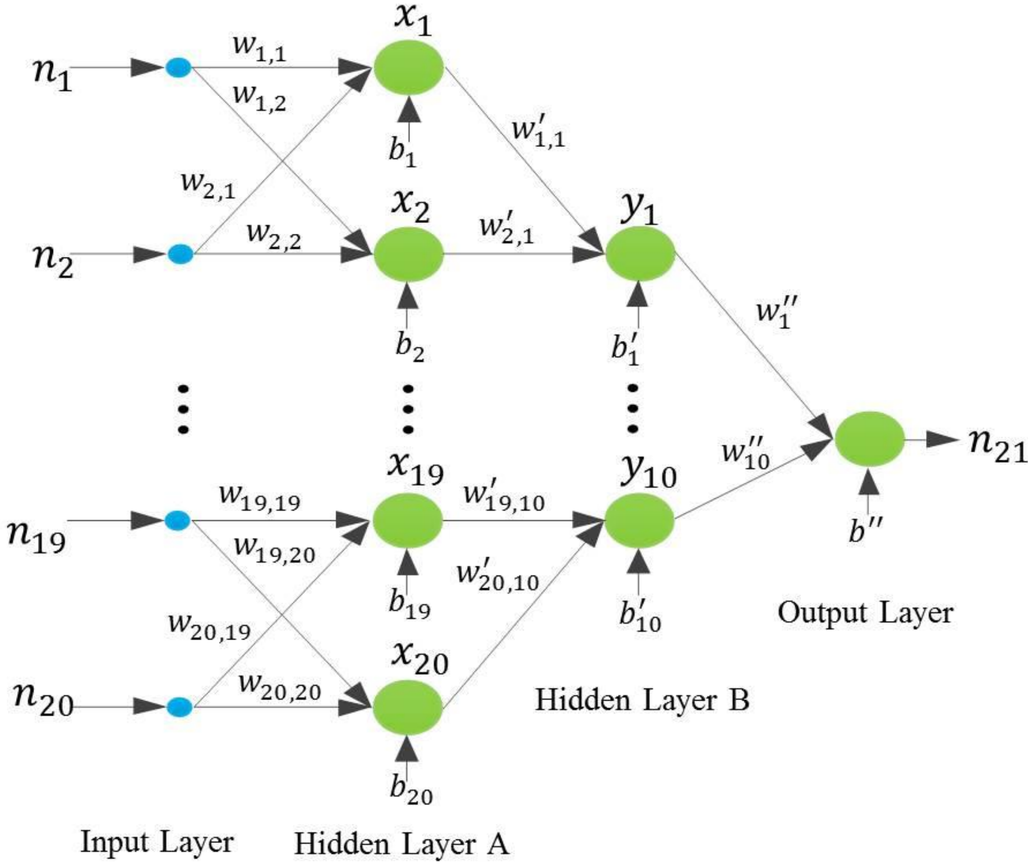

Figure 6: A neural networks-based predictor for numbers of total players and AFK players.

Chapter 4

Evaluation Results

4.1 Experimental setup

For evaluation, game data including the total numbers of players and AFK players were collected by using our game data collection method, as described in Chapter 4. We used MATLAB [22] as a prediction tool to predict the total numbers of players and AFK players in the near future. The evaluation period was from 2012/03/10 to 2012/5/20. To evaluate the number of VMs used, we assumed one VM can support 31 avatars [23] and VM startup time is 10 minutes [24]. If players are AFK, we reduced their frame rates to 8 FPS, which is the lowest rate supported in WoW. Experimental parameters are summarized in Table II.

Table II: Experimental parameters.

Game data	From WoW
Prediction technique	Neural network
Prediction tool	MATLAB API (*nntool*)
Evaluation period	2012/03/10 11:05 ~ 2012/5/20 18:30
VM capacity	31 avatars [23]
VM startup time	10 minutes [24]
AFK player's frame rate	8 FPS (lowest rate supported in WoW)

4.2 Comparison of three prediction methods

We compared the neural network-based prediction method with the last value and moving average prediction methods. We aimed to minimize the prediction error (PE) which was calculated by the following equation [20]:

$$PE = \frac{\sum_{i=1}^{N} \left| n_i^{(real)} - n_i^{(pred)} \right|}{\sum_{i=1}^{N} n_i^{(real)}}$$

where $n_i^{(real)}$ is the actual number of players at time step i and $n_i^{(pred)}$ is the predicted number of players at time step i. A prediction method with the lower PE represents the better prediction accuracy.

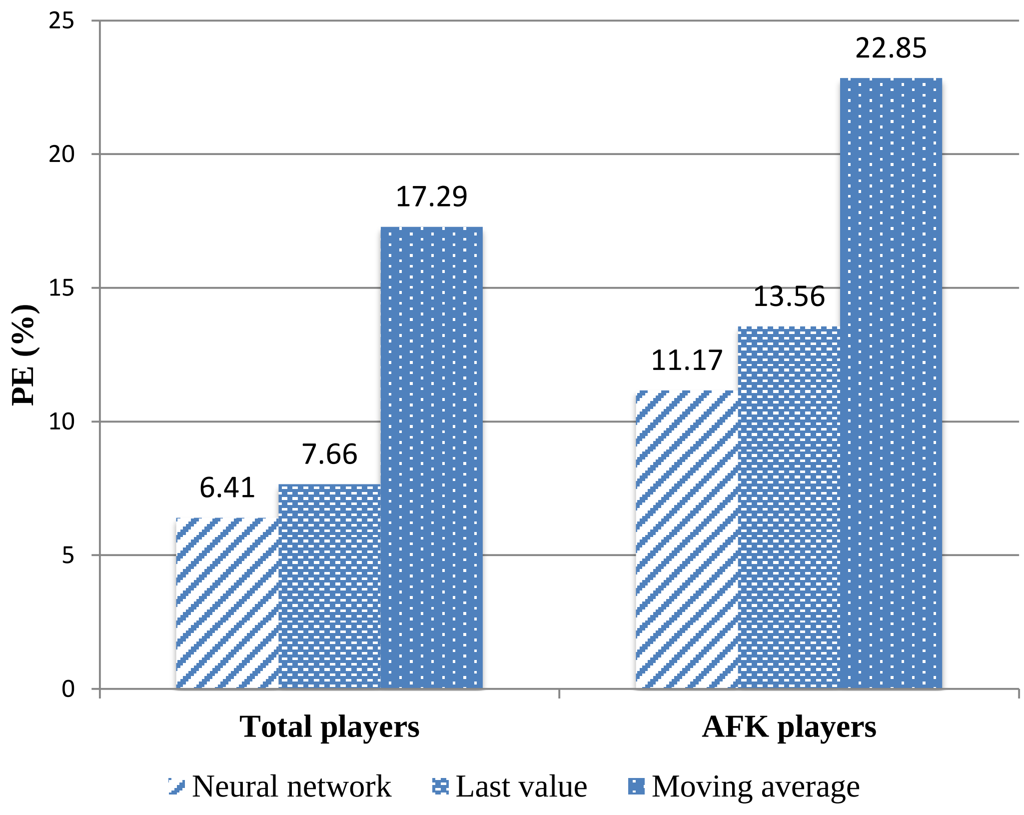

Figure 7: Prediction errors (PEs) of the three prediction methods.

The PEs of the above three prediction methods are showed in Figure 7. Since the neural network-based predictor has high prediction accuracy, the PE of the neural network-based prediction method is much lower than that of moving average prediction and that of last value prediction.

4.3 The percentage of extra players supported considering AFK players

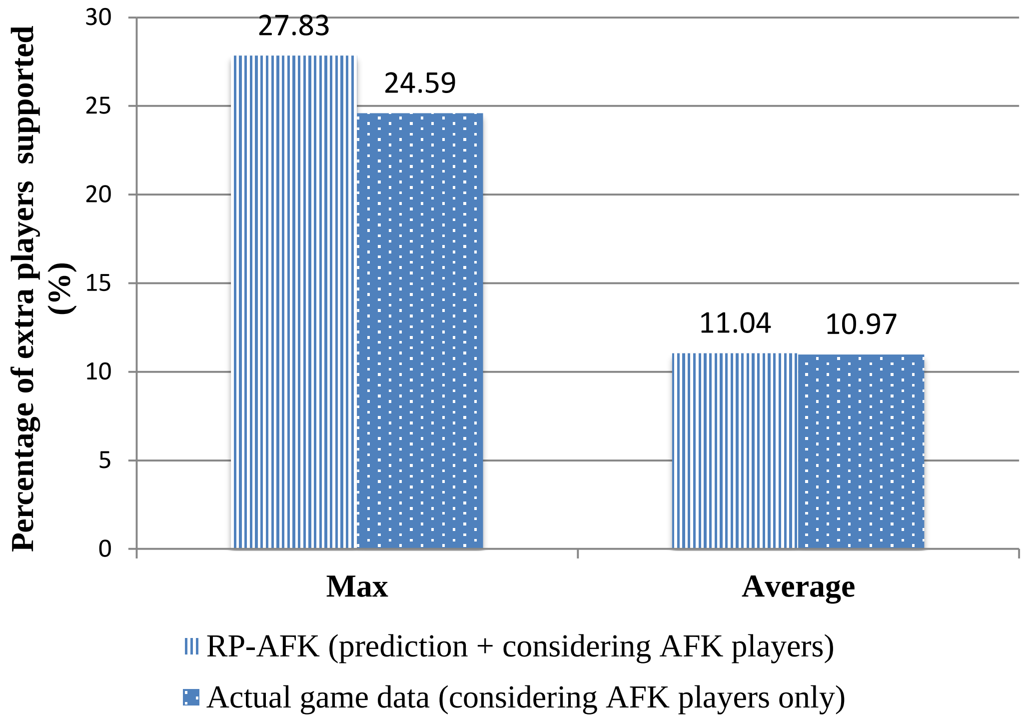

Figure 8: The number of extra players supported considering AFK players.

We evaluated the percentage of extra players supported considering AFK players. Experimental results show that the average percentage of extra players supported for RP-AFK is 11.04%, and it is almost equal to the average percentage of extra players supported based on the actual game data,, as shown in Figure 8.

4.4 The average usage of VMs in one zone of a WoW realm

We also evaluated the usage of VMs in one zone of a WoW realm. Figure 9 shows RP-AFK used one less VM for 46.05% of the observed time period and used two less VMs for 1.75% of the observed time period. It supports that our scheme can save cloud resources by decreasing the provision of resources to AFK players.

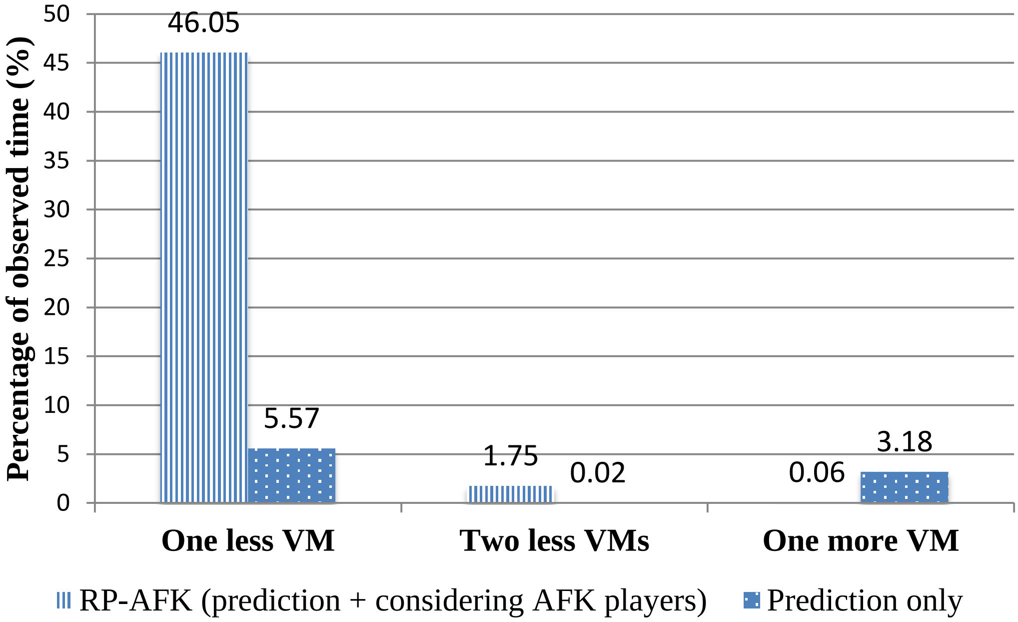

Figure 9: The average usage of VMs in one zone of a WoW realm.

4.5 Under-allocation and over-allocation

Finally, we evaluate the under-allocation and over-allocation of RP-AFK. Under-allocation means that the assignment of the number of players is more than the number of players that VMs can handle [20] and over-allocation means that the number of VMs allocated is higher than the number of VMs needed [25]. Figure 10 shows the under-allocation of RP-AFK is slightly higher than the under-allocation of resource allocation based on actual game data. However, the over-allocation of

RP-AFK is smaller than the over-allocation of resource allocation based on actual game data.

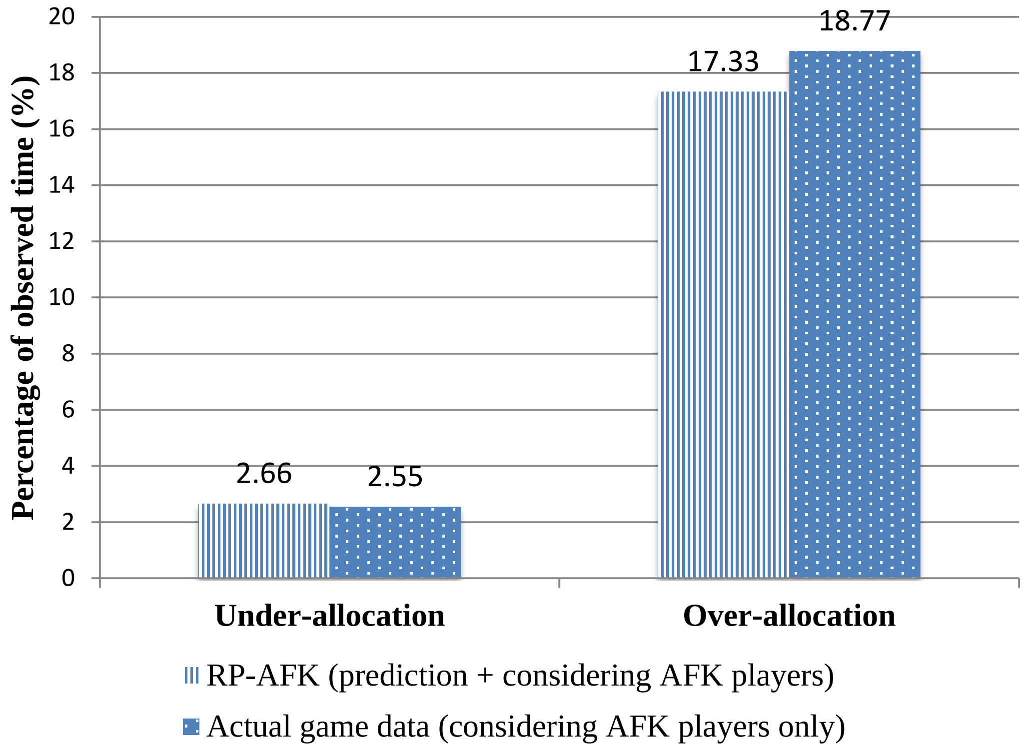

Figure 10: Under-allocation and over-allocation.

Chapter 5

Conclusions

5.1 Concluding remarks

We have presented an efficient resource provisioning scheme considering AFK players (RP-AFK) for MMOG clouds. We leverage the characteristic of AFK players, which only need low QoS for gaming, to decrease the provision of resources to them and to support more online players. We used WoW as a case study to illustrate the proposed resource provisioning scheme that is valuable to MMOG cloud providers. The proposed RP-AFK scheme can support 11.04% extra players and use one less VM for 46.05% of the observed time in a game zone of a WoW realm. Therefore, our scheme can reduce the waste of cloud resources and increase resource utilization by supporting more online players.

5.2 Future work

In our current resource provisioning scheme, we focused only on number of players. In the future, we will consider players' interaction in our resource provisioning scheme. In addition, we will implement our proposed resource provisioning scheme considering AFK players in an experimental MMOG cloud.

Bibliography

[1] Y. Lee, K. Chen, Y. Cheng and C. Lei, "World of Warcraft avatar history dataset," in *Proc. the Second Annual ACM Conference on Multimedia Systems*, pp. 123-128, Feb. 2011.

[2] "Onlive," [Online.] Available: *http://www.onlive.com/*.

[3] "Ubitus," [Online.] Available: *http://www.ubitus.net/index.html*.

[4] "Gailai," [Online.] Available: *http://www.gaikai.com/*.

[5] "Cloud gaming," [Online.] Available: *http://en.wikipedia.org/wiki/Cloud_gaming*.

[6] "World of Warcraft," [Online.] Available: *http://us.battle.net/wow/en/*.

[7] "Counter Strike," [Online.] Available: *http://www.counter-strike.net/*.

[8] "StarCraft II," [Online.] Available: *http://us.battle.net/sc2/en/*.

[9] "World of Warcraft subscriber numbers dip 100,000 to 10.2 million," [Online.] Available: *http://wow.joystiq.com/2012/02/09/world-of-warcraft-subscriber-numbers/*.

[10] "Realms list," [Online.] Available: *http://www.wowwiki.com/Realms_list*.

[11] "Realms," [Online.] Available: *http://www.wowwiki.com/Realm*.

[12] "Who," [Online.] Available: *http://www.wowwiki.com/Who_List*.

[13] "World of Warcraft API," [Online.] Available: *http://www.wowwiki.com/World_of_Warcraft_API*.

[14] "Frame rate," [Online.] Available: *http://en.wikipedia.org/wiki/Frame_rate*.

[15] "AFK - away from keyboard," [Online.] Available: *http://internetgames.about.com/od/glossary/g/afk.htm*.

[16] "Away," [Online.] Available: *http://www.wowwiki.com/Away*.

[17] "Top cloud computing gaming providers," [Online.] Available:

http://www.cloudtweaks.com/2011/03/top-cloud-computing-gaming-providers/.

[18] "OnLive: partners," [Online.] Available: http://www.onlive.com/corporate/partners.

[19] "OnLive: all game," [Online.] Available: http://www.onlive.com/games/featuredgames#&tab=all_games.

[20] V. Nae, A. Iosup and R. Prodan, "Dynamic resource provisioning in massively multiplayer online games," *IEEE Transactions on Parallel and Distributed Systems*, pp. 380-395, March 2011.

[21] Truong Vinh Truong Duy, Y. Sato and Y. Inoguchi, "Performance evaluation of a green scheduling algorithm for energy savings in cloud computing," in *Proc. IEEE International Symposium on Parallel & Distributed Processing, Workshops and Ph.D Forum (IPDPSW)*, pp. 1-8, April 2010.

[22] "Matlab," [Online.] Available: *http://www.mathworks.com/products/matlab/*.

[23] Y. Lee and K. Chen, "Is server consolidation beneficial to MMORPG? a case study of World of Warcraft," in *Proc. IEEE 3rd International Conference on Cloud Computing (CLOUD)*, pp. 435-442, July 2010.

[24] "How quickly will systems be running," [Online.] Available: *http://aws.amazon.com/ec2/faqs/#How_quickly_will_systems_be_running*.

[25] "Do you overwork your resources," [Online.] Available: *http://www.brighthub.com/office/project-management/articles/11638.aspx*.