國立交通大學

資訊學院資訊科技（IT）

產業研發碩士專班

碩 士 論 文

雲端大型多人線上遊戲遊戲下基於負載預測之資源分配方法

SVM-based load prediction for resource allocation in MMOG clouds

研 究 生：吳昭霖

指導教授：王國禎　博士

中 華 民 國 一 百 年 六 月

雲端大型多人線上遊戲遊戲下基於負載預測之資源分配方法

SVM-based load prediction for resource allocation in MMOG clouds

研　究　生：吳昭霖　　　Student：Zhao-Li Wu

指導教授：王國禎　　　Advisor：Kuochen Wang

國　立　交　通　大　學

資訊學院資訊科技（IT）產業研發碩士專班

碩　士　論　文

A Thesis

Submitted to College of Computer Science

National Chiao Tung University

in Partial Fulfillment of the Requirements

for the Degree of

Master

in

Industrial Technology R & D Master Program on

Computer Science and Engineering

June 2011

Hsinchu, Taiwan, Republic of China

中華民國一百年六月

雲端大型多人線上遊戲下基於負載預測之資源分配方法

學生：吳昭霖　　指導教授：王國禎 博士

國立交通大學資訊學院產業研發碩士專班

摘 要

近年來大型多人線上遊戲(MMOG)已成為主流遊戲，其使用者超過數百萬人。對大型多人線上遊戲來說，值得研究的課題非常的多，其中又以資源分配為重要的研究課題之一。如何有效的進行資源分配使遊戲可以正常運作，又可使其滿足使用者之QoS，如讓使用者滿意之反應時間，是為產業與學界共同研究的課題。本論文利用支持向量機來進行資源預測並結合雲端的資源分配，以便有效地進行資源分配以提升使用者之QoS。支持向量機能運算大量的資料並具備快速計算的能力，使其被廣泛地運用在各種預測上。由於其預測準確率高於類神經網路，因此更能有效用於MMOG負載的預測。我們的方法是基於支持向量機於多伺服器架構下進行MMOG負載的預測，即利用每個地圖區域的虛擬機器之歷史負載量來預測該區域未來負載量的層級，從而進行有效的資源分配，以滿足使用者反應時間的要求。實驗結果顯示，我們基於支持向量機的預測方法在預測準確率上比基於類神經網路的預測方法高12.24%，且其在虛擬機器的使用數量上也減少8%。

關鍵詞：雲端計算、負載預測、大型多人線上遊戲、類神經網路、資源分配、支持向量機。

SVM-based load prediction for resource allocation in MMOG clouds

Student: Zhao-Li Wu **Advisor: Dr. Kuochen Wang**

Industrial Technology R & D Master Program of
Computer Science College
National Chiao Tung University

Abstract

In recent years, massively multiplayer online games (MMOGs) have hundreds to thousands of active concurrent players so as to have become a popular research topic in academia and industry. MMOGs consume huge resources due to a massive number of players. To utilize resources efficiently, we integrate MMOGs with cloud computing environments. An MMOG game world is composed of game regions. MMOG load management allocates resources (i.e. virtual machines, VMs) to each game region. In this paper, we propose an SVM (support vector machine)-based load prediction scheme to first forecast cloud resources needed based on the loading class of each game region in an MMOG cloud. Then we allocate resources needed to achieve reasonable response time with less resources used. Simulation results show that the proposed SVM-based load prediction is 12.24% better than neural network-based load prediction in term of prediction accuracy. In addition, the proposed SVM-based load prediction reduces 8% of the number of VMs used compared with the neural network-based load prediction.

Keywords: cloud computing, load prediction, MMOG, neural network, resource allocation, support vector machine.

Acknowledgements

Many people have helped me with this thesis. I am in debt of gratitude to 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 inspirations. The support by the National Science Council under Grants NSC 99-2221-E-009-081-MY3 is also gratefully acknowledged. Finally, I thank my father, my mother and my friends for their endless love and support.

Contents

Abstract (in Chinese)……………………………………………………………i

Abstract iii

Contents...v

List of Figures vii

List of Tables...viii

Chapter 1 Introduction 1

1.1 Motivation...1

1.2 Resource allocation 2

1.3 Load prediction 2

1.4 Research objective 3

1.5 Thesis organization 3

Chapter 2 Preliminaries...5

2.1 SVM..5

2.2 The Applications of SVM...6

2.3 Pros and cons comparison of the proposed SVM-based and neural network-based load prediction 6

Chapter 3 Related Work..8

3.1 Neural network-based load prediction 8

3.2 Qualitative comparison of proposed SVM-based with neural network-based load prediction 8

Chapter 4 Proposed SVM-based Load Prediction for Resource Allocation....... 10

4.1 SVM-based resource allocation architecture for MMOGs 10

4.2 Preprocessing 11

4.3 SVM-based load prediction 14

4.4 Resource assignment.. 17

Chapter 5 Performance Evaluation 18

5.1 Simulation setup and evaluation metrics 18

5.1.1 Prediction accuracy 19

5.2 Comparison between SVM-based and NN-based load prediction 19

Chapter 6 20

Conclusion 20

6.1 Concluding remarks 20

6.2 Future work.. 21

Bibliography.. 22

List of Figures

Figure 1: Resource allocation for each region [13]. 3

Figure 2: SVM hyperplane and the support vector [9]. 6

Figure 3: Proposed SVM-based resource allocation architecture for MMOGs. 10

Figure 4: Preprocessing procedure and the MMOG player datasets format. 13

Figure 5: Preprocessing procedure for deriving the loading class of each game region *j* at time interval *i*. 14

Figure 6: SVM using the loads in previous time intervals to predict the future load of a region. 15

Figure 7: Flowchart of the proposed SVM-based load prediction and resource assignment. 16

Figure 8: Prediction accuracy comparison of the three load prediction schemes. 20

List of Tables

Table I. Pros and cons comparison of the proposed SVM-based and neural network-based load prediction........7

Table II. Qualitative comparison of the proposed SVM-based with neural network-based load prediction........9

Table III. MMOG loading level and its SVM category........17

Table IV. Simulation parameters [1,16]........18

Chapter 1

Introduction

1.1 Motivation

In recent years, the development of massively multiplayer online games (MMOGs) has been booming. In [6], it is reported that over 55% of Internet users are also online gamers, while the number of MMOG players grows to 21 million in 2011. Today, MMOGs causes high interest in academia and industry. MMOGs need to consume huge resources due to a massive number of players. MMOG game servers are responsible for communicating information among players. The challenges of MMOGs include scalability, reasonable resource allocation, QoS, fairness among players and meeting response time requirement of players. An important goal is to allocate or release resources (such as virtual machines - VMs) to meet the response time requirement of players and to have efficient resource utilization for power saving. In this paper, we pay attention to the resource allocation problem in MMOGs and use the multi-server architecture in the cloud. Cloud computing integrates distributed computing and virtualization techniques. It has powerful computing ability and high scalability. Cloud computing applications are for distributed computing, data storage, and sharing of various software. Cloud computing environments are suitable for MMOGs. By the statistics of WoW (World of Warcraft), there are millions of players online [3]. So we leverage the powerful computing ability and high scalability of cloud computing environments to deal with a large number of MMOG players. In an MMOG cloud, it flexibly uses VMs instead of physical machines to reduce response time and enhance QoS of players. That is, it can allocate the resources of an MMOG cloud more effectively. In [3], it mentioned that players' log files can be determined by (i) system workload modeling/prediction, (ii) system QoS modeling/prediction, and (iii) player loyalty modeling/prediction. Based on these reason, we use players' log files for predicting the VMs load in the proposed scheme.

The load of MMOGs depends on the number of active concurrent players [12]. Therefore, fast and accurate load prediction algorithms are required to dynamically allocate resources for MMOGs so as to provision VMs to game sessions with the minimum resources dynamically.

1.2 Resource allocation

A game world is composed of regions. Each region is supported by multiple VMs of servers. Assume that each VM has the same CPU, memory, and network bandwidth capability. The methods of resource allocation can be classified into two categories, *real-time resource allocation* and *pre-allocation* [10]. The real-time resource allocation scheme dynamically assigns resources when the user has the demand. The pre-allocation scheme can achieve better system resource stability, but a good load prediction algorithm is needed. In this paper, we combine real-time resource allocation with a load prediction scheme to predict the user load and allocate resources efficiently. By prediction, resources can be allocated before players' login to the MMOG servers.

1.3 Load prediction

Load prediction is based on historical data to forecast the load level of each region in advance for VM assignment. Load prediction can help to achieve efficient real-time resource allocation. It can help reduce resource allocation time to meet the response time requirement of each player. As shown in Figure 1, initially each region is allocated a VM. Then, the number of VMs used in each region will be adjusted based on number of players.

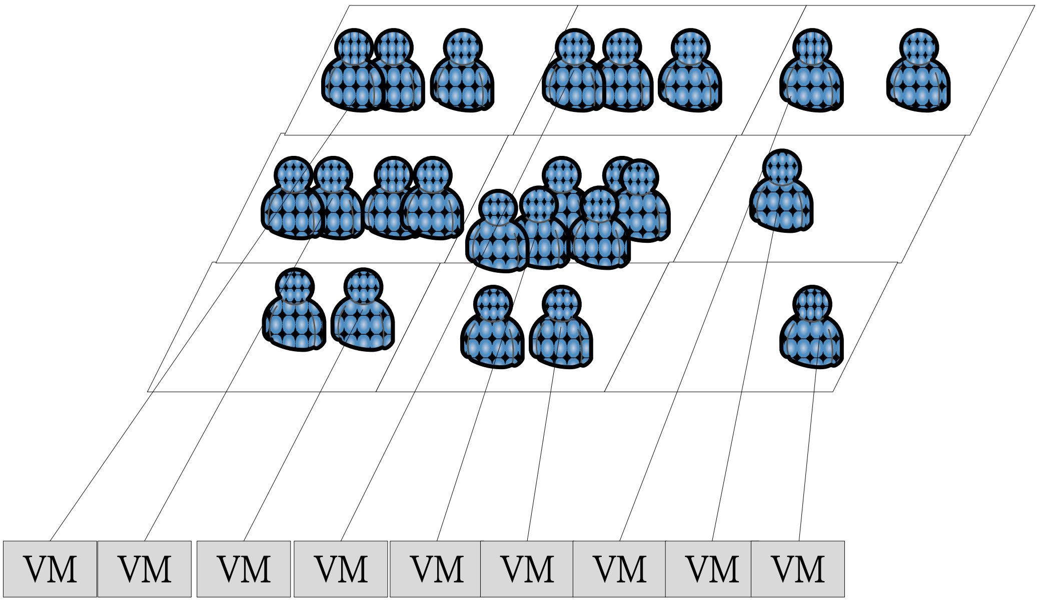


Figure 1: Resource allocation for each region [13].

1.4 Research objective

SVM (support vector machine) is a suitable tool for predicting MMOGs load. In this paper, we propose an efficient real-time resource allocation scheme to support the workload generated by MMOGs players. The task involves predicting the load of each region in an MMOG game world using historic players' log files. We use SVM to predict MMOGs game world load. We integrate current MMOG multi-server architecture with a cloud computing environment to achieve better performance.

1.5 Thesis organization

The rest of this paper is organized as follows. In Chapter 2, we introduce the basic concept of SVM. In Chapter 3, we review an existing prediction-based real-time resource provisioning scheme for MMOGs. We present the proposed SVM-based load prediction scheme in detail in Chapter 4. In Chapter 5, we show simulation results that include comparison with neural network-based load prediction [13] in terms of prediction accuracy. In Chapter 6, we conclude this paper and outline future work.

Chapter 2

Preliminaries

2.1 SVM

The support vector machine (SVM) was developed by Vapnik (1995) [1]. The main idea of SVM comes from the binary classification, namely to find a hyperplane as a segmentation of the two classes to minimize the classification error. The maximum margin hyperplane gives the maximum separation between decision classes. The training data are closest to the maximum margin hyperplane; it is called support vectors [1]. SVM analyzes data and recognize patterns for classification and regression analysis (i.e. time sequence analysis). SVM concentrates on applying regression to short-term electricity load forecasting. Its forecasting accuracy outperforms other forecasting models. SVM has good performance on reducing the outliers and improves the prediction accuracy efficiently. The SVM parameters are defined as follows: $\{X_i, Y_i\}, i = 1, \ldots, n, \; Y_i \in \{-1, 1\}, \; X_i \in R^2$, where X_i is a training vector, Y_i represents a class. Figure 2 shows the SVM hyperplane and the support vector w [9].

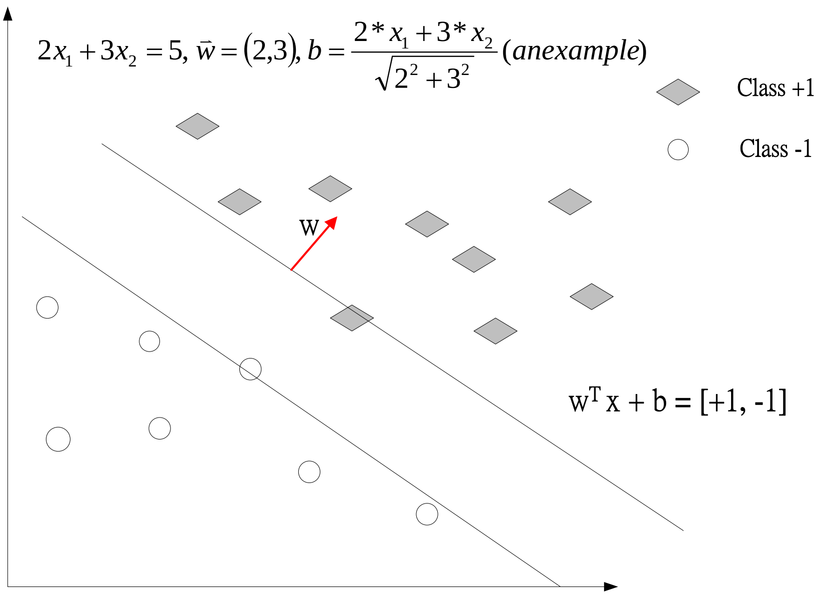


Figure 2: SVM hyperplane and the support vector [9].

2.2 The Applications of SVM

SVM can be applied to some applications, such as flood forecasting, power prediction [18], image recognition (large dataset), three-dimensional classification, text classification, and time series prediction[1] [18].

2.3 Pros and cons comparison of the proposed SVM-based and neural network-based load prediction

Table I shows the pros and cons between SVM-based and the NN-based load prediction. For the SVM-based load prediction, it has low performance with a small amount of data. The NN-based load prediction has slow training speed, complicated neural nodes, and high resource consumption.

Table I. Pros and cons comparison of the proposed SVM-based and neural network-based load prediction

Approach	**Pros**	**Cons**
SVM-based (proposed)	• Fast training speed[18] • Easy to implement [1] [18] • Optimal solution [17]	• Low performance with a small amount of data
Neural network (NN)-based [5] [16]	• Accepting different input data formats [15] • Ambiguous inference ability [16]	• High resource consumption [15] • Slow training speed [15] • Complicated neural node

Chapter 3

Related Work

3.1 Neural network-based load prediction

In [5], it proposed a neural network (NN) prediction-based method for dynamic resource provisioning and scaling of MMOGs in distributed grid environments. The load prediction method predicts the future game world entity distribution from history trace data by using the neural network-based method. It developed generic analytical game load models used to forecast the future hot-spots that congest the game servers and make the overall environment fragmented and unplayable [5]. Finally, a resource allocation service performs dynamic load distribution, balancing, and migration of entities that keep the game servers reasonably loaded such that the real-time QoS requirements are maintained[5].

3.2 Qualitative comparison of proposed SVM-based with neural network-based load prediction

The NN-based scheme [5] bases neural network to foresee future hot-spots to allocate resources for the grid MMOG. However, it did not propose any mechanism to allocate resources and had low prediction accuracy to predict the MMOG load. The above problems motivate us to design a more efficient SVM-based load prediction for MMOGs. We propose an efficient resource allocation mechanism to predict the load level of each game region and allocate resources to each game region accordingly.

The two approaches mentioned above are qualitatively compared in terms of *prediction technique*, *fairness among players*, and *prediction accuracy*, as shown in Table II. The proposed SVM-based load prediction scheme will be described in Chapter 4. In Table II, the prediction technique indicates the operation type of the prediction [5] [16]. The fairness indicates that the same number of VMs is allocated to each game region at first. The prediction accuracy indicates the fraction of prediction correctness for each load prediction scheme.

Table II. Qualitative comparison of the proposed SVM-based with neural network-based load prediction.

Method	Prediction technique	Fairness among players	Prediction Accuracy
SVM-based load prediction (proposed)	SVM	Yes	90% - 95%
Neural network (NN)-based[5] load prediction	Neural network	Yes	82% - 87%

Chapter 4

Proposed SVM-based Load Prediction for Resource Allocation

4.1 SVM-based resource allocation architecture for MMOGs

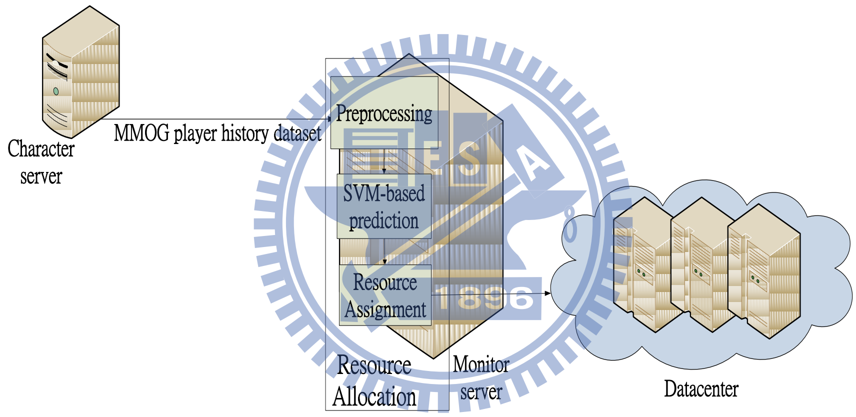


Figure 3: Proposed SVM-based resource allocation architecture for MMOGs.

The proposed SVM-based resource allocation architecture for MMOGs is shown in Figure 3. We obtain MMOG player history datasets from a character server. In a monitor server, there are three main components *preprocessing* module, *SVM-based prediction* module, and *resource assignment* module. The first component, preprocessing module, calculates the loading of each game region at each time interval. The SVM-based load prediction module predicts the loading class of each game region. Finally, the resource assignment module adjusts resources (i.e. number of VMs needed) based on the predicted loading class in each region. With this architecture, the objective is to meet the response time requirement of players in MMOGs. After load prediction, we allocate the resources to each region in the datacenters of the MMOG cloud.

4.2 Preprocessing

MMOG loads are dynamic because of the player interaction [8]. Thus, accurate and fast load prediction is necessary to dynamically allocate resources for MMOGs. In our MMOG cloud system, we obtain MMOG player history datasets from the character server. Figure 4 shows the preprocessing procedure and the MMOG player datasets format. The detailed preprocessing procedure is described as follows:

Preprocessing procedure

1) Read MMOG player history datasets from the character server.
2) Count the number of players (*NP*) in each region at each time interval.
3) Calculate the loading class of each region.

The formula of estimating the loading level of region *j* at time interval *i*, L_j :

$$L_{ij} = NP_{ij} / NV_{ij} \tag{1}$$

where NP_{ij} = total number of players in region *j* at time interval *I* and NV_{ij} = number of VMs used in region *j* at time interval *i*. We define MAX_LOAD $_{\text{vm}}$ = the maximal number of players in a VM. The upper threshold of L_{ij} is T_u and the lower threshold of L_{ij} is T_l, which are defined as follows:

$$T_u = \frac{NV_{ij} \times MAX\_LOAD_{vm}}{NV_{ij}} = MAX\_LOAD_{vm} \tag{2}$$

$$T_l = \frac{(NV_{ij} - 1) \times MAX\_LOAD_{vm} + 1}{NV_{ij}} \tag{3}$$

The loading state of a region can be: *overloading*, *underloading*, or *normal*.

$$\text{Overloading:} \quad L_{ij} > T_u \tag{4}$$

$$\text{Underloading: } L_{ij} < T_l \tag{5}$$

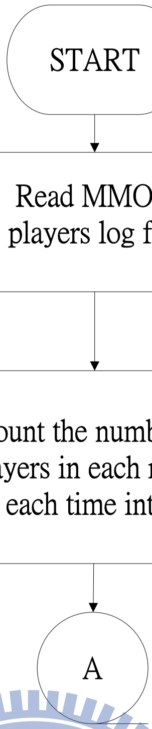
$$\text{Normal: } T_l < L_{ij} < T_u \tag{6}$$

In addition the loading class of a region is either: *class 1* or *class 2*. Class 1 means the loading state is overloading or underloading and has the value +1 in SVM. Class 2 means the loading state is normal and has the value -1 in SVM. The actions taken by each region based on its loading class is as follows:

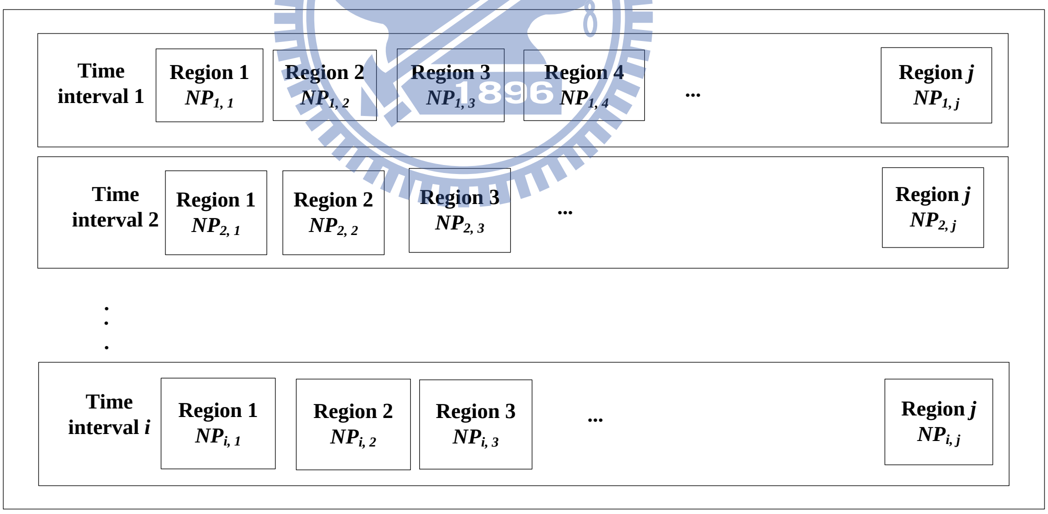
If the loading class of a region is class 1, then the number of VMs in this region should be adjusted,

else if the loading class of a region is class 2, then the number of VMs in this region does not need to be adjusted.

Figure 5 shows the preprocessing procedure for deriving the loading class of each region j at time interval i.



(a): Preprocessing procedure.



(b): MMOG player datasets format for recording the number of players in each region at each time interval.

Figure 4: Preprocessing procedure and the MMOG player datasets format.

.

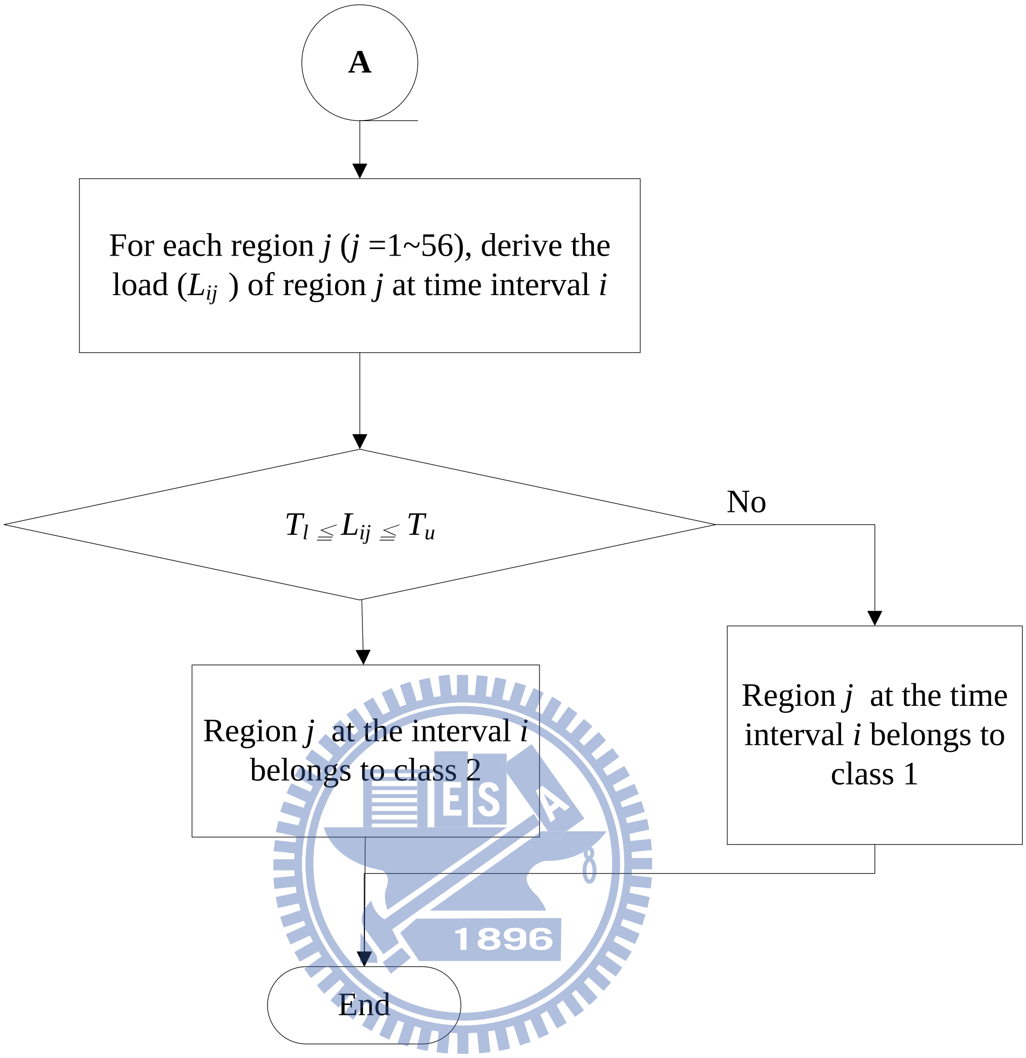


Figure 5: Preprocessing procedure for deriving the loading class of each game region *j* at time interval *i*.

4.3 SVM-based load prediction

In this section, we describe the proposed SVM-based load prediction for MMOG resource allocation. As shown in Figure 6, the SVM-based prediction bases on the loads in previous time intervals to predict the future load of a region. We use SVM-based load prediction to predict the loading class of each region in MMOGs. We first input MMOG player history datasets into the preprocessing module. We can get output datasets which are the predicted loading class of each region. The loading class of each region is then transformed to the SVM format, as shown in section 2.1. Then resources can be assigned to each region based on predicted loading class. The flowchart of the SVM-based prediction and resource assignment is shown in Figure 7. Table III summarizes MMOG load levels and their associated SVM-based loading classes.

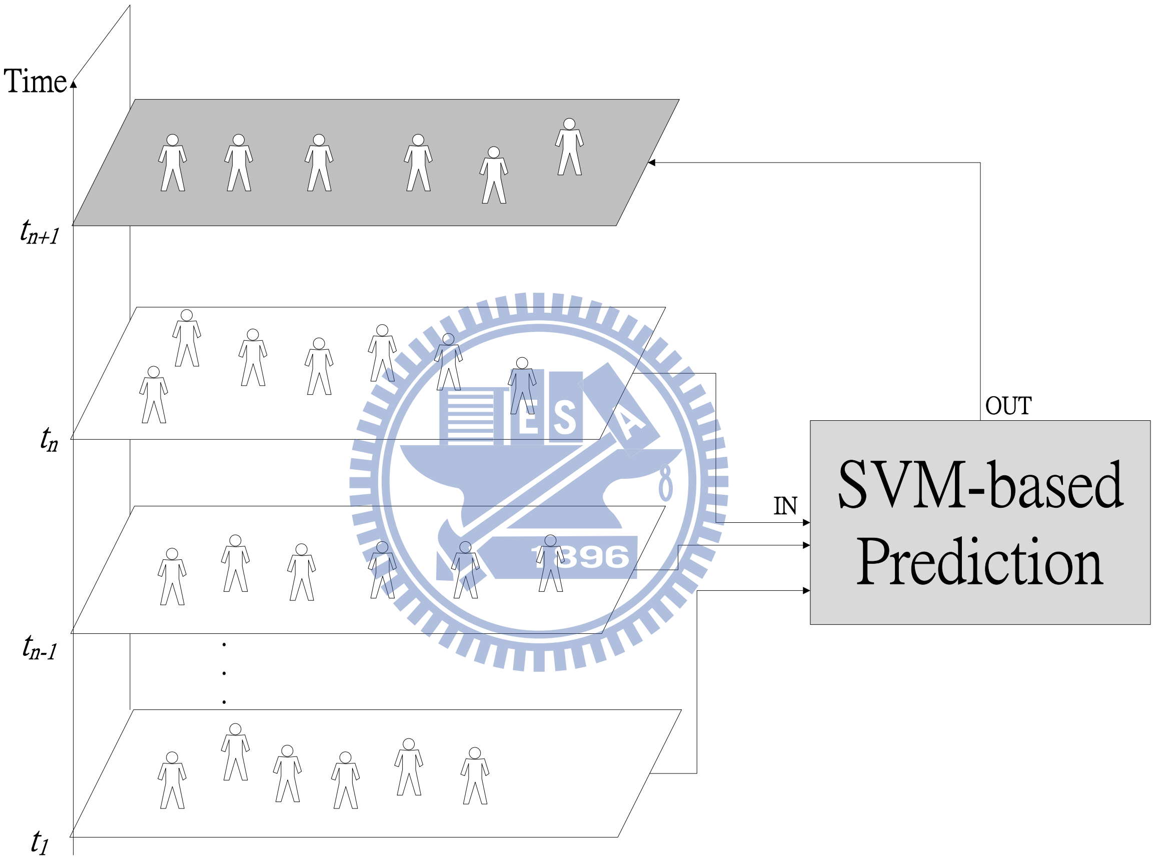


Figure 6: SVM using the loads in previous time intervals to predict the future load of a region.

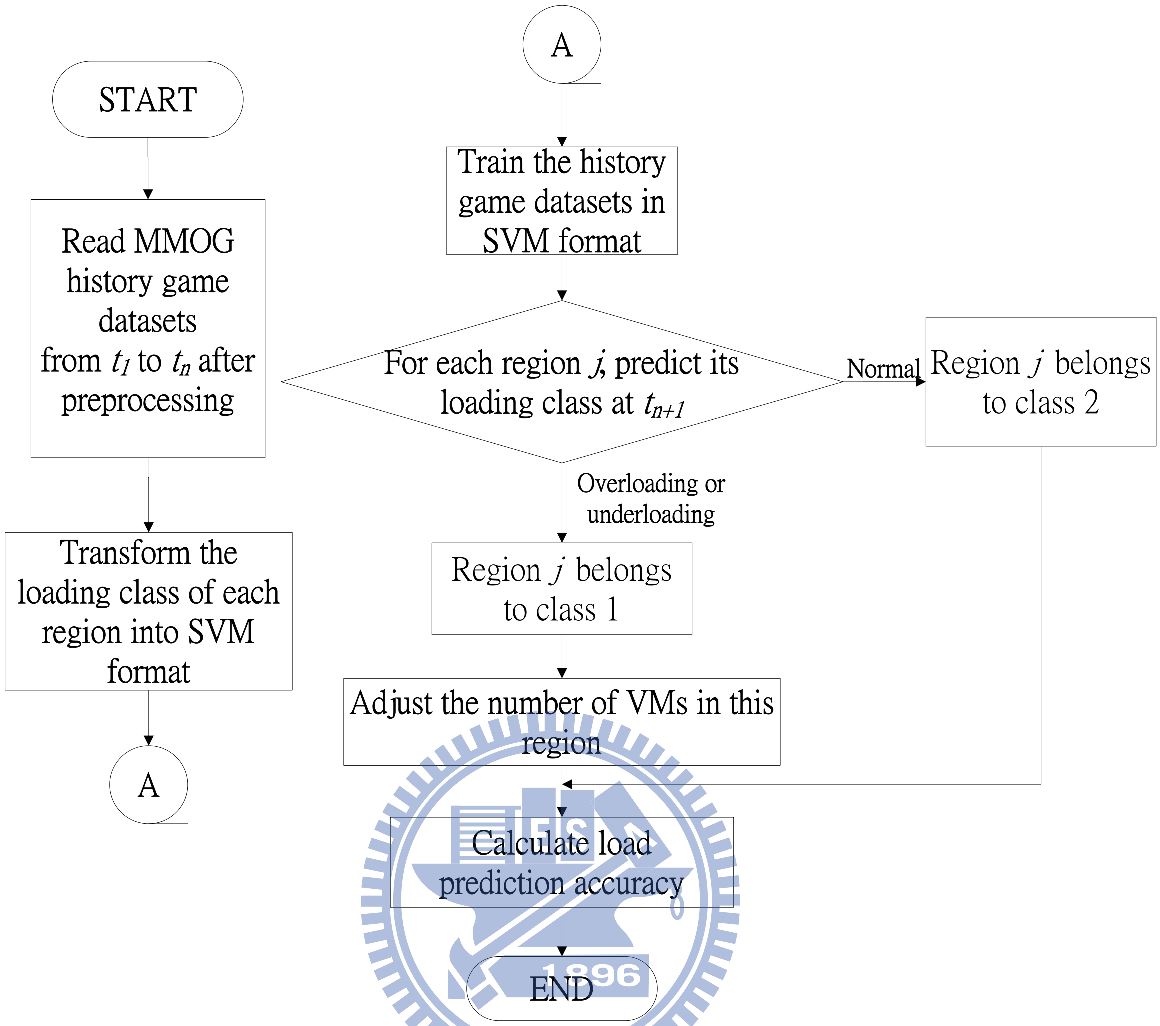


Figure 7: Flowchart of the proposed SVM-based load prediction and resource assignment.

Table III. MMOG loading level and its SVM category.

MMOG load level	Load level range	SVM category	Resource assignment type
Overloading	$L_j > T_u$	Class 1	Adjustable
Normal	$T_l \leq L_j \leq T_u$	Class 2	Unchanged
Underloading	$L_j < T_l$	Class 1	Adjustable

4.4 Resource assignment

Based on the results of load prediction, we propose the rule of resource assignment to add or release VMs in each region. To allocate resources in an MMOG cloud efficiently, we assume that the VMs in the cloud have the same specification. The resource assignment objectives are to meet the QoS, such as the response time requirement of players and to reduce the waste of resources. We adjust the number of VMs used in region j at time interval $i+1$ (NV_{ij}), as follows:

$$NV_{i+1j} = NP_{ij} \,/\, \mathrm{MAX\_LOAD}_{\mathrm{vm}} \tag{7}$$

Note that NP_{ij} is number of players in region j at time interval i and $\mathrm{MAX\_LOAD}_{\mathrm{vm}}$ is the maximum number of players in the VM.

Chapter 5

Performance Evaluation

In this chapter, we first describe simulation setup and evaluation metrics. Then, we compare the proposed SVM-based load prediction with the NN-based load prediction [5] in terms of *prediction accuracy*.

5.1 Simulation setup and evaluation metrics

We ran *libsvm[1]* for SVM-based load prediction. SVM related simulation parameters are shown in Table IV. MMOG running environment setting are as follows:

Table IV. Simulation parameters [1,19].

Parameter	Value
Kernel type	Radius Basic Function
Gamma	125
Degree	2
Scale	-1 ~ 1
Label	1-12 (time interval)
Index	1-56 (region number)

(1) We ran the libsvm [1] software package for MMOG load prediction in Intel 2.67 GHz CPU, 4GB, Windows 7 OS.
(2) MMOG: WoW datasets [3], from Oct. 1 to Oct. 16, 2008.
(3) VM capability: 2000 MIPS CPU, 4 GB RAM, 20 MB/s bandwidth.
(4) Maximum number of players in a VM is 100.

5.1.1 Prediction accuracy

The prediction accuracy is defined as the number of correct game region load predictions divided by the total number of game region load predictions which is expressed as follows:

$$Prediction\ accuracy = \frac{Number\ of\ correct\ game\ region\ load\ predictions}{Total\ number\ of\ game\ region\ load\ predictions} \quad (7)$$

5.2 Comparison between SVM-based and NN-based load prediction

As shown in Figure 8, the proposed SVM-based load prediction scheme achieves 12.24% and 26.77% better prediction accuracy than the NN-based load prediction and the last value-based load prediction, respectively. The proposed SVM-based load prediction reduces 8% and 78.3% of the number of VMs needed compared with the neural network-based load prediction and the last value-based load prediction, respectively.

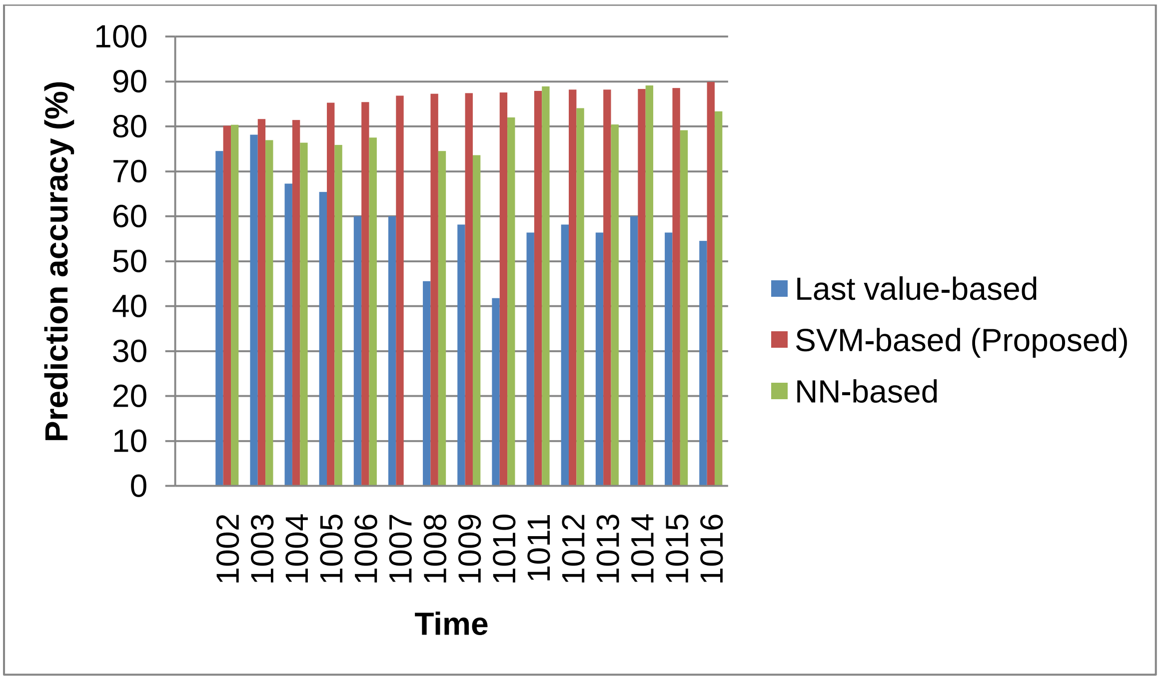


Figure 8: Prediction accuracy comparison of the three load prediction schemes.

Chapter 6

Conclusion

6.1 Concluding remarks

In this thesis, we have presented an efficient SVM-based load prediction scheme for resource allocation in MMOG clouds. We use historic load data of players in each region to predict the future load. By using the load prediction results, we can allocate (or release) an appropriate number of virtual machines to each game region in advance in an MMOG cloud so as to meet the response time requirement of each player and to have better resource utilization in MMOG clouds. The load prediction accuracy of the proposed SVM-based scheme is 12.24% better than that of the NN-based scheme. In addition, the proposed

SVM-based load prediction reduces 8% of the number of VMs used compared with the neural network-based load prediction. With a higher prediction accuracy and a smaller number of VMs used, our SVM-based load prediction is feasible for efficient resource allocation in MMOG clouds.

6.2 Future work

We will integrate the proposed SVM load prediction scheme to an efficient resource allocation method for MMOG clouds. In addition, we will combine a fuzzy membership function to each input datum of SVM and design a fuzzy-based SVM load prediction scheme to further increase prediction accuracy.

Bibliography

[1] *Libsvm*, software available at http://www.csie.ntu.edu.tw/ cjlin/libsvm, 2001.

[2] C. Hsu, C. Chang, and C. Lin, “A practical guide to support vector classication,” http://www.csie.ntu.edu.tw/~cjlin/papers/guide/guide.pdf, 2003.

[3] Y. Lee, K. Chen, Y. Cheng, and C. Lei, "World of warcraft avatar history dataset," in the *ACM Multimedia Systems*, Feb. 2011.

[4] K. W, “Resource allocation for massively multiplayer online games using fuzzy linear assignment technique,” in *Proc. of 5th IEEE Consumer Communications and Networking Conference*, pp. 1035 – 1039, Jan. 2008.

[5] R. Prodan and V. Nae, “Prediction-based real-time resource provisioning for massively multiplayer online games,” in the *Future Generation Computer Systems*, Vol. 22, Issue 3, pp. 785-793, July. 2009.

[6] Z. Wan and X. Xu. “An analysis of WoW players’ game hours,” in *Proc. ACM SIGCOMM Workshop on Network and System Support for Games*, pp. 47-52, 2008.

[7] R. Suselbeck, G. Schiele, and C. Becker. “Peer-to-peer support for low-latency massively multiplayer online games in the cloud,” in *Proc. Network and Systems Support for Games*, pp. 1-2, Apr. 2009.

[8] V. Nae, A. Iosup, and R. Prodan, “Dynamic resource provisioning in massively multiplayer online games,” in the *IEEE Transactions* on *Parallel and Distributed Systems*, Vol. 22, Issue 3, pp. 380-395, Jan. 2011.

[9] http://www.support-vector-machines.org/ .

[10] http://www.cs.nccu.edu.tw/~lien/VOIP/BBQresrc/ .

[11] B. Hayes. “Cloud computing,” in the *Communications of the ACM*, Vol. 51, Issue 7 , pp. 9-11, July 2008.

[12] H. Thorsten, B. Thomas, and H. Robert, “A peer-to-peer architecture for massive multiplayer online games.” in *Proc. ACM SIGCOMM Workshop on Network and System Support for Games*, Oct. 2006.

[13] S. Zhi, L. Yang, and Y. Tao. “Short-term load forecasting based on LS-SVM optimized by bacterial colony chemotaxis algorithm,” in *Proc. International Conference on Information and Multimedia Technology*, pp. 306-309, Dec. 2009.

[14] M. Mohandes , “Support vector machines for short-term electrical load forecasting.” *in the International Journal of Energy Research*, pp. 335-345, 2002.

[15] http://en.wikipedia.org/wiki/Neural_network.

[16] P. Marco, “Peer-to-peer architecture for real-time strategy MMOGs with intelligent cheater detection,” in *Proc. International ICST Conference on Simulation Tools and Techniques*, pp. 1-8, Mar. 2010.

[17] Y. Li and T. Fang, “Application of fuzzy support vector machines in short-term load forecasting,” in *Proc. International Conference on Rough Sets, Fuzzy Sets, Data Mining, and Granular computing*, pp. 363-367, 2003.

[18] M. Javier and A. Muñoz, “Support vector machines with applications,” in *Statist. Sci*, Vol. 21, pp. 322-336, Dec. 2006.

[19] L. Gang, et al. “Short-term load forecasting using support vector machine with SCE-UA algorithm,” in *Proc. International Conference* on *Natural Computation*, pp. 290-294, Nov.2007.