논문 2008-45CI-1-5

DVE에서 시간 기반 균등 부하 분산 방식 설계

(Design the Time-Interval Based Fairness Partitioning Method in DVE)

원 동 기*, 안 동 언*, 정 성 종*

(Dongkee Won, Dongun An, and Seungjong Chung)

요 약

MMORPG 게임 에서는 많은 사용자들이 다수의 서버를 통해 관리 된다. 이러한 환경에서 사용자 수는 몇 명에서 몇 천명에 달할 수 있고 사용자들은 같은 가상 세계에서 서로의 정보를 주고받는 과정에서 여러 가지 문제들, 서버와의 연결 지연이나 끊김 현상 등이 발생 할 수 있다. 이러한 문제는 사용자들이 게임을 즐기는데 있어서 큰 방해가 되며 위와 같은 문제로 인한 명령어 전달이나 메시지 전달이 늦게 되면 사용자는 가상 세계에서 불이익을 받을 수 있다. 많은 정책들이 이와 같은 문제들을 해결하기 위해 자원 여유에 따른 등급에 따라 서버들에 사용자를 배분 하고 있지만 불행히도, 최적의 환경으로 사용자를 서버에 배분 하였다 하더라도 급변하는 서버와 네트워크의 상태 변화로 인해 시간이 지남에 따라 차선책이 되고 만다. 그로 인해 MMORPG 환경에서 원활한 서버 자원 관리가 힘들어 진다. 이러한 문제들을 해결하기 위해 시간 기반의 균등 분할 방법(TIP)를 제안한다. TIP는 게임 유저들을 게임 서버들에 특정 시간마다 균등하게 분할하여 부하 불균형으로 인한 문제점을 해결할 것이다.

Abstract

MMORPGs may involve a great number of concurrent players, and those servers usually have to manage hundred, or even thousands of avatars co-existing in the same virtual world. So if failing to send a command or an event message, or sending it too late may cause damages to the avatar evolution(death, injury, loss of resources), and may result in unjustified penalties for the player. Many policies could be defined to realize a ranking evaluation of available servers. Unfortunately, due to the highly dynamic characteristics of server loads and network performances, any optimal allocation would soon become sub-optimal. In order to solve those problems we propose the "time-interval based fairness partitioning method" (TIP). TIP will distribute the avatar to the game server equally with time-interval in order to avoid the problems form the unfairness of game servers load.

Keywords: DVE, MMORPG, TIP

I. Introduction

Recently, the presence of computer science growing up rapidly with communication technology, this becomes the crucial symbol of high speed development about global information technology and industry. Especially, the flying development of network technology makes an essential transformation of information application system and one of representative instances is Distributed Virtual Environment (DVE)[1]. DVE has been used in many fields, such as collaborative design, civil and military distributed training, e-learning[2] or multi-player games[3~5]. These systems make multiple users and multiple virtual environments connected via internet to share the information and interact together where users working on different computers which are interconnected through different networks. In DVE, each user is represented in the shared virtual environment by an entity called avatar, whose state

* 정회원, 전북대학교 컴퓨터공학과
(Department of Computer Eng., Chonbuk National University)
접수일자: 2007년11월16일, 수정완료일: 2008년1월11일

is controlled by the user input. Since DVE supports visual interactions between multiple avatars, every change in each avatar must be propagated to the rest of the avatars in the shared virtual environment.

One of crucial issues in design of a scalable and efficient DVE system is the partitioning problem[9]. It consists of efficiently assigning the workload (avatar) among the servers in the system[6]. The partitioning problem is considered as a key of designing DVE, because it directly determines overall performance of DVE, and it not only affects the workload which each avatar generated, but also affects the communication cost among different inter-servers. Some methods for solving the partitioning problem have already been mentioned in such reference paper [7, 9]. Besides, there are still some improvements comprising the ad hoc search method which is proposed in[8] and several heuristic search methods [11]. The aim of all kinds of methods is to relate the quality function used in the search method with DVE system performance. Otherwise, the proposed method could provide inefficient solutions for the partitioning problem.

Ⅱ. Related work

1. Related concepts

가. Avatar

Under the DVE, we usually use an avatar, which is a 3D active object, to represent a DVE client in the virtual world. In order to provide the interactive capability of a client, the avatar can move or travel in the virtual world[9].

나. AOI

In general, each avatar only needs to know the activity that happened near his or her neighbors[9]. AOI is the region of the virtual world that if there is any activity happened in this region, the avatar needs to update its own state and make his or her view consistent.

다.Quality Function CP

MSDVE is a general trend on designing DVE systems[9]. The massage generated by avatar in MSDVE can be divided into two kinds: intra-server massage and inter-server messages. Thus, in order to design a scalable DVE, we would better minimize the number of inter-server messages. An excellent DVE should include more intra-server messages and fewer inter-server messages.

The partitioning problem can be considered as efficiently distributing the workload (assigning avatars) among the different servers in the system . Lui and Chan developed an efficient approach for finding a good assignment of avatars to servers in order to ensure both a good assignment of avatars which generate less workload and the minimum network traffic in DVE systems[12, 16]. They propose a kind of quality function named as Cp[7]. Such quality function Cp could evaluate each partitioning strategy (assignment of avatars). This quality function includes two parts. One of function is named as C_P^W which in order to compute the workload generated by clients in the DVE system. The other part of the quality function is named as C_P^L which in order to evaluate the overall number of inter-server messages. The quality function Cp is defined[7] as

$$C_{\mathrm{p}} = W_1 C_P^W + W_2 C_P^L$$

Where $W_1 + W_2 = 1$. W_1 and W_2 are two coefficients that weight the relative importance of the computational and communication workload, respectively. W_1 and W_2 are tuned values. Different DVE has different values of W_1 and W_2. Adopting this quality function (they assume that $W_1 = W_2 = 0.5$) Lui and Chan propose a technique that the clients can be dynamically re-assigned to servers after initial partition[7]. The technique should be periodically executed for adapting the current state of partition. The DVE are on demand as clients can join or leave the DVE system at any moment, and they can also move everywhere within the simulated

virtual world. Lui and Chan also have proposed a testing platform for the performance evaluation of DVE systems, as well as a parallelization of the partitioning algorithm[7].

라.Partitioning method

In order to obtain an excellent partitioning method, we need a low Cp value. Thus, the value of Cp becomes a evaluation factor of a partitioning method. Generally, there are two kinds of partitioning methods: Ad hoc partitioning method and heuristic partitioning method. One of famous ad hoc partitioning methods is Linear Optimization Technique (LOT) and RBP LP CRP which was initially published by Lui and Chan in[9] and revisited in[7] from their ideas published in[10] about graph theory[7]. Besides ad hoc partitioning method, here are also some heuristic methods[8, 11] which show us an excellent result for partitioning problem in DVE. We will compare TIP with RBP because RBP is general initial partitioning method.

2. General problems

MMORPGs may involve a great number of concurrent players. While in other games, like strategic and first-person-shooters, the maximum number of interacting players seldom exceeds a dozen [14] MORPG servers usually have to manage hundreds, or even thousands of avatars co-existing in the same virtual world[13].

The gaming infrastructure is realized by the set of servers, the distributed client applications, the interaction rules and game management protocols, and the interconnecting communication network infrastructure. The challenging and attractive set of today's multiplayer games lets the users to live and act by pro-actively generating local and personal action, which any affect every other user within the game context.

The main problem faced in this work is the heterogeneity and dynamically in the latency that could exist on the Internet connections among many players connected to the distributed multi-servers architecture supporting the game execution.

Many policies[7~11] could be defined to realize a ranking evaluation of available servers. Unfortunately, due to the highly dynamic characteristics of server loads and network performances, any optimal initial allocation would soon become sub-optimal.

As an example, network loads, links utilization and router congestion over generalized multi-purpose networks and subject to fast and unpredictable changes, so that any optimal choice at time t may became suboptimal after few seconds. In practice, the proposal is to let the client, being connected to one server, to the "most performance" server evaluated at runtime. This should be evaluated on the basic of server-measurement metrics and heuristic polices.

For this reason, the unfairness problem frequently leads to a generalized users' annoyance: slow users get frustrated because they lose easily despite their skills, while fast (and fair) users get bored because they win easily thanks to their network performances.

In order to avoid such phenomenon of system saturation, we design a new partitioning method which considers quality function Cp with time-interval and in order to maintain of game server's Cp fairly and to avoid imbalance of those Cp.

III. Time-Interval Partitioning Method

1.Method description

We propose a new partitioning method named TIP (Time-Interval based Partitioning Method) for the aim that a method can avoid DVE system saturation and keep off such kinds of situations that unfairness of game servers. The main idea of this method is allocating the border avatar to a candidate server with low imbalanced cost every interval time.

Define as Simulation Time (ST) the time concept considered for synchronization issues in the distributed simulation. And assign SN to Number of Server and S assign to Server. During each ST step a client can perform at most one move, that is, a

single command can be sent from the client to the server with this system. Assumption, can be controlled the maximum speed of every player under the fairness of Cp.

The TIP is based on graphic theory, in other words, an initial partition must be needed which divides DVE world into several regions initially[12]. After initial partition, each avatar has an allocated server and each server can manage many regions and many avatars in these regions. In our implementation, we adopt Recursive Bisection Partition (RBP) as our initial partition first.

The first step in TIP method is to select the subset of border avatars from the set of all avatars in the system. A given avatar is selected as a border avatar if it is assigned to a certain server S in the initial partition and any of the avatars in its AOI (Area Of Interest) is assigned to a server different from it's server.

For each of the border avatars, a list of candidate servers is constructed which can be considered as the second step of TIP. This list contains all of the different servers that the avatars in the same AOI are assigned to (including the server that the avatar is currently assigned). At that time, each element in candidate server list of a border avatar will obtain a Cp value which is computed for evaluating the system cost if this border avatar is re-assigned into this server. Because of the requirement of DVE, the less Cp value which server possesses, the better re-assignment solution it is.

The last step is selecting the server with the lowest value of ev in ranked server list. During this process, we set an evaluation function (1) for calculating an evaluation value (ev) of fairness of game servers in ST. The server with the lowest value of ev is the allocating server for relative border avatar.

$$ev = \sum_{T}^{ST}\left\{\sum_{n=0}^{SN}(C_p(S_n)) - \sum_{n=0}^{SN}(C_p(S_n)/SN)\right\} \quad (1)$$

After selects the final allocated server for a border avatar, TIP will continue working for the next time-interval after specific ST, TIP will work again during the interval-time.

2.Experiment

In this section, we present experiment about the TIP which we discussed in the previous sections and apply the TIP on both a small and a large scale virtual world.

We want to simulate MMORPG networks with thousands clients connected at the same time, where each client communicates whit at least one server and each server communicates with every other server so we implemented, simulator written in java language.

The dimension of a small virtual world is 8×8 units. The total number of avatars in this world is equal to 100 and the number of servers is equal to 4. The dimension of a large virtual world is 64×64 units. The total number of avatars in this world is equal to 2500 and the number of server is equal to 16. Some parameters which are mentioned about evaluation function for selecting candidate server in the previous section also have been tuned. Table 1 shows some parameters as follow

For comparison purposes, we have simulated the proposed TIP and RBP[7] these two techniques.

There is no interval time in RBP but in order to compare with TIP, we define the interval time for RBP as table 1.

표 1. 실험에 사용된 파라미터
Table 1. Related Parameters.

	Small Virtual world	Large virtual world
Amount of avatars	100	2500
Amount of servers	4	16
Time-interval	30sec	
interval for RBP	150sec	
ST	600sec	
Avatar distribution	Uniform Distribution	

3. Evaluation result

Fig 1. and Fig 2. show implementation evaluation results a small virtual world. The map has been performed initial distribution of uniform.

From the Fig1, we can find that both of RBP and TIP has low ev. It can be observed that the value ev of each method in DVE almost same data.

We also performed two partitioning methods on large DVE.

Next, we can see the Fig 2. which show implementation evaluation results which are obtained from a large virtual world after implemented RBP and TIP method. Both of RBP and TIP have been performed initial distribution of uniform

From the presented data, it can be seen that the value of ev provided by the TIP method is lower than the RBP method. TIP prevents an increase of the ev for performing the re-allocate the border avatar by time-interval. To perform the RBP by periods also prevents a increase the ev but it is expected that performing the RBP very often could cause the overhead of game serves then TIP Because TIP is performing only border avatar but RBP performing every avatar in the game servers.

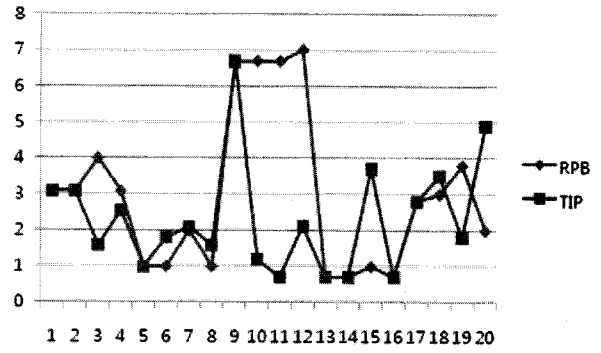


그림 1. Small world에서 RPB와 TIP 의 결과
Fig. 1. Result RBP and TIP in the small world.

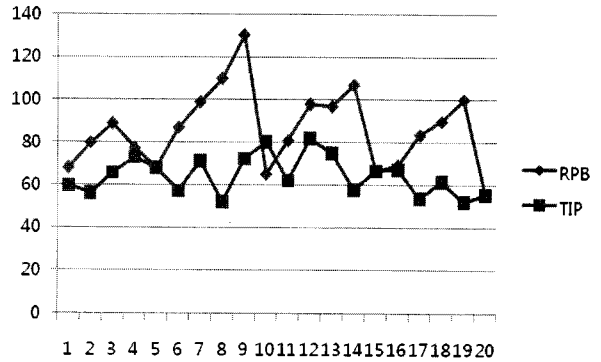


그림 2. Large world에서 RPB와 TIP 의 결과
Fig. 2. Result RBP and TIP in the large world.

IV. Conclusion and future work

In this paper, we focus on the field of partitioning method which as one of efficient issues on improving performance of DVE. Generally, the partitioning methods include two kinds: one is ad hoc partitioning method such as LOT; the other is heuristic partitioning method such as Ant Colony System (ACS), and so on. According to the blank of considering CPU utilization of server in DVE among those partitioning methods, we purposed a new partitioning method TIP in order to prevent DVE from saturation. We simulated the proposed partitioning method TIP and also compared with the other method Recursive Bisection Partition (RBP) both on a small virtual world and a large virtual world. We performed uniform distribution as avatar distribution. The results show us that TIP method presents a balancing state of participant servers in DVE and keeps away from system saturation. The method TIP could select a suitable allocated server to each border avatar based on the aim of balancing server ev in DVE.

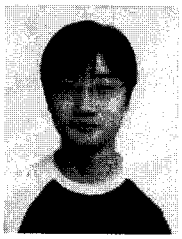
As our future work, we propose to perform the similar simulations based on other different avatar distributions. And as some parameters in evaluation function of TIP method should be tuned, since now, we are still looking for the best value for the TIP.

References

[1] M.Ye,LCheng, "System-performance modeling for massively multiplayer online role-playing games" IBM System Journal, vol 45, NO1, 2006.
[2] Tohei Nitta, Kazuhiro Fujita, Sachio Cono, "An Application Of Distributed Virtual Environment To Foreign Language", in IEEE Education Society, October 2000.
[3] Michael Lewis and Jeffrey Jacboson, "Game

Engines in Scientific Research", in Communication of the ACM, 45(1), January 2002.
[4] L. Zou, M. Ammar, C. Diot, "An Evaluation of Grouping Techniques for State Dissemination in Networked Multi-User Games", in Proceedings of the Ninth International Symposium on Modelling, Analysis and Simulation of Computer and Telecommunication Systems, (MASCOTS'01), August 2001.
[5] J. Smed, T, Kaukoranta and H. Hakonen, "A Review on Networking and Multiplayer Computer Games", Tech Report 454, Turku Centre for Computer Science, April 2002.
[9] G. Huang, M. Ye, and L. Cheng, "Modeling System Performance in MMORPGs," Proceedings of the 1st IEEE International Workshop on Networking Issues in Multimedia Entertainment NIME'04 at IEEE GlobeCom Entertainment NIME'04 at IEEE GlobeCom 2004.
[6] P. Morillo, J.M. Orduna, J. Duato, et al "Improving the Performance of Distributed Virtual Environment Systems", In IEEE Transaction on Parallel and Distributed Systems, 16(7) , July 2005.
[7] J. C. S. Lui, M. F. Chan, K. Y. Oldfield, "An efficient partitioning algorithm for distributed virtual environment system", IEEE Transactions on Parallel and Distributed System 13, 2002.
[8] P. Morillo, J.M. Orduna, J. Duato, et al "A comparison study of evaluative algorithms for solving the partitioning problem in distributed virtual environment systems" in Parallel Computing 30, 2004, 586-610.
[9] C. Greenhalgh, A. Bullock, E. Frecon, D. Llyod, and A. Steed, "Making Networked Virtual Environments Work" presence, vol.10, no. 2, Apr. 2001.
[10] J.C.S.Lui, W.K. Lam, "General methodology in analyzing the performance of parallel/distributed simulation under general computational graphs", in: Third International Conference on the Numerical Solution of Markov Chain, September 1999, 6-10.
[11] P. Morillo, M. Fernandez, J. M. Orduna, "A Comparison Study of Modern Heuristics for Solving the Partitioning Problem in Distributed Virtual Environment Systems", in: Proceedings of International Conference on Computational Science and its applications (ICCSA' 2003), Lecture Notes in Computer Science, 2669, Montreal, Canada, May 2003.
[12] Dongkee Won, Beobkyun Kim, Tingting Li, and so on, "Dynamic Map Partitioning with predefined map information," Pro. The 14th KSII fall conference, Korea, November, 2006, 525-528.
[13] H. Engum, J.V. Iversen, and O. Rein. Zereal: "A semi-realistic simulator of massively multiplayer online games," http://citeseer.nj.nec.com/555870.html.
[14] W Feng, F. Chang, W. Feng, and J. Walpole. "Provisioning on-line games: a traffic analysis of a busy counter-strike server," In Proc. of the second ACM SIGCOMM Workshop on Internet measurement, 151 - 156. ACM Press, 2002.

저 자 소 개



원 동 기(정회원)

2003년 전북대학교 컴퓨터과학과 공학사

2006년 전북대학교 컴퓨터공학과 공학석사

2007년~현재 전북대학교 컴퓨터공학과 박사 과정

<주관심분야 : DVE, 분산 시스템, 그리드 컴퓨팅>

정 성 종(정회원)

1975년 한양대학교 전기공학과 공학사

1981년 휴스턴대학교 전자공학과 공학석사

1988년 충남대학교 전자공학과 공학박사

1985년~현재 전북대학교 컴퓨터공학과 교수

<주관심분야 : 인터넷 응용기술, 지능 공학>

안 동 언(정회원)

1981년 한양대학교 전자공학과 공학사

1987년 한국과학기술원 전산학과 공학석사

1995년 한국과학기술원 전삭학과 공학 박사

1995년~현재 전북대학교 컴퓨터공학과 교수

<주관심분야 : 정보검색, 한국어 정보처리>