Journal of Network and Computer Applications 40 (2014) 21–30

Contents lists available at ScienceDirect

Journal of Network and Computer Applications

journal homepage: www.elsevier.com/locate/jnca

Reducing view inconsistency by predicting avatars' motion in multi-server distributed virtual environments

Haiyang Hu [a,*], Yizhi Ren [b,**], Xu Xu [a], Liguo Huang [c], Hua Hu [a]

[a] School of Computer Science and Technology, Hangzhou Dianzi University, Hangzhou 310018, China
[b] School of Software Engineering, Hangzhou Dianzi University, Hangzhou 310018, China
[c] Department of Computer Science and Engineering, Southern Methodist University, Dallas, TX 75275-0122, USA

ARTICLE INFO

Article history:
Received 7 January 2013
Received in revised form
16 June 2013
Accepted 5 August 2013
Available online 8 September 2013

Keywords:
Distributed virtual environments
View consistency
Probability models
Virtual avatars
Density-based clustering

ABSTRACT

Multi-server distributed virtual environment (DVE) system contains a set of servers to support a huge number of geographically distributed users. It provides a shared virtual scenario where users represented by avatars participate in and interact with each other through networks. However, there are two technical challenges in designing an efficient DVE system with this architecture: view inconsistency caused by network delays and server overloading caused by uneven distribution of users. While the first problem affects users' perception of the VE and causes user disputes, the second problem affects the system response time. We present a new partitioning framework to simultaneously address the two problems, in which the DVE scenario is divided into a set of disjoint grids called virtual cells. The virtual cells are then indexed by a region quadtree and an algorithm is proposed to search the region quadtree so as to equally partition avatars among servers. By using probabilistic models to predict each avatar's future motion and the virtual region its area of interest (AOI) will cover, a density-based method is proposed to help each server construct avatar clusters from its local avatars in parallel. By estimating the possible distances of avatar's future position to the geometry center of each cluster, an algorithm is proposed to repartition non-assigned avatars among servers so as to reduce view inconsistency and preserve workload balancing among the servers. We conducted comprehensive experiments to evaluate our method by comparing it with other approaches.

1. Introduction

With the proliferation of network, computer graphics, and distributed systems technologies, the management of distributed virtual environment (DVE) systems (Lui and Chan, 2002; Singhal and Zyda, 1999; Zhou et al., 2004; Bouras and Tsiatsos, 1994) has attracted considerable attention in various different applications, such as online training (Singhal and Zyda, 1999), collaborative design (Dias et al., 1997), e-learning (Bouras et al., 1998) and, most important of all, multi-player online games (MOG) (Sheua et al., 2009) over the last decade. A DVE system allows geographically distributed users represented by an entity called avatar to move around, interact with each other, and inquire information and states of relevant objects within a share virtual environment (VE) through networks, in particular the Internet.

There are two popular architectures to support DVE systems, peer-to-peer and client–server (Singhal and Zyda, 1999; Sheua et al., 2009; Frecon, 1998; Diot and Guatier, 1999). The peer-to-peer architecture has a relatively lower overall latency due to the direct connections among relevant peers. On the other hand, the main advantages of the client–server architecture are that the server may also serve as a monitor to enforce security and provide information persistency. It is these advantages that make the client–server architecture more popular in commercial DVE systems, such as online games. With the client–server architecture, more servers may be added as the number of users increases.

Whenever an avatar moves in the VE, an update message is needed to be sent to all relevant clients. To reduce the network bandwidth consumption, and message processing overheads, the area of interest (AOI) (Morillo and Orduna, 2005; Morillo et al., 2007; Singhal and Zyda, 1999) may be used to determine the set of relevant clients. For example, if visual information is considered, the motion of an avatar will only affect other avatars which are within a certain distance from it. Hence, we may define an AOI such that whenever this avatar makes a move, we only need to send the state update message to the clients of the avatars which are now within this AOI.

In designing a large scale DVE system that is scalable and cost-effective, we are faced with many technology challenges. First, since multiple users may participate in the VE, it is important for the system to keep all avatars within the VE in a "view consistent" state, i.e., if an avatar makes a move or changes its state in the VE, all relevant avatars should be aware of the change timely. Second, to provide high throughput and low response time for processing queries submitted by avatars, a large scale DVE system should have

* Corresponding author. Tel.: +86 13429129739.
** Co-corresponding author.
E-mail addresses: huhaiyang@hdu.edu.cn, sanse1977@163.com (H. Hu), renyizhi@hdu.edu.cn (Y. Ren).

1084-8045/$ - see front matter © 2013 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.jnca.2013.08.011

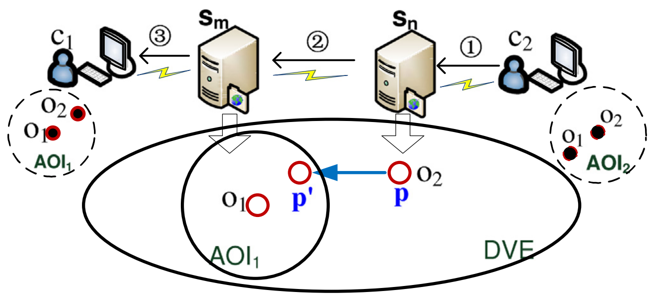

Fig. 1. View inconsistency in online game adopting multi-server architecture.

the ability to efficiently assign the workload among the servers so as to improve the overall performance of the system. This implies that some partitioning algorithm is needed to partition the workload among the servers in an approximately equal manner. As a result, how to meet the requirements for view consistency and load balancing in a VE is crucial in designing a scalable and cost-effective DVE system.

In a large scale DVE adopting multi-server architecture like massively multi-player online games (e.g., Asherons Call, 2013), each client/user has a corresponding avatar in the game and each server manages a subset of clients and maintains the states of their avatars. As shown in Fig. 1, avatar o_1 with its client c_1 is managed by server s_m and o_2 with its client c_2 is managed by s_n. Now, o_2 moves into the AOI of o_1. Its client c_2 immediately sends s_n a state update message indicating its new position. s_n finds that o_1 will be influenced and it immediately sends a sync message to s_m. Upon receiving the message, s_m notifies o_1 with client c_1 to update o_2's position. Now, o_1 can see o_2 at position p' in its AOI. In this way, the total delay from the moment that o_2 starts to move till the moment that o_1 is aware of this change is the sum of the intra-server communication delay between o_2 (i.e., c_2) and s_n, the inter-server communication delay between s_m and s_n, and the intra-server communication delay between o_1 (i.e., c_1) and s_m. During the period of the communication delay, o_1 cannot be aware of o_2's correct location in its AOI. On the other hand, if the two avatars are managed by one server (e.g., s_m), the delay will only contain the intra-server communication delay. In this way, to reduce the amount of view inconsistency in the game, it may need the system to provide an approach to estimate the avatar's possible moving behavior and partition those avatars that may meet with each other to one server while preserving workload balancing among the servers at the same time.

A few methods have been proposed for assigning the avatars to different servers in order to achieve one of the two goals, load balancing or view consistency (Lui and Chan, 2002; Morillo and Orduna, 2005; Morillo et al., 2007; Tang and Zhou, 2010; Zhang and Tang, 2011, 2012). As these methods adopt a partitioning approach based on avatars' static positions before they make movements, if a given avatar makes a move or changes its behavior, the server that it is connected/assigned to will distribute its state to all relevant servers that handle those avatars within its AOI. In the worst case, if each avatar within its AOI is assigned to a different server due to their movements, then the state update from any one of these avatars has to be sent by its server to all other servers. As such, these partitioning methods may give rise to an increase in the amount of network traffic among the servers, causing a longer response time to the avatars. In addition, if the servers are located at different geographical locations, the delay among these servers may be high, causing view inconsistency among the avatars across the servers. Although reducing the view inconsistency is very important in designing an effective DVE system, there is only limited work in this area.

This paper presents a partitioning approach by predicting avatars' future motions so as to reducing the view inconsistency between avatars in the context of maintaining workload balancing among servers. In the approach, avatars are first equally assigned to servers based on their static neighboring positions in the DVE. Servers construct avatar clusters from its assigned avatars by considering the possible virtual region that avatars' AOI may cover in the coming time frames. The avatars not belonged to any cluster are then exchanged among the servers based on their distances to the clusters. In summary, the contributions of this paper are as follows:

- We present a new partitioning approach for jointly addressing the two issues, view consistency and workload balancing by predicting avatars' motions through probabilistic models.
- We index the virtual environment by a region quadtree and propose an algorithm called *QD_InitialPartition* to equally partition avatars at neighboring positions among servers so as to average the computational workload of each server.
- We propose a density-based method called *Parallel_UMCluster* to help each server construct avatar clusters from its local avatars in parallel by calculating each avatar's future motion and the virtual region its AOI may cover through probabilistic models.
- We propose an algorithm called *LB_Repartition* to repartition non-assigned avatars among servers so as to reduce view inconsistency and preserve workload balancing in the DVE, which is based on the distances of avatar's possible position to the geometry center of each cluster in the coming time frames.

The rest of the paper is organized as follows. Section 2 briefly reviews related works. Section 3 introduces the problem of view inconsistency caused by avatars' movements and network delay. Section 4 presents probabilistic models for analyzing avatars' future movements. Three algorithms are also presented in this section for assigning avatars to servers so as to achieve the goal of reducing view inconsistency among avatars and maintaining workload balancing among servers at the same time. Section 5 discusses a number of experiments to evaluate our method by comparing it with other approaches. Finally, Section 6 briefly concludes the work presented in this paper.

2. Related works

To meet the requirement for load balancing among servers in a DVE system, several partitioning algorithms are presented. In Lui and Chan (2002), the authors presented an efficient partitioning algorithm based on linear optimization to divide the workload of a large scale DVE system evenly among the servers and reduce the inter-server communication. In the designing the algorithm, two parameters were taken into account, one measuring how much a partitioning strategy deviated from the ideal load balancing partitioning strategy, and the other measuring the overall inter-server communication cost. The authors also presented a parallel version of the partitioning algorithm.

In Morillo et al. (2007), the authors presented a partitioning algorithm based on heuristic search to meet the requirement of reducing latency below some given threshold provided in DVE system for the avatars. In the algorithm, they used the aggregated CPU bandwidth of the servers to avoid system saturation and further used the remaining CPU bandwidth to divide avatars surrounded with the greatest number of neighbors in the same server, thus, the latency provided to the maximum number of avatars can be decreased.

In Morillo et al. (2004), the authors considered the non-linear state of DVE servers with the number of its assigned avatars, and presented a fine-grain partitioning method. The method contained