Journal of Network and Computer Applications 41 (2014) 263–273

Contents lists available at ScienceDirect

Journal of Network and Computer Applications

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

Update schedules for improving consistency in multi-server distributed virtual environments

Yusen Li, Wentong Cai

Parallel and Distributed Computing Center, School of Computer Engineering, Nanyang Technological University, Singapore 639798, Singapore

ARTICLE INFO

Article history:
Received 24 June 2013
Received in revised form
23 December 2013
Accepted 7 January 2014
Available online 25 January 2014

Keywords:
Distributed virtual environment
Time–space consistency
State update
Optimization
Scheduling

ABSTRACT

Distributed Virtual Environments (DVEs) have been widely used in online games, military training and collaborative engineering applications, etc. In DVEs, the primary task is to maintain a consistent view of the virtual world among all users. Multi-server architecture has been shown to have good scalability to support a large population of users in DVEs. However, servers can still get saturated with network bandwidth due to several reasons. In this case, state updates cannot be disseminated timely and the consistency of the virtual world cannot be guaranteed. This paper investigates update schedules to prioritize state updates from a global perspective for improving consistency in multi-server DVEs where each server has a limit on upload bandwidth. Using time–space inconsistency (TSI) metric, we analyze the problem using Lagrange Multipliers for a DVE in an ideal situation with unchanged configurations and the results are then used to develop an updating algorithm in practical systems. Experimental results show that the proposed algorithm significantly improves the consistency compared to other existing update algorithms.

1. Introduction

Distributed Virtual Environment (DVE) allows users from different places to communicate and interact with each other through a shared virtual world. Each user is represented in the shared virtual environment by an entity called avatar, whose state is controlled by the user through a client computer. Nowadays, DVE applications have been widely used in many areas such as military training, e-learning and online games (Bouras et al., 2002; Calvin et al., 1993; Chim et al., 2003; Li et al., 2003; SecondLife, WoW).

Multi-server architecture offers a cost-effective and scalable solution to the increasing resource demands (Everquest; Ng et al., 2005, 2002). In multi-server DVEs, multiple servers are geographically distributed and communicate with each other over the internet. The entire virtual world is assumed to be partitioned into regions and each region is maintained by a server. In order to provide a consistent view of the virtual world among all users, servers should disseminate state changes to the interested clients in a timely manner. Adding entities to the virtual world will no doubt increase the resource demand of servers. As the number of entities increases, the total resource demand may become huge. At present, a large scale DVE like Massively Multiplayer Online Games (MMOG) can include millions of concurrent users spread across the world, which demands a huge amount of computing and bandwidth resources on various servers. Moreover, workload like network bandwidth requirement in DVEs is also highly dependent on avatars' interactions (Svoboda et al., 2007). When avatars are crowed in the virtual world, more bandwidth for each client is required. For instance, the peak value of the bandwidth requirement from a server to a player in "World of Warcraft" (WoW) can exceed 64 kbps which is much larger than the median value 6.9 kbps (Svoboda et al., 2007).

With huge resource demand and large resource demand variability, if the resource renting or capacity planning polices are improper, server's resource, such as CPU, memory, disk and network bandwidth, may get saturated. In the presence of resource limitations, some entities' state updates may not be disseminated timely and thus inconsistency may occur between the primary copies of entities at the server side and their replicas at the client side. This can have a great influence on the experiences of users and the problem will become even more serious as the scale of the DVE grows. Therefore, an important issue in DVEs is how the limited resources can be used effectively and efficiently to reduce inconsistency.

Among all the resources in DVEs, the upload bandwidth is worth of paying special attention. The upload bandwidth used at servers generally grows in order of $O(n^2)$, which may take a large part of budget (Svoboda et al., 2007). Like the most popular online game WoW, the peak number of simultaneous active users has exceeded 5 million (WoWCencus). In total, 34.5 Gbps upload bandwidth from WoW data centers is demanded for gameplay. Therefore, this paper is primarily concerned with the upload

E-mail addresses: s080007@e.ntu.edu.sg (Y. Li), ASWTCAI@ntu.edu.sg (W. Cai).

1084-8045/$ - see front matter © 2014 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.jnca.2014.01.007

bandwidth resource. However, the proposed techniques can be applied to manage other type of resources as well. In multi-server DVEs, upload bandwidth consumption on state update is usually divided into two categories: the traffic from servers to clients and communication between servers. It is important to point out that, in order to keep the view consistent among the participating clients, some form of server to server communication is necessary.

Assuming the upload bandwidth is constrained at servers in a multi-server DVE, the aim of this paper is to investigate state update schedules to minimize total inconsistency in the system. We take time–space inconsistency (TSI) as the metric to evaluate the total inconsistency of a DVE. It has been shown TSI can effectively reflect the impact of inconsistency between entities' primary copies and replicas on the performance of DVEs (Zhou et al., 2004a). The contributions of this paper are as follows: firstly, we analyze a DVE under an ideal situation that the configurations of the DVE are assumed to keep unchanged. We formulate the problem of finding optimal update schedule to minimize total TSI as an inequality constrained problem (ICP). Secondly, Lagrange Multipliers are used to help derive a heuristic update algorithm that can be used in practical systems. The development of the algorithm consists of three steps:

1. For the ideal situation, Lagrange Multipliers are used to derive the conditions when the ICP is minimized. Based on the analytical results, for practical systems, a global updating priority is defined for each replica according to long-term benefit in terms of inconsistency reduction.
2. Since the update priority measured in a centralized manner may incur too much computational and communication overhead, a local regulating algorithm is proposed to estimate the update priority for each replica in a distributed manner.
3. Based on the update priority of each replica, a distributed update algorithm is developed for minimizing total TSI.

The rest of this paper is structured as follows. The related work is summarized in Section 2. In Section 3, the system model is introduced and the problem is defined in Section 4. In Section 5, the derivation of the heuristic update algorithm by using Lagrange Multipliers is introduced. Experimental results are presented in Section 6. In the last section, conclusions are made and future work is discussed.

2. Related work

A great deal of work on task scheduling issues has been studied in distributed high performance computing environments such as clusters and Grid (e.g., Aida and Casanova, 2008; Gupta et al., 2010). Most of the work focuses on allocating efficiently the workload among the processors to minimize the overall computing time. In some real-time systems, the work focuses on scheduling events with strict completion deadlines and the goal is to minimize the fraction of events that miss their deadlines (e.g., Ramamritham and Stankovic, 2006; Zhao et al., 2006). However, in our problem, we consider the scheduling issue at the application level for minimizing a continuous inconsistency metric in a specific application domain, i.e., DVE. Moreover, some techniques have been proposed to address the problem of minimizing bandwidth utilization with constraints on the accuracy of replicas (e.g., Olston and Widom, 2000; Sharaf et al., 2004). On the contrary, we consider the dual problem which is to minimize inconsistency with network bandwidth constraints.

Many research activities have been carried out for saving network bandwidth in DVEs. For example, dead reckoning (Brun et al., 2006; Pantel and Wolf, 2002; Zhang et al., 2006) and relevance filtering (Pan et al., 2010; Van Hook et al., 1994) are widely used in DVEs to compensate the network latency and reduce the network traffic. In dead reckoning, a DR model is used to predict an entity's state between updates. This helps to reduce the number of state updates needed. Relevance filtering can eliminate the irrelevant information by using the concept of Area of Interest (AOI). For an avatar, if some entities are not in its AOI (that is, the avatar is not interested in these entities), state updates of these entities can be saved. Although the total network traffic can be greatly reduced by using these techniques, the total requirement of bandwidth may still be very high as the scale of the DVE grows. If servers are not able to disseminate all the required state updates due to the constraints of bandwidth, an update scheduling algorithm needs to be used to determine the priority for each state update according to their potential impact on inconsistency.

Research has also been carried out on investigating update schedules in DVEs for improving consistency. In Faisstnauer et al. (2002), the authors proposed a priority round-robin algorithm to reduce the expected error in an entity state. The error-metric (such as visual error) is defined by user and the priorities are set with the purpose of minimizing the expected error. However, the network latency is not considered in this algorithm. The work presented by Yu et al. (2007) considered state update optimization issues in the client-server MMOG under mobile network environment. The inconsistency is defined as the distortion between avatar's actual location at the server side and the location interpreted by players. Suppose the bandwidth which is provided by the underlying wireless access points is constrained. In order to minimize the inconsistency between the servers and the clients, they proposed a network aware bandwidth allocation algorithm. However, only is the spatial inconsistency considered in their work while the temporal impact on the inconsistency is not considered. In Zhou et al. (2004b), a utility model was proposed to evaluate the relative importance of a simulation entity. The utility of an entity is determined by the number of entities on which this entity may have influence and the distance between this entity and the entities within its area of influence. Based on the utility model, some flexible updating mechanisms were devised for utilizing the bandwidth more efficiently. However, the calculation of utility is rather complicated and may incur considerable overhead.

A state update scheduling algorithm to minimize total TSI in single server DVEs with constraint of network bandwidth was proposed in Tang and Zhou (2010). The proposed algorithm can be easily extended for multi-server DVEs where a client is allowed to directly connect to multiple servers. However, the situation gets much more complicated if a client is only allowed to connect to a single server. In this case, inter-server communication to forward state updates is unavoidable. Some preliminary works on the update schedules to improve inconsistency in multi-server DVEs have also been carried out by Li and Cai (2010) and Li and Cai (2011). In Li and Cai (2010), Quality of Service (QoS) of a replica was defined using TSI metric and a simple approximation algorithm was proposed to minimize the total number of replicas without QoS. The problem addressed in this paper was initially formulated in Li and Cai (2011) where an interior point method was adopted to obtain the optimal update period for each replica in a centralized manner. However, the solution assumes that DVE configurations (that is, DVE partitioning, avatar-partition association, partition-server mapping, and client-server assignment) are static. In this paper, we further refine the problem formulation. Instead of solving the problem and obtaining the update periods directly, a scheduling algorithm based on update priority of each replica is proposed. The algorithm is extended in a distributed manner and the update priorities are also regulated with the gradual change of DVE configurations.