Advances in Engineering Research (AER), volume 82
2016 International Conference on Engineering and Advanced Technology (ICEAT-16)

An Interval Consistency Control Method to Meet Responsiveness Requirement in DVE Systems

Wei Zhang[1, a], Hangjun Zhou[2, b]

[1]College of Computer, National University of Defense Technology, Changsha, China;

[2]Department of Information Management, Hunan College of Finance and Economics, Changsha, China.

[a]zw_nudt@163.com, [b]zhjnudt@gmail.com

Keywords: DVE systems, responsiveness, consistency control method.

Abstract. In a distributed virtual environment (DVE) system, maintaining the consistency of each node is the key factor to preserve system usability. In addition, maintaining the responsiveness of users in the system is also an important factor affecting the users' interactive experience. Due to the trade-off between consistency and responsiveness, existing consistency control methods often ignore the responsiveness requirement, which makes them difficult to satisfy user's real needs. In this paper, we propose a new interval consistency control method that can effectively meet the requirements of system responsiveness. This method takes into account system consistency and responsiveness satisfaction degree simultaneously, and achieves an optimized balance of performance objectives of DVE system, thus effectively improving system usability. The experimental results proved the effectiveness of our method.

1. Introduction

Distributed virtual environment [1] is one of the most important application fields of distributed simulation technology [2,3]. It consists of lots of subsystems on different nodes and constructs an interactive virtual world for geographical distributed users to support them carry out some activities such as military training, online games, and distance education in the virtual world. Consistency problem is one of the most important issues in distributed virtual environment systems. To ensure that all users can see a consistent, unified view of the virtual world, it is necessary to keep the various entities in the virtual world in the same state on different nodes. Existing consistency control methods mainly consist of order consistency control method and interval consistency control method, which are respectively oriented to discrete models and continuous models, and jointly maintain the consistent state of each node in the system.

In addition, responsiveness is also an important factor to affect system usability. To ensure that in the virtual world users can have a good interactive experience that is similar to the one in the real world, we need to provide feedback in an acceptable time for users after they issue some actions. If the interval is too long, it will seriously affect the user's interactive experience in the virtual world. However, due to the trade-off between consistency and responsiveness, it is hard to optimize the two elements simultaneously, so the existing consistency control methods usually ignore the requirement of responsiveness, which makes them difficult to meet the real needs of users especially when the network delay is large.

Therefore, in this paper we propose a new interval consistency control method which can meet the demand of system responsiveness for the continuous model in virtual environment system. This method takes into account both system consistency and responsiveness satisfaction degree, and achieves an optimized balance of performance objectives of the system, thus effectively improving the usability of the system.

The rest of the paper is organized as follows: Section 2 reviews the existing methods of consistency control, section 3 gives the system model and the proposed consistency control method, section 4 shows and discusses the experimental results, finally this paper is summarized in section 5.

2. Related Work

Existing consistency control methods can be divided into two categories according to the optimization objectives. The first category is to keep the consistency of the message sequence as the goal, which is to ensure that all events can be executed in the same order in all nodes. In [4], the authors proposed a method to adjust the sequential consistency control policy in real time according to the application conditions, which can effectively realize the real-time switching of the control methods under the small switching overhead, and improve the practicability of the total order consistency method. In [5], the authors proposed a sequential consistency control method based on distributed queues, which can effectively achieve the scheduling of concurrent events on different nodes, and ensure the order consistency in DVE system.

In the order consistency control method, there is a kind of control method which aims at maintaining causal order consistency among the events. They only maintain the key cause and effect relation in the message sequence and improve the efficiency of event handling. In [6], the authors proposed a two-dimensional vector clock based causal order consistency control method, it can be used in the system which supports the unicast or multicast protocols. In [7], Zhou proposed a causal consistency control method based on key causal pairs, which can achieve a good consistency control effect when the system network provides FIFO service support. The above methods can only maintain the consistency of the discrete model in the DVE system, but cannot guarantee the correctness of the continuous model.

The second type of consistency control method is to maintain the interval consistency between events, which can guarantee the consistency of continuous models. In [8], the authors proposed a delay-based consistency control method which distinguishes local events and remote events in the process of event handling, and provides a good responsiveness for local events, but the consistency of the system is difficult to be guaranteed in this method. In [9], the authors proposed an interval consistency control method based on delay estimation, which can ensure the consistency of message execution interval among all nodes in the system. However, this method does not consider the responsiveness factor.

In addition, in this category some methods can guarantee the absolute time consistency of event execution. As in [10], the authors proposed a lock based consistency control technology, it forces all nodes to achieve consistent state before entering the next cycle, so as to maintain the absolute consistency of the system nodes. In [11], the authors proposed a local delay based consistency control technique, which guarantees the consistency of the system by delaying all the events for a period of time. In [12], the authors proposed an asynchronous clock consistency control method based on callback clock. By setting the deviation time of each node, a better balance between system responsiveness and absolute consistency is achieved.

3. Problem Statement and Algorithm Description

Problem Statement. In this paper, we use $V = \{v_1, v_2 ... v_n\}$ to denote the set of all nodes in DVE system, and $O = \{o_1, o_2 ... o_m\}$ to denote the set of all events. Given an $v_i \in V$, G_i denotes the set of all events generated by node v_i, R_i denotes the set of all events that node v_i can receive. Given an $o_m \in O$, if $o_m \in G_i$, $TG_i(o_m)$ indicates the generate time of event o_m at node v_i, $TS_i(o_m)$ denotes the excepted execution time of event o_m at node v_i. If $o_m \in R_i$, $TR_i(o_m)$ indicates the receive time of event o_m at node v_i, $TE_i(o_m)$ denotes the actual execution time of event o_m at node v_i.

In order to ensure the consistency of each node, it is necessary to maintain the consistency of message processing. We divide the time stamp consistency into order consistency and interval consistency.

Definition 1 (Order Consistency)

$$\forall v_i, v_i \in V; o_m, o_n \in R_i \cap R_j, TE_i(o_m) \le TE_i(o_n) \to TE_j(o_m) \le TE_j(o_n) \quad (1)$$

Order consistency can ensure that all events can be executed in the same order at each node. To maintain the consistency of the continuous model, it is also necessary to ensure that the events are executed at same intervals.

Definition 2 (Interval Consitency)

$$\forall v_i, v_i \in V; o_m, o_n \in R_i \cap R_j, TE_i(o_m) - TE_i(o_n) = TE_j(o_m) - TE_j(o_n) \quad (2)$$

To keep the interval consistency, we need to make sure that all events must be executed in accordance with the expected execution time. For the local events generated by the node, they can be executed at the expected time. However, for remote events, due to the existence of network transmission delay, there may be a receiving node v_j, which received the event at a time later than the expected execution time, that is $TR_j(o_m) > TS_j(o_m)$. In this case the event cannot be executed as scheduled. There has been inconsistent phenomenon. Here we define the event inconsistency as the difference between the actual execution time and the expected execution time.

Definition 3 (Event Inconsistency)

$$IC_j(O_m) = \begin{cases} 0 & , \; TR_j(O_m) \le TS_j(O_m) \\ TE_j(O_m) - TS_j(O_m), & TR_j(O_m) > TS_j(O_m) \end{cases} \quad (3)$$

For each node in the system, the inconsistency of a node is defined as the average inconsistency of all events it received.

Definition 4 (Node Inconsistency)

$$IC_i = \sum_{\forall o_m \in R_i} IC_i(o_m) / |o_m| \quad (4)$$

In addition to the functional correctness requirements of the system, the responsiveness requirements should also be considered because it represents the system ability to respond to the interaction. In [10], we proved to maintain the interval consistency in each node, the following condition should be satisfied.

Condition 1 (Interval Consistency Constraint)

$$\forall o_m, o_n \in G_i, TS_i(O_m) - TG_i(O_m) = TS_i(O_n) - TG_i(O_n) = DT_i = RT_i \quad (5)$$

In the above condition, DT_i represents the difference between the expected execution time and the generation time of the event, we call it the node delay time. RT_i represents the responsiveness of a node, whose value is equal to the node delay time. We use rr_i to denote the responsiveness requirement of the node, then the node responsiveness dissatisfaction degree RTU_i is defined as.

Definition 5 (Node Responsiveness Dissatisfaction Degree)

$$RTU_i = \begin{cases} 0 & , \; RT_i \le rr_i \\ RT_i - rr_i, & RT_i > rr_i \end{cases} \quad (6)$$

It can be seen from the above description that the goal of the system is to reduce the inconsistency and responsiveness dissatisfaction of each node as much as possible. However, due to the trade-off between these two factors, it is difficult to optimize them at the same time. In the next section, we will introduce the interval consistency control method proposed in this paper.

Interval Consistency Control Method. In our previous work [9], we proposed a consistency control method based on delay estimation for the consistency requirements of continuous models. Although this method ensures the consistency of events, it does not have a quantization calculation method for inconsistency. On the other hand, it does not take into account user's responsiveness requirements. So its practical application range is limited. In fact, from the definition of node inconsistency and node responsiveness dissatisfaction degree, we can see the node delay time is a very important factor to determine system performance. If the delay time is large, the node inconsistency will be greatly reduced, but the node responsiveness dissatisfaction degree will be increased. Conversely, if the execution time is set to be closer to the event generation time, the responsiveness will be guaranteed, but the inconsistency of the remote events will be increased. Therefore, to achieve a better balance, the consistency needs in different applications should be consider specifically. In this paper, we introduce the equilibrium condition.

Condition 2 (Equilibrium Constraint)

$$\forall v_i \in V, W_c \times IC_i = W_r \times RTU_i \tag{7}$$

In the condition, W_r and W_c represent the responsiveness weight and consistency weight respectively, their values depend on the type of application and DVE system optimization goals. The greater the weight is, the more important that kind of requirement is. For example, if the system has a high demand of responsiveness requirements, a higher value of W_r and a lower value of W_c should be set so that the value of the responsiveness dissatisfaction degree could be low. To meet the above conditions, we designed the following distributed algorithm to adjust the node delay time periodically which is shown in Figure 1.

Algorithm 1.

```
set DT_i=rr_i at initial stage
if TimerCounter > Counter_TH then
    get the first event o_1 from IntO
    if DT(o_1)<rr_i then
        DT_i=rr_i
    else
        IC_i=0
        RTU_i= DT(o_1)-rr_i
        n=2
        while RTU_i * W_r>IC_i * W_c do
            get event o_n from IntO
            IC_i=((n-2)*IC_i+ (DT(o_n-1)-DT(o_n)))/(n-1)
            RTU_i= DT(o_n)-rri
            n=n+1
        end while
        DT_i= RTU_i
    end if
else
    insert new o_m ∈R_i to IntO in descending order of delay time
end if
```

Fig. 1 Distributed interval consistency control method.

In the initial stage, the delay time of each node is set to the value of responsiveness requirement of the node. Then, we divide the execution time into many periods in the running stage. In each period, every node collects information such as delay time and inconsistency of remote events, and sort the remote events according to the delay time. When the period is end, an algorithm is run to adjust the node delay time according to the information collected in this period, and then we enter the next period.

In the adjustment process, if the delay time of all events are less than the responsiveness requirement of the node, the delay time of the node is set to the responsiveness requirement of the node. Otherwise, the delay time of the node is set to the delay time of the event with the largest value, and then the delay time of the node is gradually reduced according to the delay time of the events in the queue. In this process, the responsiveness dissatisfaction degree of the node is gradually decreased, and node inconsistency will increase until the equilibrium constraint defined in condition 2 is satisfied. The node delay time obtained in this case will be the delay time of the node in the subsequent period. This process runs periodically during the run time, so that the overall consistency and responsiveness of the system can be adjusted and controlled so as to achieve the balance between responsiveness satisfaction and interval consistency. In the next section, we will show the experimental results of the method.

4. Experimental Results

We constructed a simulated DVE system with 1,000 nodes to verify the effectiveness of the proposed method. To simulate the communication characteristics of the network, we generate the inter-node delay by the DS^2 [14]. DS^2 is a delay synthesis tool based on the measurement of real Internet communication. It can effectively simulate the delay patterns of Internet. In the experiment, we set the parameters of the equilibrium constraint   to 0.8, and   to 0.2, so the responsiveness requirements is more important in our optimization goal. We run the virtual environment for 10,000 cycles, in each cycle every node has a certain probability to send events to its neighbors and run the adjustment algorithm every 100 cycles. The node's average inconsistency and responsiveness dissatisfaction degree are recorded. The statistics results are shown in Figure 2.

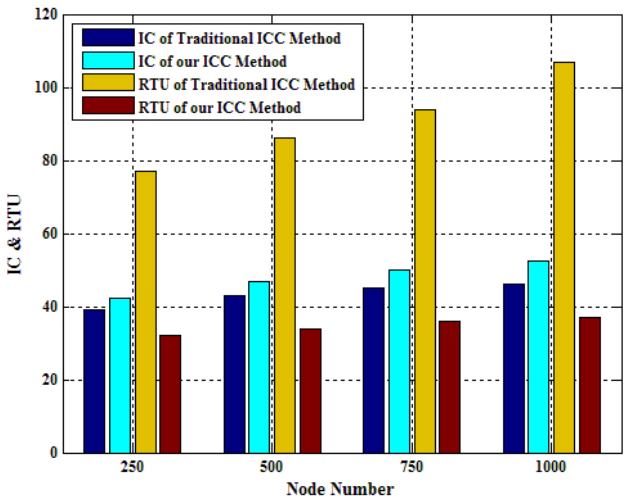

Fig. 2 Comparison of experimental results.

In Figure 2, the x-axis represents the number of nodes in the virtual environment, varying from 250 to 1000, and the y-axis represents the average inconsistency and the average responsiveness dissatisfaction of the nodes. The dark blue bars represent the inconsistency achieved by the interval consistency control method we proposed in [10]. The light blue bars represent the inconsistency of the consistency control method proposed in this paper. The yellow bars represent the responsiveness dissatisfaction of the method in [10], and the red bars represents the responsiveness dissatisfaction of the method proposed in this paper. It can be seen from the results that the average inconsistency of our method is slightly higher than that of the method of proposed in [10]. This is because the main optimization goal of the original method is to keep consistency, and it sets a fixed proportion as the time-out event, so the system inconsistency is relatively fixed. In this paper, the responsiveness and consistency are considered at the same time, and the inconsistency of the system is dynamically adjusted at run time to provide a balanced state. In terms of responsiveness, our method can significantly reduce the responsiveness dissatisfaction degree compared with the traditional method, the ratio is more than 60%.

The above experiments clearly show that the proposed method can achieve an optimized balance of the system performance objectives, thus improving the usability of the system.

5. Summary

Consistency and responsiveness are two core elements that ensure DVE system usability. In order to keep the state consistency of continuous model in each node, it is necessary to maintain the interval consistency of events. Existing interval consistency control methods mainly aim at the consistency of each node, ignoring the influence of the responsiveness satisfaction on the system interactive efficiency. In this paper, we propose a new consistency control method for the equilibrium state of all factors. We can achieve an optimized balance of the performance objectives of the system on the basis of satisfying the responsiveness requirements of the nodes. The effectiveness of our method is verified by simulation experiments.

6. Acknowledgement

The work described in this paper was supported by the National Natural Science Foundation of China (Grant No. 61303187).

References

[1] Valadares A, Gabrielova E, Lopes C V. On designing and testing distributed virtual environments[J]. Concurrency and Computation: Practice and Experience, 2016.

[2] Tang Y H, Zhang B D, Wu J J, et al. Parallel architecture and optimization for discrete-event simulation of spike neural networks[J]. Science China-Technological Sciences, 2013, 56(2), pp. 509-517.

[3] Hou B, Yao Y, Wang B, et al. Modeling and simulation of large-scale social networks using parallel discrete event simulation[J]. Simulation-Transactions of The Society for Modeling and Simulation International, 2013, 89(10) , pp. 1173-1183.

[4] Mocito J, Rodrigues L. Run-time switching between total order algorithms[C]. European Conference on Parallel Processing. Springer Berlin Heidelberg, 2006: 582-591.

[5] Zhou H, Zhang W, Peng Y, et al. A distributed approach to consistent order delivery of concurrent events in asynchronous DVEs[C]. 2010 2nd International Conference on Education Technology and Computer. IEEE, 2010, 4: V4-95-V4-98.

[6] Raynal M, Schiper A, Toueg S. The causal ordering abstraction and a simple way to implement it[J]. Information processing letters, 1991, 39(6): 343-350.

[7] Zhou S, Cai W, Turner S J, et al. Critical causal order of events in distributed virtual environments[J]. ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM), 2007, 3(3): 15.

[8] Qin X. Delayed consistency model for distributed interactive systems with real-time continuous media[J]. Journal of Software, 2002, 13(6): 1029-1039.

[9] Zhang W, Zhou H, Peng Y, et al. An Interval Consistency Control Method Based on Estimating Network Delay in DVE Systems[J]. Computer Engineering & Science, 2012, 34(3): 55-61.

[10] Funkhouser T A. RING: a client-server system for multi-user virtual environments[C]. Proceedings of the 1995 symposium on Interactive 3D graphics. ACM, 1995: 85-ff.

[11] Mauve M, Vogel J, Hilt V, et al. Local-lag and timewarp: providing consistency for replicated continuous applications[J]. IEEE transactions on Multimedia, 2004, 6(1): 47-57.

[12] Zhang W, Zhou H J, Peng Y X, et al. Asynchronous time consistency control methods in distributed interactive simulation[J]. Journal of Software, 2010, 21(6): 1208-1219.

[13] Zhang B, Ng T S, Nandi A, et al. Measurement based analysis, modeling, and synthesis of the internet delay space[C]. Proceedings of the 6th ACM SIGCOMM conference on Internet measurement. ACM, 2006: 85-98.