Simulation Modelling Practice and Theory 68 (2016) 54–79

Contents lists available at ScienceDirect

Simulation Modelling Practice and Theory

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

An accurate interest matching algorithm based on prediction of the space–time intersection of regions for the distributed virtual environment

Yong Peng, Mei Yang, Quanjun Yin*, Yabing Zha

College of Information Systems and Management, National University of Defense Technology, Changsha, Hunan 410073, PR China

ARTICLE INFO

Article history:
Received 3 August 2015
Revised 21 July 2016
Accepted 5 August 2016

Keywords:
Interest management
Interest matching
Distributed simulation
DVE
HLA

ABSTRACT

Interest matching is an important data-filtering mechanism for a large-scale distributed virtual environment. Many of the existing algorithms perform interest matching at discrete timesteps. Thus, they may suffer the missing-event problem: failing to report the events between two consecutive timesteps. Some algorithms solve this problem, by setting short timesteps, but they have a low computing efficiency. Additionally, these algorithms cannot capture *all* events, and some *spurious events* may also be reported. In this paper, we present an accurate interest matching algorithm called the predictive interest matching algorithm, which is able to capture the missing events between discrete timesteps. The PIM algorithm exploits the polynomial functions to model the movements of virtual entities, and predict the time intervals of region overlaps associated with the entities accurately. Based on the prediction of the space–time intersection of regions, our algorithm can capture *all* missing events and does not report the *spurious events* at the same time. To improve the runtime performance, a technique called region pruning is proposed and used in our algorithm. In experiments, we compare the new algorithm with the frequent interest matching algorithm and the space–time interest matching algorithm on the HLA/RTI distributed infrastructure. The results prove that although an additional matching effort is required in the new algorithm, it outperforms the baselines in terms of event-capturing ability, redundant matching avoidance, runtime efficiency and scalability.

1. Introduction

A distributed virtual environment (DVE) provides a virtual world where the entities can interact with each other. With the growing scale and complexity of the simulation system, the number of virtual entities in the virtual world may be extremely large. And all-to-all data exchange between computational nodes, on which virtual entities are hosted, will consume a large amount of network bandwidth and decrease the runtime performance of the system significantly. Therefore, efficient data distribution becomes crucial to reduce the exchange of irrelevant data between nodes and to meet performance and usability requirements [1,2].

In past decades, a number of techniques are proposed with the idea of interest management: they ensure the entities receive only interested data and filter the uninterested data. In the interest management mechanism, two regions in the virtual space are usually defined for each entity: the *update region* and the *subscription region* [3–5]. The update region is

* Corresponding author.
E-mail address: yin_quanjun@163.com (Q. Yin).

http://dx.doi.org/10.1016/j.simpat.2016.08.001

the area corresponds to the location of an entity and in which the entity is expected to be moving. The subscription region, which is also called area-of-interest (AOI), is the region in which the entity is interested. A subscription entity only needs to receive attribute updates from the other entities whose update regions overlap with subscription regions. The process of checking the overlap state of update-subscription region pairs is referred to Interest Matching (IM) [2].

Most of distributed simulations have the concept of logical time (simulation time). In real-time simulation, logical time is equal to physical time. In HLA, the logical time is represented by a variable. Time step in this paper has different meanings under different contexts. For advance of simulation time, a time step is one time of logical time advancement, which is not necessary in uniform step. For region update, a time step is an update of region. For IM algorithm, a time step is an execution of IM algorithm, which is a variable and is denoted as the beginning logical time of an execution. Generally, each time of IM algorithm execution is driven by the region update or by the advance of the simulation time.

Because entities update their update/subscription regions at discrete timesteps, most existing IM algorithms also perform matching processes at discrete timesteps. If those regions had been moving continuously from their locations at one time step to their locations at the next timestep, the discrete IM algorithms that cannot detect this overlap state will cause the missing-event happening. This may lead to the missing-event problem [6]: the IM algorithms cannot capture all the actual events. Moreover, the separation of two overlapped regions is important but is ignored by many existing IM algorithms. For instance, in a military simulation, when an aircraft is flighting out of the range of a radar, a separating event occurs. If IM algorithms can report the accurate time of the separating event, the radar can make better decisions in the next step, and the aircraft can immediately stop updating attributes of the radar.

To filter the irrelevant messages accurately, some of the existing IM algorithms define multidimensional and complex update or subscription regions for virtual entities. However, matching these kinds of regions is always time consuming. Furthermore, because the update region and subscription region are changed frequently, many of the existing IM algorithms have to perform the region matching operations at every timestep [7–12], which produces many unnecessary overlap tests. Both of the two reasons make these IM algorithms may suffer heavy computational workload, which is unacceptable in some applications. Therefore, people always have to make a trade-off between filtering precision and runtime efficiency in the interest matching processes.

To improve the filter precision and the runtime efficiency in the IM processes simultaneously, we propose an accurate IM algorithm called the Predictive Interest Matching algorithm (PIM), which is based on the prediction of the space–time intersection of regions. The PIM algorithm uses multidimensional rectangles to represent the update and subscription regions of virtual entities, which is similar to the definition of the extent of a region in the routing space of HLA [13–14]. The space–time intersection of regions in our algorithm is predicted by solving the movement inequations of regions. By accurately predicting time intervals at which the update regions and the subscription regions overlap with each other, our algorithm can catch the missing events and distinguish the overlapping events of regions and the separating events of regions. Furthermore, by predicting the space–time intersection of regions, the PIM can avoid performing redundant interest matching operations.

To further improve the runtime efficiency of PIM, we introduce a region pruning method in the PIM, which is based on the maximum area of potential overlap of a region. The region pruning method is used for culling out the region pairs that are unlikely to overlap with each other. Our algorithm uses the region pruning method to test the potential overlap state of region pairs before predicting space–time intersections of the region pairs. In this way, many unnecessary overlap tests of regions can be avoided and the computational workload of the PIM algorithm is reduced.

In experiments, we test our algorithm in a World War II dogfight scenario, which is run on the HLA\RTI distributed simulation infrastructure. Four sets of experiments are conducted to evaluate event-capturing ability, redundant matching avoidance, runtime efficiency and scalability of our algorithm. Three IM algorithms are chosen as baselines: the Frequent Interest Matching algorithm (FIM) [15], the Space–Time Interest Matching algorithm (STIM) [2] and the Predictive Interest Matching algorithm using Brute Force approach (PIM-BF).

The rest of this paper is organized as follows. Section 2 briefly describes the existing IM algorithms and discusses their advantages and disadvantages. In Section 3, the prediction of the space–time intersection of regions in our algorithm is presented. Section 4 describes the region pruning process of our algorithm in detail. In Section 5, the implementation of the PIM algorithm is discussed. Section 6 presents the experimental results of the proposed algorithm and a comparison analysis with the existing matching algorithms. In Section 7, issues for discussion are presented. Finally, Section 8 concludes the paper.

2. Related work

In this section, we briefly describe the existing IM algorithms. Their advantages and disadvantages are also discussed briefly.

2.1. Interest matching algorithms

In recent years, a number of IM algorithms have been proposed for distributed simulation or DVE. These algorithms can be generally classified into four categories: the brute force algorithm, the grid-based algorithm, the hybrid-based algorithm and the sort-based algorithm.

The brute force algorithm checks all update-subscription pairs of regions to find the overlapping region pairs [11]. In an early version of this algorithm, the region is defined as the multidimensional rectangle [16–18]. To improve the filtering