Simulation Modelling Practice and Theory 48 (2014) 112–128

Contents lists available at ScienceDirect

Simulation Modelling Practice and Theory

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

Un-identical federate replication structure for improving performance of HLA-based simulations

Zengxiang Li [a,*], Wentong Cai [b], Stephen John Turner [b]

[a] *Institute of High Performance Computing, 1 Fusionopolis Way, Singapore 138632, Singapore*
[b] *Nanyang Technological University, 50 Nanyang Avenue, Singapore 639798, Singapore*

ARTICLE INFO

Article history:
Received 22 October 2013
Received in revised form 19 June 2014
Accepted 24 June 2014
Available online 25 August 2014

Keywords:
Parallel and distributed simulation
High Level Architecture
Time synchronization
Performance
Replication
Software diversity

ABSTRACT

The execution of an HLA-based simulation (federation) is usually time consuming, as it usually involves a number of compute-intensive simulation components (federates). To improve simulation performance, an un-identical federate replication structure is proposed in this article. For the same federate, multiple replicas are developed in software diversity manner by employing different synchronization approaches. The simulation performance is improved by always choosing the fastest replica to represent the federate in the federation. The replication structure is implemented in a transparent manner without increasing federation scale. Message exchange and time management mechanisms are developed to handle the different behaviors of those un-identical replicas. Correctness of the replication structure is proved in theory and verified by experiments. The experimental results have also shown that the un-identical federate replication structure achieves significant performance enhancement with good scalability and marginal overhead.

1. Introduction

The *High Level Architecture* (HLA), IEEE standard [1] provides a framework to build a parallel and distributed simulation (federation) by re-using and inter-operating a group of simulation components (federates). HLA-based simulations, i.e., parallel and distributed simulations developed following HLA standard, are usually developed to study the problems of a complex system, e.g., supply chains, computer networks, and ecological systems. Their executions are usually time consuming, as they involve a number of compute-intensive federates which simulate corresponding subsystems in desired detail and fidelity. In addition, the simulation is usually executed in many times with various parameter inputs for the purpose of obtaining statistical simulation results. Moreover, real time responses is required in most symbiotic simulations [2] which are embedded in the real system. Therefore, accelerating simulation executions is very important to make simulation economical and attractive to industry and research communities.

In order to improve simulation performance, an un-identical federate replication structure is proposed in this article, inspired by the concepts of active replication and software diversity [3]. Using active replication, the computation is performed at each replica independently and concurrently. Using software diversity, also known as N-Version Programming [4], multiple functionally equivalent programs are developed independently following the same initial specifications. For HLA-based simulations, we propose to develop un-identical replicas for the same federate using either a conservative

---

* Corresponding author.
E-mail addresses: liz@ihpc.a-star.edu.sg (Z. Li), aswtcai@ntu.edu.sg (W. Cai), steve@pmail.ntu.edu.sg (S.J. Turner).

http://dx.doi.org/10.1016/j.simpat.2014.06.016
1569-190X/

(e.g., CMB protocol [5,6]) or an optimistic (e.g., Time Warp protocol [7]) synchronization approach. Therefore, the replicas are likely to exhibit different runtime performances during the simulation execution. Consequently, simulation performance can be improved by always choosing the fastest replica to represent the federate in the federation.

Using active replication, the execution cost (e.g., the number of computers) may increase dramatically. Fortunately, computing resources can be obtained more and more easily in modern computing environments, e.g., multi-core or many-core processes and on-demand virtual machine instances on Cloud. Hence, it is meaningful to improve simulation performance relying on active replication in spite of the increased resource consumption. Furthermore, active replication can also be used for fault-tolerance purpose [8].

There are two reasons to design federate replicas in software diversity manner by employing different synchronization approaches. Firstly, simulation developers usually hesitate to make the choice on synchronization approaches, as it is difficult to predict their performances for simulation executions with different parameters and inputs [9]. The un-identical federate replication structure relieves simulation developers from the burden by taking advantages of both conservative and optimistic synchronization approaches in the same simulation execution. Secondly, conservative and optimistic synchronization go towards two extremes: the former preserves causality constraint by blocking fast federates; while the latter allows federates to progress freely but may rollback execution when a causality error occurs. Hence, their performance difference is generally significant, and thus, the replication structure is expected to achieve significant performance improvement. Most real-world simulations employ conservative synchronization due to its simplicity. Additional costs are introduced to develop an OPT replica for the same federate. Fortunately, some approaches and frameworks have been proposed to relieve simulation developers from the burden of handling complex optimistic synchronization details. Wang et al. [10] have proposed a rollback controller using a middleware approach to handle the complex rollback procedure on behalf of the simulation model. Santoro and Quaglia [11] have implemented a MAgic State Manager (MASM) to handle state management issues for optimistic synchronization in a way completely transparent to the federate itself. They have further designed and implemented a Time Management Converter (TiMaC) to perform mapping of the conservative HLA synchronization interface onto the optimistic one [12]. Such a mapping allows transparent optimistic execution for simulation models originally designed using conservative synchronization. Besides state-based optimistic execution, μsik [13] and Rensselaers Optimistic Simulation System (ROSS) [14] support reverse computation-based optimistic execution. Furthermore, LaPre et al. [15] have proposed a tool capable of automatic emission reverse event handlers, with execution speed comparable to the hand-written code.

Traditionally, replicas of a federate are treated as individual federates. As a result, federation scale, communication traffic and time synchronization overhead increase accordingly. It is also difficult to keep consistent among replicas of the same federate. In contrast, the un-identical federate replication structure is designed in a transparent manner. Replicas of the same federates are connected to the federation through a middleware called replication manager, which masks the presence of multiple replicas of the federate without increasing federation scale. Replicas employing different synchronization approaches have different behaviors on message exchange and time advancement. Message exchange and time management mechanisms are developed to handle the different behaviors and keep replicas eventually consistent, i.e., processing the same events in TS order. Consequently, the simulation execution with our replication structure will produce the same results as normal simulation execution using either conservative or optimistic synchronization.

As a follow-up work of [16], this article (i) describes the principles of the un-identical federate replication structure using a middleware approach; (ii) revises the implementation of message exchange and time management mechanisms; (iii) proves that the same simulation results are produced with and without replication structure; and (iv) reports more experimental results regarding to performance improvement, scalability and overhead.

This article is structured as follow: Section 2 briefly introduces HLA-based simulations. Section 3 and 4 respectively provides the implementation details and correctness proof of the un-identical federate replication structure. Section 5 introduces the P-hold simulation model and reports the experiment results. The related work on improving simulation performance is discussed in Section 6. Section 7 concludes the article and outlines the future work.

2. HLA-based simulations

In an HLA-based simulation, federates participate in a federation execution through underlying Runtime Infrastructure (RTI), as shown in Fig. 1. Several groups of management services are defined in the HLA standard. They are usually

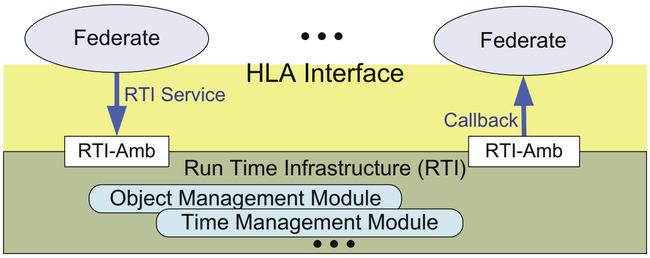

Fig. 1. Overview of an HLA-based simulation.