Hierarchical Federation Composition for Information Hiding in HLA-based Distributed Simulation

Jung Hyun Ahn, Moon Gi Seok, Chang Ho Sung, and Tag Gon Kim
Systems Modeling Simulation Laboratory
Korea Advanced Institute of Science and Technology (KAIST)
373-1 Kuseong-dong, Yuseong-gu, Daejoen, Korea
{jhahn, mgseok, chsung}@smslab.kaist.ac.kr, tkim@ee.kaist.ac.kr

***Abstract*—This paper presents a hierarchical federation system interconnected among federations and proposes a hierarchical federation composition algorithm that composes federations into different compositions of hierarchical federations and then efficiently assigns composite federations to a distributed environment while enabling information hiding. We implement the hierarchical federation system and evaluate it using real networks. The experimental results show that our composition algorithm performs better than a random composition algorithm in optimizing the composition of federations while achieving information hiding.**

***Keywords*-HLA, information security problem, hierarchical federation system, information hiding, hierarchical federation composition**

I. Introduction

As a standard architecture for interoperation between heterogeneous distributed simulations that are developed with different languages and platforms, HLA was approved as IEEE Standard 1516 in 2000 [1]. It provides a common framework within which simulation developers can structure and describe their simulations, and hence it enables cooperation to achieve overall objectives in an extensible and scalable manner.

In spite of HLA's support for simulation interoperation, it does not provide information hiding, i.e., a mechanism to avoid the sharing of security information (i.e., unclassified national security, proprietary interests, and top secret information), during the simulation. When several nations or competitive companies in different organizations participate for interoperating simulations, an information security problem may arise, particularly in military applications [2] and competitive distributed supply chains [3]. If improvement in information hiding, as a solution to this security problem, is achieved, then simulation participants can engage in HLA-based distributed simulations with reduced security clearance conflict.

Therefore, we present a hierarchical federation system wherein a tree-like structure is constructed and each federation shares information below it and acts as a federate to the federation above. In this paper, we propose a hierarchical federation composition algorithm in order to reduce the communication overhead and satisfy the information hiding requirements at the *same* time. This enables the compositions of federations that frequently interact with each other and deploys them into the same execution environment. In particular, the main concern is how to compose a hierarchical structure of federations with respect to a cost function based on the level of balance between the capacity of information hiding requirements and reduction of the communication overhead. Then, by using the federation composition result, we can deploy the hierarchical federation system into distributed execution environments. For an evaluation of our hierarchical federation composition algorithm, we implement a prototype of the hierarchical federation system to demonstrate the performance of a real network. The experimental results demonstrate the feasibility of the hierarchical federation composition.

II. System Model

Before we discuss system model, we consider federations in a hierarchical federation system manage a data model which is an external FOM (eFOM). It describes the simulation information used by a particular federation. Explicitly, the federation model has an eFOM with a format as described in Fig. 1; it has a list of external FOM Data Nodes (*eFDNs*), which an *eFDN* is a record of the type [object-name, object-handle, parent indicator and p/s (publication/subscription)]. Informally, the *eFDN* in the lower federation's eFOM is included in the upper federation's eFOM. However, an *eFDN'* is security information, and as such it is private information; hence it is not open to other federations.

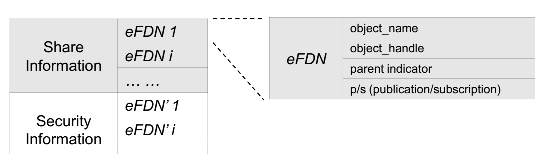

Figure 1. The format of external FOM in the document data file

Next, we consider a composite federation (CF) model as a system model. This model is a subset of the hierarchical

1550-6525/10 $26.00 © 2010 IEEE
DOI 10.1109/DS-RT.2010.35

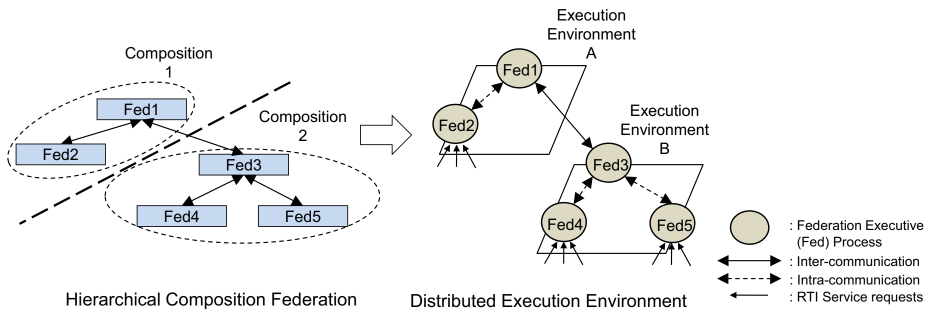

Figure 2. An illustration of hierarchical composition of federations and their distributed execution environment.

federation system, which interconnects federations into a hierarchical structure of federations. The CF model describes a combination of federations that are closely related and coupled to each other. Unlike a federation, this model has another data model, eFOM'. This model represents the share information that can be exposed outside of the CF model. Specifically, the CF model is also defined as follows.

Definition 1. *The composite federation model is $CF =< eFOM', F, C >$, where eFOM' is the shared eFDNs, which are the union of eFOMs at each federation in F except secured eFDN's (i.e., eFOM' is the same as eFOM's eFDNs of the highest federation). F is a set of federation models and C a set of connection edges. A connection $c(i, j) \in C$ is a link between two federations in CF; $i, j \in F$.*

A composition $\mathbb{C}$ for the CF models is represented by a *disjoint* composition. This allows us to reduce the communication overhead within an optimal location of nodes. Two federations, $\mathcal{F}_1$ and $\mathcal{F}_2$, belong to the same CF if affinity between them is high. Otherwise, the two are assigned to different CFs. Through the affinity, a hierarchical federation system is decomposed by several CF models, thus yielding different compositions. This matter is discussed in further detail in Section III-A.

As shown above, several federations' eFOMs in the CF model are physically located on the same node. By locating federation's eFOM document files in the same node, a federation does not share security information with federations of the other CF model. As a result, it is possible to keep eFOMs solely inside the CF model. Federations can protect their eFOMs from exposure to the other CF models. The composition enhances the CF model's *capability* of information hiding by avoiding sharing of security information in a particular federation of the hierarchical federations. This capability enables federations to select to share or protect their information from other CF models by using its eFOM'. Therefore, this capability ensures that the exchange of information between CFs is permitted only within an individual CF's eFOM'.

Definition 2. *Given n composite federations in a hierarchical federation system, $CF_i =< eFOM'_i, F_i, C_i >$, $i = 1, \ldots, n$, the information hiding is determined by $eFOM'_i$.*

III. Hierarchical Federation Composition Algorithm

A. Overview of Hierarchical Federation Composition

First, the composition in simulation-based distributed systems [4] has been considered in depth in efforts to cut down the communication overhead. Therefore, we apply this to the division of the hierarchical federation system into several combined federations, namely, *composite federations*. As described in Section II, these CFs cannot only selectively hide the security information, but also partially reduce the communication overhead that occurs between federations involved in the hierarchical federation system. These findings imply that there is a *trade-off* between the capability of information hiding and reduction of the inter-communication overhead. Therefore, the goal of the hierarchical federation composition is to search for balanced level that minimizes the amount of inter-communication while ensuring that information is secured in the distributed environment.

In Fig. 2, we illustrate a hierarchical federation composition and the distributed execution environment in the hierarchical federation system. If federations are mapped to corresponding federation execution processes, they are composed such that they can be deployed into the distributed execution environments with regard to reduction of the inter-communication overhead. The hierarchical federation composition enables partitioning of the initially created federations into several composite federations, such as *Composition 1* and *Composition 2*. Furthermore, from the perspective of information hiding, we can physically deploy and allocate each federation's eFOM information files (i.e., information document files) as well as federation execution processes into the distributed execution environments, such as *Execution Environment A* and *Execution Environment B*. Therefore, federations' security information in the eFOM information files is protected by *composition* in different execution environments.

As seen in the figure, intra-communication is performed through federation execution processes within the same execution environment. On the other hand, inter-communication is accomplished between federation execution processes over different execution environments. In addition, communication is carried out between federate to federation as well as between federation to federation in the distributed execution environments. The federate application joins a federation execution process and requests RTI services to its federation.

B. Detailed Refinement Composition Algorithm

We propose the following composition algorithm. The hierarchical federation composition algorithm enables the hierarchical federation system to find the optimal set of CFs, which provides a balanced level between the capability of information hiding and reduction of the inter-communication overhead. In order to support a large hierarchical federation system, an efficient composition algorithm should be considered. To this end, the Detailed Refinement Composition (DRC) algorithm is proposed. The basis of the refinement is a dynamic programming technique to find the optimal composition of the local information.

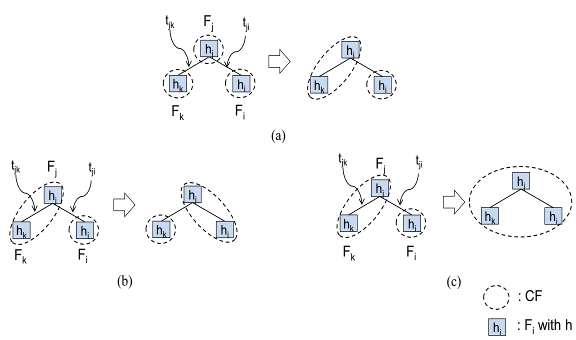

Figure 3. Example of mering refinement process for composition. (a) Merging Refinement. (b) Spilt-Merging Refinement. (c) Remerging Refinement.

Since a CF opens the security information to federations of the same node, we assume that there is a hiding cost, C_{hiding}. An individual federation $\mathcal{F} \in CF$ has an amount of information hiding $\boldsymbol{H}$ that reflects how the information hiding is achieved in proportion to the extent of security information. C_{hiding} is proportional to the sum of the amount of security information of each federation. The C_{hiding} of CF_k can be calculated as follows:

$$\Big(C_{hiding}\Big)_{CF_k} = \left(\sum_{\mathcal{F}_i \in CF_k} \boldsymbol{H}_i\right) - \boldsymbol{H}_0, \qquad (1)$$

where H_i is the amount of the information hiding of $\mathcal{F}_i$ in CF. H_0 is H of the highest federation in CF.

In addition, we assume there is a communication cost C_{com} if federations are located in different physical nodes. This depends on the amount of information in proportion to the extent of communication overhead when communication is achieved within the distributed environment. This edge has an amount of communication $\boldsymbol{T}_{ij}$, and thus we define x_{ij} to denote whether *c(i, j)* is for inter-communication between federations or not.

$$x_{ij} = \begin{cases} 1, & \text{if } c(i,j) \text{ is inter-communication} \\ & \text{between } \mathcal{F}_i \text{ and } \mathcal{F}_j \\ 0, & \text{otherwise.} \end{cases} \qquad (2)$$

Therefore, the cost of communication which represents the bottleneck of the performance is significantly important, and the C_{com} of CF_k can be expressed

$$\Big(C_{com}\Big)_{CF_k} = \sum_{\mathcal{F}_i \in CF_k, \mathcal{F}_j \notin CF_k} \Big(x_{ij}\boldsymbol{T}_{ij}\Big). \qquad (3)$$

Next, we formulate the hierarchical federation composition by finding the optimal set of CFs such that the following cost function is minimized. From the specified composition $\mathbb{C}$, the cost function can be expressed as:

$$Cost_{\mathbb{C}} = \alpha\left(\sum \Big(C_{hiding}\Big)_{CF_i}\right) + (1-\alpha)\left(\sum \Big(C_{com}\Big)_{CF_i}\right)$$
$$\text{with } 0 \leqslant \alpha \leqslant 1, \qquad (4)$$

where the composition coefficient α is a tunable parameter that *balances* the relative importance between the information hiding cost and the communication cost. This coefficient depends on the purpose and usage of the simulation applications.

Therefore, the DRC algorithm refines all federations by finding the optimal composition using the objective function of the hierarchical federation composition which is to minimize the cost in (4). In a given hierarchical federation system, the DRC algorithm initially assigns each federation to one candidate CF. The first step of the DRC algorithm is to generate a trivial composition while calculating how much the cost is minimized according to whether each CF includes a federation or not. Next, the DRC determines a new set of CFs that minimizes the cost by merging and splitting of federations.

An example of the refinement process is illustrated in Fig. 3. For the given candidate CFs, $\mathcal{F}$ in the candidate CF can be merged with another federation by merging or balancing, compared to the local current value of the $TotalCost$ function when α is fixed at 0.5. For example, in Fig. 3(a), $\mathcal{F}_k$ is merged with $\mathcal{F}_j$ when the following (5) is satisfied:

$$Cost_{previous} > Cost_{current} \qquad (5)$$

$Cost_{previous}$ is $0.5(t_{jk} + t_{ji})$, and $Cost_{current}$ is $0.5(h_k + t_{ji})$. Therefore, if $t_{jk} > h_k$, $\mathcal{F}_k$ and $\mathcal{F}_j$ are merged. In the other cases (Fig. 3(b) and Fig. 3(c)), if $(h_k+t_{ji}) > (h_i+t_{jk})$, $\mathcal{F}_k$ and $\mathcal{F}_j$ are split and $\mathcal{F}_j$ and $\mathcal{F}_i$ are merged, or if $t_{ji} > h_i$, $\mathcal{F}_i$, $\mathcal{F}_j$, and $\mathcal{F}_k$ are remerged.

IV. Experimental Evaluation

This section describes the experiments used in our evaluation. Our hierarchical federation system is built on HRTI, which provides the functionalities of simulation interoperation with respect to information hiding, as described in [5]. For our evaluation, we implement a prototype of the hierarchical federation system based on the HRTI and conduct experiments under a real distributed environment.

We compare the DRC composition with random composition (hereafter, referred to as **DRC** and **Random**, respectively). Our DRC algorithm, discussed in Section III-B, provides the optimal composition of federations. On the other hand, a random composition algorithm *randomly* composes the hierarchical federation system into composite federations. The latter approach is very simple and does not require system information collection except the constraint of the number of compositions.

Using the hierarchical federation composition results, we carry out experiments to evaluate the performance of the hierarchical federation system on a real network. For enhanced accuracy and reliability, we repeat the experiment more than 30 times. The 95 percent confidence interval is estimated for all experimental results. We experiment with different parameters of the hierarchical federation system.

We present experimental results from the prototype, which is implemented by HRTI and performed on a distributed environment. To evaluate the performance of our composition algorithm under a real network, we deploy the hierarchical federation system on a separate Local Area Network (LAN) testbed. Our experiments were conducted in an environment consisting of 8 PCs that were connected via a 1GB Ethernet LAN. The PCs were configured with a 2.67GHz Intel Core2 TM CPU, with 3GB of RAM under Microsoft Windows XP. The composite federations were allocated over the distributed nodes, and they were automatically executed at the beginning of the experiment.

Fig. 4 shows the performance comparison over time. In our evaluation, we assume that the coefficient α is fixed at 0.5 and we use this value for all the subsequent experiments. It compares the performance over time, where the threshold of request services is allocated as 10000 , the number of federates that join to each federation is 2.

The throughput of **DRC** is about 830 requests/sec, and that of **Random** is about 700 requests/sec in the number of federations, N=20. Similarly, **Random** at the number of federations, N=10 yields poor performance compared with **DRC**, as expected. Therefore, **DRC** performs better than **Random** (i.e., by 15.7% on average). This shows that, in terms of selecting a good composition candidate, the DRC algorithm outperforms the random composition, as it reduces the amount of inter-communication, which leads to enhanced performance, while achieving information hiding.

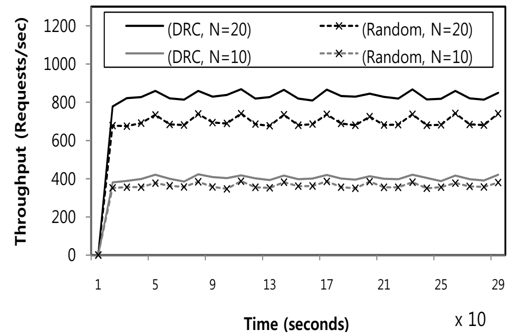

Figure 4. Performance over time.

V. Conclusion

This paper presents a composite federation framework to achieve a level of balance between information hiding and reducing the communication overhead. The proposed a hierarchical federation composition algorithm is, to the best of our knowledge, the first attempt to establish accountable security information in a secure distributed environment and it minimizes the critical security problem in HLA/RTI. Our experimental results show that the proposed composition algorithm performs better than a random composition algorithm in optimizing the composition of federations while enabling information hiding.

References

[1] IEEE Std 1516-2000, IEEE standard for Modeling and Simulation (M&S) High Level Architecture (HLA) - Framework and Rules, 2000.

[2] M. R. Stytz and S. B. Banks, "The distributed mission training integrated threat environment system architecture and design," *ACM Trans. Model. Comput. Simul.*, vol. 11, no. 1, pp. 106–133, 2001.

[3] S. J. E. Taylor, X. Wang, S. J. Turner, and M. Y.-H. Low, "Integrating Heterogeneous Distributed COTS Discrete-Event Simulation Packages: An Emerging Standards-Based Approach," *IEEE Transactions on Systems, Man, and Cybernetics, Part A*, vol. 36, no. 1, pp. 109–122, 2006.

[4] A. Boukerche, A. Mikkler, and A. Fabbri, "Resource control for large-scale distributed simulation system over loosely coupled domains," *J. Parallel Distrib. Comput.*, vol. 65, no. 10, pp. 1171–1189, 2005.

[5] J. H. Ahn and T. G. Kim, "Design and Implementation of Hierarchical RTI (HRTI)," in *Proc. European Simulation Interoperability Workshop (ESIW '09)*, 2009, pp. 111 – 121, 09-ESIW-016.