http://ansinet.com/itj

ITJ

ISSN 1812-5638

INFORMATION TECHNOLOGY JOURNAL

ANSInet
Asian Network for Scientific Information
308 Lasani Town, Sargodha Road, Faisalabad - Pakistan

Information Technology Journal 6 (8): 1270-1275, 2007
ISSN 1812-5638

Research of Updating Method of HLA Compatibility of Multi-Element Model in Simulation

[1]Xiangyang Liang, [1]Fengju Kang, [2]Lianjiong Zhong and [2]Rong Li
[1]School of Marine Engineering, Northwestern Polytechnical University, Xi'an, 710072, China
[2]School of Computer Science and Engineering, Xi'an Technological University, Xi'an, 710032, China

Abstract: With the simulation technique keeping rapid development, updating HLA-incompatibility simulation models to HLA compatibility, especially multi-elements simulation model, become a useful way to realize the aim and save the money and time for user. After introducing the HLA and HLA compatibility of model, in this paper, several facing problems during updating model including architecture, creating mode of data interface, simulation time driven mechanism, interactive mode of information and internal data structure were proposed. Aiming at these main problems, updating architecture should been done firstly from software and hardware which would reconstruct the connection modes among elements of simulation model. Partition of information flow module would distinguish the exchanging data between among elements and with outside of model and remove unnecessary sub-functions away from simulation model. Furthermore, it simplifies the difficulty of updating work and reduces workload which increases the updating project successful. The original time driven mechanism can accord to the HLA/RTI simulation clock by adjusting simulation time driven mechanism. Therefore the methods proposed in this study would improve universal and reusable of the simulation model of HLA incompatibility.

Key words: Updating method, compatibility, multi-element mode, HLA, distributed interoperability simulation

INTRODUCTION

With the simulation technology kept progress, using network to complete the more complex simulation mission becomes a popular simulation way. Multiple simulators in different place could join one simulation network with same standard. The High Level Architecture (HLA) is one of standards, which was initiated to support interoperability among distributed simulators. It defines a standard architecture for modeling and simulation of a complex distributed simulation by IEEE Std 1516-2000 (2000) and becomes the international standard, IEEE Std 1516-2000, for the distributed simulation.

However, the HLA is not the one and only simulation standard, there are a lot of simulators based on others standard and platforms have been created. Some of them can run all just by themselves and need not interact with other simulators; others can interact with each other in network but no based-HLA standard. If these simulators can be reused effectively and interact with each other in the same standard, there is no doubt that large numbers of time and money can be saved and the users will take great benefits. Here, a feasible approach to realize the goal is updating these simulators to according with HLA standard and then the simulators become HLA compatibility. Some researches about it have been done, which were shown from reference papers, but they are all based on one element model.

We focus on the methods of updating multi-element model to HLA compatibility, in this study. The multi-element model consists of several simulation elements. During updating to HLA-compatibility, more aspects such as architecture, internal data structure, simulation driven mechanism, data flow and control flow within elements will all need be adjusted and reconstructed. Obviously, it is more complex and difficult than that of one-element model.

RESEARCH OF COMPATIBILITY HLA-BASED

Introduction of HLA: The HLA is an architecture proposed for reuse and interoperation of simulation and brings a common framework to different simulation. The HLA is based on the premise that no simulation can satisfy all uses and users. An individual simulation or set of simulation developed for one purpose can be applied

Corresponding Author: Xiangyang Liang, School of Marine Engineering, Northwestern Polytechnical University, Xi'an, 710072, China

to another application under the HLA concept of the federation: A composable set of interacting simulations. A HLA-based simulation system is made up of one federation and several federates.

The HLA mainly comprises three parts. The HLA rules, which define interoperability and what capabilities a simulation must have to achieve it within HLA. The object model template, which is a format and method for specifying simulation data in terms of a hierarchy of object classes and their attributes and interactions between objects of those classes and their parameters. The Interface Specification, which is a precise specification of the interoperability-related services that a simulation may invoke, or be asked to provide, during a simulation execution.

Within HLA, a set of collaborating simulations is called a federation, each of the collaborating simulations is a federate and a run of the distributed simulation is a federation execution. Federates that adhere to the rules can exchange data defined according to the object model template by invoking the services defined in the Interface Specification. An HLA Run-Time Infrastructure (RTI) is a software implementation of the interface specification. The RTI actually provides the services defined in the Interface Specification, including services to start and stop a federation execution, to send data between interoperating federates, to control the amount and routing of data that is passed and to coordinate the passage of simulated time among the federates. The federates perform those functions by invoking the appropriate RTI services. The RTI may also invoke services that the federates must provide to it, such as receiving data; those services are likewise defined in the Interface Specification.

HLA compatibility of model: The HLA compatibility, just for federate, is discussing the problem whether the federate can join the federation to interoperate with other federate by the common standard including the interface specification OMT and HLA rules mentioned in above paragraph. Therefore a HLA-incompatibility model will become one or more federates to join the federation after it was updated to HLA compatibility.

Early the simulation system was HLA-incompatibility and created only for the specific application areas. The internal data structure, way of message interaction and simulation time driven mechanism were all different among these system. Each architecture of them run short of universal without the same technique framework, which make them much more difficult to reuse and interoperate with each other. However, HLA could bring the most compatibility when these HLA-incompatibility simulation systems get together because of flexible, universal and simple of HLA.

PROBLEMS DURING UPDATING MODEL

When a simulator of HLA-incompatibility was updated, we think there are some things need to be considered. First, there are two function modules in the system have to be divided completely. One function modules is exchanging messages between inside and outside of model, another is simulation realizing. Second, according the rules of HLA federate running, the model only calls for RTI services to exchange messages with outside of it. Any other methods are forbidden. Last, the logic of model action and coherence of space-time of simulation system must be same with original system after the updating work is finished.

From mentioned above, the problems are proposed during updating the model. Not only effect each other but also cooperate between them.

Architecture: Updating the architecture is the first step of update multi-element model. It is more complex because there are a lot of the ways of exchange messages between the elements. From the hardware, the elements can exchange messages with COM, LPT, USB and special Adapter. From the software, the ways include socket-based, model agent, middleware and gateway and so on. Both aspects need to be modified when the elements join the HLA.

In fact, not all elements need to join the federation as a federate sometimes. On the other hand, all elements would maybe divided completely to become independent federate node in others time. The method how to reconstruct the new architecture is determined not only by it self's structure character, but also need to reference driven mechanism, internal data structure and creating mode of data interface which will be discussed as follows.

Creating mode of data interface: In general, each multi-element simulator is a whole system and can complete its simulation work independent without any outside helps. As the multi-element simulator needs to join the HLA federation, the prevenient run-mode have to be destroyed in order to create a data interface in the system for exchanging messages of data and control with outside simulators. It is pay more attention to that two aspects both hardware and software need to be taken into account during creating the data interface.

Simulation time driven mechanism: As one of the core technology of DIS simulation, simulation time driven

mechanism is the most important run rules that ensuring the simulation could run normally. Driven mechanism of early development of simulation system include database model, fixed step model, variable step model and so on. However simulation of HLA-based is event-driven mechanism differing from that of ahead described. When transforming these different types of mechanism to HLA-compatibility, not only to make data flow layer analysis but also to make clear control boundaries for simulation model.

Interactive mode of information: Generally, in order to realize the information interactive between updated model and HLA network, the RTI API functions would be embedded in the internal software of simulation system. These RTI API functions are not difficult to learn and use. However in which position to embed the function in internal software? Which functions should be used? The timing and sequencing and conditions for the use function are main issues to be resolved. At the same time, using these functions is not permitted to affect basic logic rules and run efficiency of the original simulation model. It is more difficult part in updating work and requires that the program flow and data structure must be made corresponding adjustments.

Internal data structure: Design of data structure generally combines with arithmetic to satisfy the software requirement. When the driven mechanism and interactive mode etc. were changed, the data structure of system must be justified for keeping the correctness, stability and the high efficiency of original system. As a result of the superiority of HLA, the extent of justifying the data structure of system is small in fact. Sometimes it could be completed along with updating the mode of data interface or creating data interface.

KEY METHODS OF UPDATING MODEL

Updating architecture: Updating architecture is the first step of the whole updating work. Comprised with multiple elements, the multi-element simulator can run independently and show the characteristic of DIS (Distributed Interactive Simulation) among the elements as the HLA. However the way of exchanging message and data transmission protocol and simulation driven mechanisms between elements are so different with HLA that they can not be used directly. Many instances show that the updating work would become more and more difficult with the number of element increase.

In updating architecture work, the key work is distinguishing the data exchanging among the elements and the data exchanging between the element and other

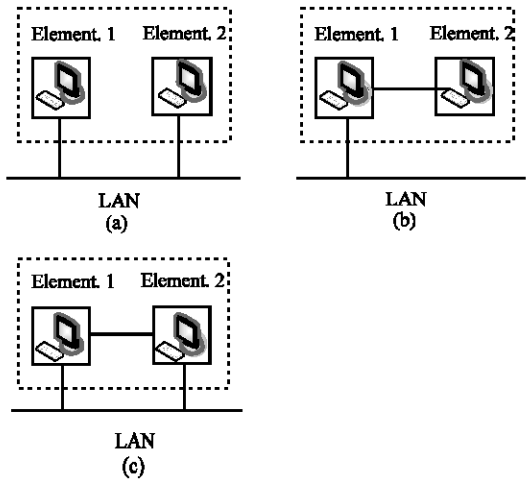

Fig. 1: Architecture of two-element simulation model

federates of HLA. The parts of system relating latter data must be updated for HLA-compatibility. And the data exchanging among the elements can still utilize the existing means of communication and protocol, which is not permitted to be opposite the driven mechanism of HLA. Then the whole multi-elements will join the federation smoothly and accurately.

For example of two-element simulation model, the possible architectures, after be updated, are shown in Fig. 1. From Fig. 1a, two elements of the simulator have been updated entirely and become two independent federates because the original means of exchanging data message between the two elements is replaced by that of HLA completely. From Fig. 1b, one element can interact with others federate by the HLA LAN, another element just exchange data messages with the first element without any links with HLA LAN directly. The emphasis of updating work is focused on the first element. And the mode and state of exchanging data messages can still keep existing way between two elements. From Fig. 1c, between two elements, parts of data messages can exchange by the existing way which is different with HLA, another parts of data message need to exchange with other federates using HLA LAN to transmit. The emphasis of updating work is focused on the data messages interacting with system outside, so the rest parts would be reused. It is also the most meet condition in updating multi-element model. Adopting the architecture would bring many benefits, which are not only reducing the workload of updating by keeping the original interactive mode in elements but also interacting with other federates of HLA LAN entirely. These benefits will simplify the updating work and improve work efficiency, finally promote the model's reuse.

Partition of information flow module: Information flow of system consists of two parts, control flow and data flow. Parting the two flow modules is the base to start other updating works.

Thought from software field, a kind of structure mode of system control flow composing is proposed, which is the control function module of system including three parts. First part is module of user control, through which users could put orders in simulation system and get the feedback information from simulation system. Second part is module of system control. In the system running-time, it creates the internal control orders and gets the feedback information. The last part is module of data processing whose function is receiving all control orders from other control modules and feedback the implementation conditions of these orders to them. The relation of three parts could be shown from Fig. 2. In HLA simulation, a lot of control messages of one federate come from HLA network and it also creates some control messages need to send through HLA network. Hence, during updating a simulation model, the different control messages between need net to exchange and no-need net to exchange could be parted clearly based on the structure mode, which make the target of updating work more simple and more exact.

Same to the partition of control flow module, from data flow an independent simulation system was partitioned to three major sub-function parts. They are the user-machine interface, information data sources and module of response simulation. The relation of them was shown in Fig. 3. Generally, an updated simulation model would retain the user-machine interface and one of other two sub-function parts. The function of removed sub-function part would be completed by other HLA federates through HLA network. In this study, the condition of retaining parts of sun-function and removing others was named software cut in simulation by us. Finished software cut in simulation successfully was based on partition of data flow module, which is also key technique to solving the problems about creating mode of data interface and interactive mode of information. Only clearly learn the relation of data flow, the needed sub-function parts could be retained entirely and no needed parts could be removed so as not to influence the simulation. Sometimes, an updated model must retain the whole three sub-function parts. However the sub-function part has been different with the original function. For example of information data source, it must create and receive only a part of information data comparing with original all information data, another data is sure to exchange with HLA federates through HLA network.

From researches about updating method done before, there are some good methods to be introduced, such as external programming interface (Li *et al*., 2005), COM

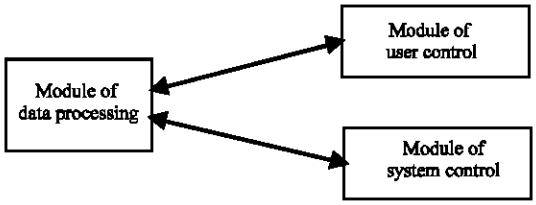

Fig. 2: Partition of control flow module

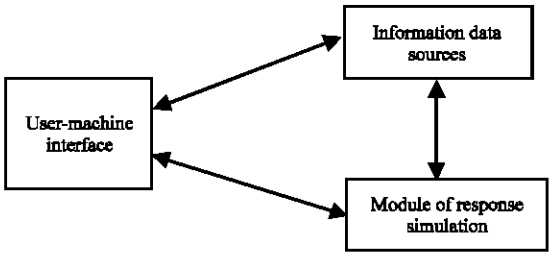

Fig. 3: Partition of data flow module

Server Component (Li and Guo, 2006), middleware of federate (Gong *et al*., 2005), gateway (Sheng *et al*., 2004), model agents (Wang *et al*., 2005a), HLA wrapper (Tian *et al*., 2002), technology of ActiveX control (Wang *et al*., 2005b), code translation (Mikel and Katherine, 2004) etc. In fact, using these methods need a precondition that is the work of partition of control flow module and data flow module has been finished.

Adjust simulation time driven mechanism: Simulation time driven mechanism is one of key aspects to ensure the simulation running normally. Early developed simulation system had multiform driven mechanisms that cause these systems not to interact each other. In order to realize the purpose of interact among them based on HLA, the time driven mechanism have to be adjusted to accord the RTI (Chai *et al*., 2002). In HLA/RTI simulation, driving simulation time is realized by calling time management API functions of HLA/RTI. So, the updated simulation model also call time management API function within internal simulation time driving of itself to ensure accord the RTI time after model joined the federation. To achieve synchronization time driving with RTI, the rules of original simulation time driving would be appended some constraints and conditions, which associate with time driven mechanism of HLA/RTI, to make itself could run according the request of rules of RTI completely. Through these constraints and conditions, model change its own time management messages to calling API function of RTI service, then callback the running result to itself and decide the next simulation driving time based on the RTI external clock and the internal clock.

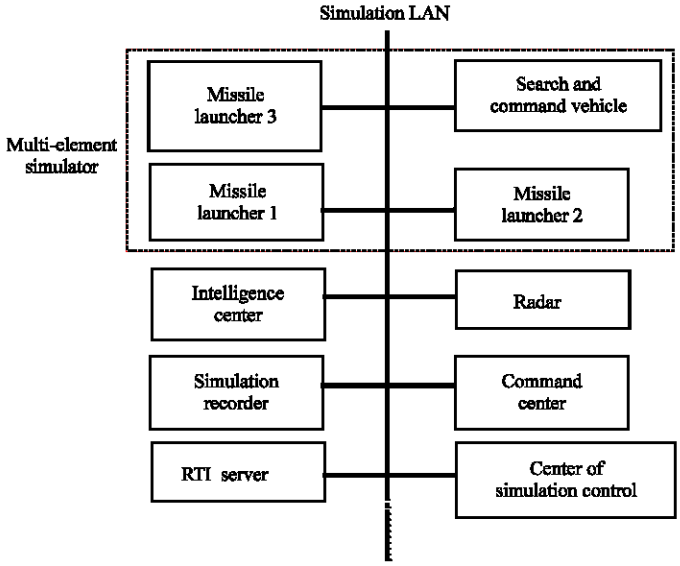

Fig. 4: HLA-based simulation of C4ISR missile air defense system

APPLICATION OF METHODS OF UPDATING MODEL

There is a HLA-based simulation of C4ISR missile air defense system as environment. The parts of federation structure are shown in Fig. 4. We used 1 PC for RTI (P-RTI1.3) and 5 PC for federates, including center of simulation control, command center, Intelligence center, simulation recorder and radar. Three missile launchers with a search and command vehicle is a set of independent training system of hardware in the loop simulation, which was developed by using OO technique and own it self's air defense information data source. The special adapter was used to link them each other. After be updated to join the HLA simulation, its air defense information would be offered by other federate instead of itself. During updating model, the four elements of original system would all link the HLA LAN with keep adapter linking each other. At the same time, air defense information data source of original was cut away and the data interface was created on the cut node to link HLA LAN. To reducing the times of calling RTI API functions, some small correlative data would be packed to one information units to send. For example the missile instantaneous parameter including longitude, dimensionality, height, direction, speed, rolling angle, pitching angle etc. The simulation driven mechanism also need to be adjusted and every advance step was admitted by order from Center of Simulation Control, which would make the simulation accord to the time step of RTI.

Through these methods, the messages among 4 elements (3 missile launchers with a search and command vehicle) could keep original communication means and other messages would exchange by HLA LAN, which not only reduce the updating workload to make updating process more easily but also furthest improve the simulating model reusability in fact updating project.

CONCLUSION

It is more significance that updating of HLA compatibility of multi-element model in distributed interaction simulation. Based on the characteristics of HLA compatibility of model, we proposed the main problems faced during updating multi-elements model to HLA compatibility. It is certain that data transmission, control process of multi-elements model is more complexity since it consists of multiple simulation elements. Generally, multi-element model simulator could run independently. So it's independent must be destroyed to bring itself to be one or more federates of HLA simulation by doing the software cut on it and creating the exchanging information with outside through the cut from simulation software methods. How and where to creating the cut is decided by several aspects including model architecture, mode of data interface, interactive mode of information and so on. Simulation time driven mechanism is another faced problem. A variety of driven mechanisms should all be adjusted for fitting the HLA/RTI simulation step.

From problems mentioned above, the methods of updating architecture, partition of control flow module and data flow layer were proposed with partition principle. At the same time, combining with the actual request of

simulation and system workflow of itself, the exchanging data between among elements and with outside would be distinguished and removing unnecessary sub-function which would be substituted by other federates of HLA simulation federation. Adding the constraints and conditions associating with time driven mechanism of HLA/RTI into original system simulation driven mechanism to synchronous the simulation clock between updating model and whole simulation federation. At last, a factual HLA compatibility updating procedure of training system of hardware in loop simulation was stated, which farther proved the reasonable and feasible of these methods in fact.

ACKNOWLEDGMENT

We would like to thank Mr. Tian Chuan of Northern Institute of Control Technology for providing us the experiment environment.

REFERENCES

Chai, X.D., B.H. Li and G.L. Xiong, 2002. Primary practice and further research on implementation technology of high level architecture system. J. Syst. Simulation, 14 (2): 152-155.

Gong, J.X., D. Wang and X.G. Qiu, 2005. Research on model's reuse in HLA federate. J. Syst. Simulation, 17 (11): 2652-2655.

IEEE Std 1516-2000, 2000. IEEE standard for modeling and simulation (M and S) high level architecture (HLA)-Framework and Rules.

Li, X.H., D.H. Sheng and Z.C. Sun, 2005. Design of compatibility transformation of the simulation/state flow simulation program. Comput. Simulation, 22 (5): 266-269.

Li, H. and B.B. Guo, 2006. Research on integration technology of HLA based on COM server component. Trans. Beijing Inst. Technol., 26 (2): 162-165.

Mikel, D. and P.L.M. Katherine, 2004. The computational complexity of high level architecture data distribution management matching and connecting processes. Simulation Modeling Practice Theory, 12 (2004): 217-237.

Sheng, D.G., X.G. Qiu and K. Zhang, 2004. Research on HLA compatibility reconstruction of simulation systems not basing on HLA. Comput. Simulation J., 21 (5): 97-100.

Tian, X.H., R.M. Feng and G.F. Weng, 2002. Change simulation/stateflow model to HLA federate. J. Syst. Simulation, 14 (7): 883-886.

Wang, Z.Y., Y.X. Ling and D.S. Qiu, 2005a. Principle and method for changing petri net model to HLA federate. Comput. Simulation J., 22 (1): 128-131.

Wang, T., L.X. Zhang and H.C. Liang, 2005b. Research on upgrading the conventional simulation system compatible with HLA. Trans. Shengyang Ligong Univ., 24 (1): 40-43.