A HLA-enabling Method in Collaborative Simulation Environment for Complex Product Development

Miaomiao Fan, Yushun Fan
Department of Automation, Tsinghua University, Beijing 100084, China
fanmm06@mails.tsinghua.edu.cn, fanyus@tsinghua.edu.cn

Keywords: HLA, multidisciplinary collaborative simulation, HLA-enabling, ABAQUS

Abstract. HLA-based collaborative simulation is an effective way to integrate different disciplinary models together for the design and development of complex product. In this paper, the HLA-enabling method based on HLA agent model is proposed to make models created through commercial software compatible with HLA standard. Next, a prototype system for forging manipulator system based on such HLA-enabling method is developed accordingly, where an approach of developing ABAQUS engine agent is proposed, which solves the difficulty of wrapping ABAQUS software. Finally, a case study on forging and manipulating process is conducted and verifies the high fidelity and effectiveness of our collaborative simulation technology.

Introduction

The design and development of a complex product is usually involved in multidisciplinary field knowledge, including mechanics, kinetics, kinematics and so on. As a single domain simulation cannot analyze the multidisciplinary behavior of a complex product comprehensively and accurately, it is indispensable to adopt multidisciplinary collaborative simulation, which is able to integrate models in different disciplines and simulation tools together, and facilitate better communication and cooperation among different development teams as well.

High Level Architecture (HLA) was initiated by the US Defense Modeling and Simulation Office (DMSO) in 1996. It defines the rules and interface specification to support reusability and interoperability among simulation object models or federates, and reusability and interoperability are the two most important features of HLA [1]. It consists of 3 parts: HLA rules, interface specification [2] and object model template (OMT). The HLA attempts to specify a standard interface between simulations and separates these interfaces from the implementation of any specific simulation. It defines an elementary simulation unit as a federate, and a set of federates working together as a federation. HLA, with minor revisions, has been adopted as an IEEE standard (IEEE 1516) [3] and is likely to be increasingly widely adopted within the simulation community.

This paper is organized as follows: The next section introduces the HLA-enabling method based on HLA agent model; In Section 3, the scheme of simulation engine agent is presented; In Section 4, a prototype system for forging manipulator is developed to verify the method proposed in this paper; A conclusion is given in Section 5.

HLA-enabling Method Based on HLA Agent Model
In nowadays, during the simulation fields, most of the analysis models are created through typical commercial software, such as Matlab, ADAMS. All of these simulation tools are weak in interoperation and do not support HLA standard and specification. In order to inherit and reuse the single domain model information and the modeling knowledge of experts, we propose a HLA-enabling method based on HLA agent model to transform original model, making them compatible with HLA standard. Under this situation, the whole transformed model can be directly integrated into HLA-based collaborative simulation platforms as a federate.

![Fig.1 Framework of HLA agent model](image)

The framework of HLA agent model is shown in Fig.1. HLA agent model mainly contains:

Simulation engine agent. It is used to operate on disciplinary model as well as specific software utilized to construct the model. It wraps the function including operation on model input and output, modification of model parameters, controlling of simulation process.

Federate object class. Corresponding to high level model, it inherits the function of simulation engine agent and RTI agent and is used to realize data interface mapping and interaction maintenance.

RTI agent. It defines a two-direction interface and realizes interoperation among different models through calling RTI service and RTI callback. This mechanism is implemented by RTI through two ambassador paradigms: RTIAmbassador and FederateAmbassador.

The principle of HLA agent model is: with the guidance of high level modeling result, different disciplinary federate object class realize the operation and maintenance on disciplinary model through simulation engine agent, where the interaction data during simulation runtime communicates with each other through RTI agent and thus multidisciplinary collaborative simulation is implemented.

**Simulation Engine Agent**

Simulation engine agent is relevant to specific simulation software and its implementation needs the secondary development interfaces provided by simulation software, while the rest part of HLA agent model have nothing to do with them, and the implementation is relatively permanent. Thereby, we mainly present the scheme of simulation engine agent.

In order to identify the function and boundary of simulation engine, we define simulation engine agent SimEngineAgent=<Model, Input, Output, Parameter, Operation, Configuration>, where:
Input and Output refer to the input and output data of simulation component and the input and output data of the operation method, which are the sets of ports data type.

Parameter refers to the parameters to be modified or the interim variables to be saved in the model.

Operation is the function called when accessing the simulation component. It can be further represented as Operation=<SetInput, GetOutput, ModifyParameter, SaveVariable, StartSimulation, SuspendSimulation, ResumeSimulation, TerminateSimulation, SimTimeAdvance>, in which the specific function includes setting input, getting output, modifying parameter, saving variable, start\suspend\resume\terminate simulation process.

Configuration refers to setting workspace path, work path, path of model and temporary data saved, etc.

Prototype System for Forging Manipulator

The forging manipulator system is one of the key equipments in the manufacturing industry, which is characterized by complex nonlinearity, strong coupling, multiple variables and multiple degrees of freedom. So far, the simulation on forging manipulator is limited in single disciplinary field, and therefore, HLA-based collaborative simulation techniques are studied to develop prototype system in this section, where the approach to develop simulation engine agent of ABAQUS is given and a case study on forging and manipulating process is carried on to verify our method.

Simulation Engine Agent of ABAQUS. The forging and manipulating process of forging manipulator system is involved in different domain models, including control, hydraulics, mechanics, deformation of workpiece and so on, in which the deformation subsystem is established by commercial FEA software. ABAQUS is one of the most advanced general nonlinear FEA software. As a result, it is preferred to work on the techniques for developing ABAQUS simulation engine agent.

According to the secondary development interfaces provided by ABAQUS [4][5], we adopt the kernel script interfaces embedded with object-oriented programming language python to fulfill the important operation of ABAQUS engine agent.

Operation Fulfillment. SetInput(parameter){ Modify InputFile }, where the modification of .inp file should conform to the format and specification of .inp file, and the time parameter also can be altered by such an operation.

GetOutput(parameter){ Exec python script }, that is to run python script file wrote by user self to access the data in result database .odb file. Data computed by ABAQUS is stored in the result database in object-oriented form. To access the result data, the path where the data is stored must be found in the first place, and then the data can be accessed by referring to the corresponding variable or through calling the methods of the data object.

The paradigm of ABAQUS engine agent is shown as Fig.2. The client application program is used as the schedule engine to call the ABAQUS engine agent to complete the standard API operation. ABAQUS engine agent is to call the background ABAQUS engine through the execution of ABAQUS internal command. ABAQUS engine is to complete the operation on data in workspace through calling the user subroutine.
As mentioned above, the forging and manipulating process is actually multidisciplinary, where exists the coupling between reaction force caused by workpiece deforming and clamps movement. When the reaction force reaches the given valve value of the cushion dashpot, it unlocks, and clamps moves backward. In the meantime, manipulator performs horizontal passive compliant movement. The movement of clamps is dependent on reaction force, and simultaneously the reaction force is influenced by the movement of clamps.

Based on the interaction, the federation model consists of several domains principally, including the mechanical multi-body dynamics model, the control model and the hydraulic servo model of the manipulator, the deformation model of workpiece. Here, the deformation model is created using ABAQUS while the rest of domain models are created using the same software Matlab/Simulink, and they are finally connected to create a complete model. As a result, the interaction is mainly between Matlab/Simulink model and the ABAQUS model. The interaction relation model is shown in Fig.3.

Fig.4 shows the simulation result of the force load that the forging workpiece acted on clamps in this process (the effect of forging process is assumed to be 1 second).

Compared to the peak value of horizontal load in the single discipline experiment, the value on this point of multidiscipline experiment decreases significantly, and it results from the passive submissive action of manipulator. Such an action could reduce the load largely acted on the manipulator, and therefore guarantee the protection of the manipulator. Collaborative simulation experiment reflects on the spot’s real working condition: cushion dashpot valves are forced to unlock twice so as to reduce the horizontal load on manipulator.
It can be seen that this experiment verifies the high fidelity and validity of the multidisciplinary collaborative simulation technology proposed in this paper.

Fig. 4 The horizontal load acted on clamps between the single discipline and the multidiscipline simulation

Conclusion

The HLA-based multidisciplinary collaborative simulation technology is widely adopted for complex product design and development. As the models created by commercial simulation tools do not support HLA standard, a HLA-enabling method based on HLA agent model is proposed in this paper. It is helpful to improve the reusability and interoperability of disciplinary model. More specifically, the development of simulation engine agent which is a vital part of HLA agent model is presented in detail; The application of the proposed method in forging manipulator system development is also implemented. In our future work, we will establish a HLA-based platform supporting WAN (Wide Area Network) simulation in combination with SOA (Service-Oriented Architecture) techniques.

Acknowledgements

Supported by National Basic Research Development Program (973 Program) of China, ID2006CB705407 and National R&D High-Tech Plan (863 Program) of China, ID 2007AA04Z150.

References