Realization of MATLAB Model Collaborative Simulation in Hopsan Environment based on Hopsan Framework

Xu Nanbo
Flight Physics Department, Shanghai Aircraft Design and Research Institute
xunanbo@comac.cc

Abstract. The design and simulation process of complex system is accelerated because of the general application of model-based system engineering method in industry. But new challenges arise along with it, such as joint simulation of multi-physical-domain models. Only by considering the coupling relationship among the physical domains, especially the Cyber-physical systems (CPS), can the simulation result be more consistent with the actual system so that can provide guidance for product design and manufacturing. Heterogeneous model is the key to solve the problem of multi-physical-domain coupling simulation [1]. Using tools integration[2] and semantic integration[3] are two mainly ways. Although the literature points out that using tools integration is challenging, the advantages of open source software integration at the source level make it possible for Hopsan software to realize joint simulation of heterogeneous models. This paper describes how to inherit a new one from the component class of Hopsan to realize the functions of second encapsulation of MATLAB model in the current Hopsan environment, the block diagram representation in the simulation schematic diagram, model scheduling in the simulation process, data input, output processing and data transmission, etc.

Keyword: multi-physical-domain coupling simulation, Cyber-physical systems

1. Introduction
Simulation models in different fields are generally built with professional simulation software. With the increasing demand of simulation in design, it is necessary to consider the influence between systems and the effect of multi physical field coupling. However, models developed by different disciplines and development tools have brought the following problems:

1. The users lack the understandable interface of unified operation model;
2. The model can only be self-closing and cannot be interconnected with models developed by other software, or customized secondary development work is required to realize interoperability. It makes the model difficult to share and reuse;
3. Large scale model database is difficult to build and maintain because of the large number of software and model file format;
4. Because of the high learning cost of simulation software in professional field, it is difficult for non-professional simulation engineers to understand the model and communicate with each other.

In summary, it is very important to organize, express and standardize the model. In order to realize the unified expression of the model, it is necessary to study the storage, operation and composition structure of the model, as well as the model management and model server architecture, etc. It is
suitable for Hopsan to be a platform software for verifying heterogeneous model encapsulation, as its characteristics of open source software. In this paper, Hopsan software is developed in depth based on the source code to verify the feasibility of integrating the model written by MATLAB m file and running calculation under its framework.

Hopsan software was a hydraulic system simulation software originally developed in Sweden in 1977, and it lasted for 8 years to public. The modeling method of Hopsan software is meta transmission line method, which originates from feature method and transmission line modeling. This method is especially suitable for parallel computing, so as to improve the computing speed and realize the distributed computing function. In the meta transmission line method, the variable time step method is added to solve the problem of system rigidity and breakpoint, and the meta transmission line method can describe the causal relationship between components. The software also has graphic modeling function, graphic component library and variable setting interface for simulation calculation. As shown in the Figure 1 below.

The idea of integrating MATLAB model in Hopsan is mainly realized by inheriting the component classes in Hopsan core. The relationship with the original Hopsan framework is shown in the figure 2 below.

Figure 1. Hopsan sofware interface diagram

The idea of integrating MATLAB model in Hopsan is mainly realized by inheriting the component classes in Hopsan core. The relationship with the original Hopsan framework is shown in the figure 2 below.

Figure 2. The relationship between Hopsan-GUI and Hopsan-Core

2. The basic idea of heterogeneous model encapsulation
In order to schedule the heterogeneous models in the complex system, a unified model description rule
is needed. The European FMI [4] [5] standard has been studied for many years. In this paper, the idea of FMI is used for reference and simplified properly.

Model is composed of algorithm and data. The algorithm exists in the form of a file, and the model description file is introduced to explain the data of the model, so as to effectively distinguish the algorithm and the model, and clarify the composition of the model. In this way, the model consists of four parts: model = algorithm + model description file + model standardization file + model data. The basic relationship is shown in Figure 3.

Figure 3. The basic relationship of Model

3. The realization of MATLAB file call in surrogate model

3.1 surrogate model

The development method of custom model is provided in the official documents of Hopsan. Mainly by integrating its component parent class and rewriting its virtual functions. These virtual functions include:

- void configure function. It is used to initialize variable members, register ports, parameters, etc. after the component model is created.
- void initialize function. Initialization function, which is called at the beginning of simulation, is not called after simulation.
- void simulateOneTimeStep function. Execute function, called once in each simulation step. In this function, the model developer is required to provide the code description of the model behavior, that is, algorithm implementation.
- void finalize function. The processing function after simulation is mainly used to clean up and release various resources allocated in initialization function.
- void deconfigure function. This function is called before the component is deleted.

The whole integration scheme is shown in the Figure 4 below. Hopsan core is the core part of Hopsan software processing simulation calculation. It sorts all components by analyzing the graphical simulation system schematic (directed graph), so that all models can be executed in one time step. Therefore, if the MATLAB model is called as one of the components, the above functions need to be implemented. But every matlab model is different, so we need to build a proxy between MATLAB model and Hopsan component. This agent mainly completes the following work:

1. The input, output and parameters of MATLAB model are transformed to complete the data transfer with other components;
2. Call MATLAB engine in the background to solve the MATLAB model.
3.2 Automatic compilation of surrogate model

Hopsan software is developed for C++, matlab agent model needs to be compiled into dynamic link library to be called by Hopsan. In order to facilitate the rapid import of MATLAB model into Hopsan, the model compilation process needs to be completed automatically. The automatic compilation process mainly includes the following steps:

1. Analyze the function structure in Matlab m file, obtain the input variable, output variable and call function;
2. Design the template file, and automatically complete the string replacement in the template file according to the analysis results of Matlab m file. Here, the third-party ntemplate library is used as the code generation engine.
3. Write makefile file and compile c++ source file into dynamic library by batch processing.
4. Generate the model description file HMF specified by Hopsan, and package and copy the compiled dynamic library, model description file and model block diagram file resources to the database directory specified by Hopsan.

The whole execution process is shown in the Figure 5 below:
4. **Cascade simulation of self-built model and MATLAB in Hopsan**

When the models are built in the above way, they will appear in Hopsan's model browsing panel.

5. **Conclusion**

Based on this method, a communication link system is constructed in Hopsan environment. The transmitter, spatial information, and receiver models are all written in MATLAB.m files. The signal analyzer model is a built-in Sink component of Hopsan. The system architecture is shown in Figure 6 below. Actual operation shows consistent results with simulation in MATLAB. It shows that centralizing MATLAB models in Hopsan can ensure the accuracy of the model, and provides solutions and methods for the integration of heterogeneous models.

---

**Figure 5.** The whole execution process
Figure 6. The system architecture

References
[1] System Design, Modeling and Simulation Using Ptolemy II, Edward Ashford Lee et al, Published by Ptolemy.org, 2014;
[2] Robert Braun, Liselott Ericson and Petter Krus, Full Vehicle Simulation of Forwarder with Semi Active Suspension using Co-simulation ,ASME/BATH 2015 Symposium on Fluid Power and Motion Control, October 12-14 , 2015 , Chicago ,USA ,2015.
[3] Sicklinger, S. and Belsky, V. and Engelmann, B. and Elmqvist, H. and Olsson, H. and Wüchner, R. and Bletzinger, K.-U. Interface Jacobian-based Co-Simulation. International Journal for numerical methods in engineering, March 2014.
[4] Functional Mock-up Interface for Model Exchange and Co-Simulation ,version 2.0,July 2014,https://www.fmi-standard.org.
[5] 1516-2000 IEEE Std. for Modeling and Simulation (M&S) High Level Architecture (HLA), 2000. http://standards.ieee.org/findstds/standard/1516-2000.html .