Modeling of an interoperability test bench for the on-board system of a train control system based on Colored Petri Nets

L. Yuan[1], T. Tang[1], K. Li[2] & Y. Liu[2]
[1]*State Key Laboratory of Rail Traffic Control and Safety,*
Beijing Jiaotong University, China
[2]*School of Electrical and Information Engineering,*
Beijing Jiaotong University, China

Abstract

The interoperability of train control systems is an essential feature for high-speed railways. It must be proven that the on-board system of the train control system has the ability to allow the safe and uninterrupted movement of each line, which accomplishes the specified performance. A third-party interoperability test bench should be built for the customer to test the interoperability of the on-board systems, which are manufactured by different appliers. In this paper, a formal model was applied on the design and the verification of the test bench. The design errors can be detected using this formal model, thus the correctness of the test bench functionality was ensured. A structured Colored Petri Nets model was proposed to describe the test bench in the aspects of system, modules and processes. The model includes three sub-models: test bench, interface and on-board system. Colored Petri Nets was used for system modeling and CPN-TOOLS was used to support the simulation and the formal analysis. The hierarchical modeling method not only reduces the complexity, but also enhances the reliability and reusability. On the basis of these models, the architecture, the information flow and the algorithms of the test bench can be verified during the system design and development. The simulation results showed that the design errors can be found and some algorithms can be verified and corrected in the modeling and simulation process.

Keywords: test bench, CPN, formal method, train control system.

WIT Transactions on The Built Environment, Vol 114, © 2010 WIT Press
www.witpress.com, ISSN 1743-3509 (on-line)
doi:10.2495/CR100261

1 Introduction

Train control systems based on communications are advanced signaling systems, which are an important part of high-speed railways, as they ensure the safety and the efficiency of the high-speed railway. Communication-based signaling systems control the train operation through radio communication, such as ETCS level 2 and the CBTC system. The high-speed trains, which are equipped with the on-board system, must be able to run on many lines of the high-speed railway network in the future in order to enhance the operational and deployment flexibility of the transportation network, for example, European Corridors and the DPL (Dedicated Passengers Line) of China. This requires the train control system to have the interoperability features. It must be proven that the on-board equipment has the ability to allow the safe and uninterrupted movement of high-speed trains, which accomplish the specified performance. There are some differences in the technical detail between each manufacturer, because of the different understandings of the specification, for example, the sequence of the message between the train and trackside. Therefore, an interoperability test is necessary to validate whether the on-board system can run on other lines, and this is used in tests in the reference laboratory.

Interoperability testing in the laboratory is different from testing by manufacturers. Interoperability tests are not based on manufacturers' design documents, but the specifications issued by the administration. The laboratory provides a test bench, which can be connected with the real equipment to run all test sequences, and therefore provide a standard environment to verify whether the on-board equipment meets the specifications. To verify the consistency between the functions of the equipment and the specifications, the interoperability test does not concern the internal implementation details of the equipment, but the external characteristics of the device. Therefore, the interoperability test is a third-party test, and it is also a test that mainly serves the users. The interoperability test bench should not only ensure the accuracy of the test, but also should prove the accuracy of itself for the authority of the laboratory. In addition, the test bench itself should be open, that is, the principle of the test bench is understandable. All of these are the requirements for the design and verification of the test bench.

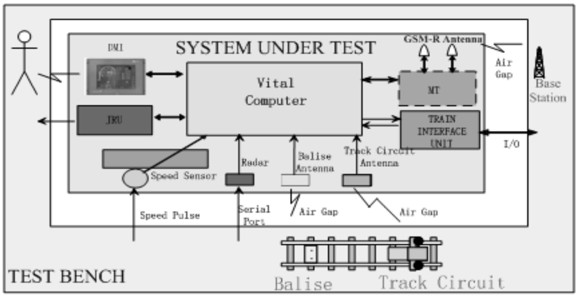

Figure 1: The interfaces between the test bench and the SUT.

Petri Nets is a formal, graphical modeling method. Petri Nets' advantages are its visual graphical modeling and its preciseness of theoretical analysis. So, it is widely used in various fields, especially in describing the complex systems or the logical relationship between processed activities, such as concurrency, competition, synchronization, etc. However, the Petri Nets models of a large-scale system will be too complex to analyze, and the correctness of Petri Nets models are based on the experience of the designer. The hierarchical model is a common way to solve the problem of the state space explosion of the Petri Nets. By hiding the internal structure of the subnet, the designer can focus on the higher abstraction level design and the subnets can be designed concurrently and reused easily, and the resulting model has a good hierarchy (Jensen [1]).

In this paper, a structured Colored Petri Nets model was used to describe the test bench and to decompose and refine it in the aspects of system, modules and processes. The model started from the system context level and described the interfaces and interactions between the test bench and the SUT. Then, the modules in the test bench are further refined in the second level, to show the state transition of the internal modules. In the third level, the module working processes are refined respectively. Colored Petri Nets was used for system modeling and CPN-TOOLS was used to support the simulation and the formal analysis. On the basis of these models, the architecture, the information flow and the algorithms of the test bench can be verified during the system design and development. The simulation results showed that the design errors can be found and some algorithms can be verified and corrected in the modeling and simulation. The models that describe the test bench can not only be used to prove the correctness of the test bench, but also be used to show the principle of the test bench to the people participating in the test (David et al. [2]).

2 Functional model of the interoperability test bench

2.1 The structure of the interoperability test bench and the basic principles of modeling

The interoperability of high-speed railways includes many aspects, e.g., vehicles, electrical system and operations. The interoperability in China focuses on whether the trains, which are equipped with different on-board equipment supplied by different manufacturers, can run continuously and safely on the line, which is equipped with trackside equipment from other suppliers, and meet the functional specifications and required performances of the system. Thus, the interoperability test concerns the ability to exchange information and to use the information that has been exchanged between trains and trackside. The interoperability test is focused on the external behavior of the on-board system. In addition, the interoperability test should not require manufacturers to complete some additional interfaces for the test. Therefore, all tests should be completed in the available interfaces of the SUT. In addition, the interoperability test should not require manufacturers to provide internal design documents. The test should validate the equipment only through its input and output behavior. Therefore, the interoperability test should be a black-box test and a data-driven test.

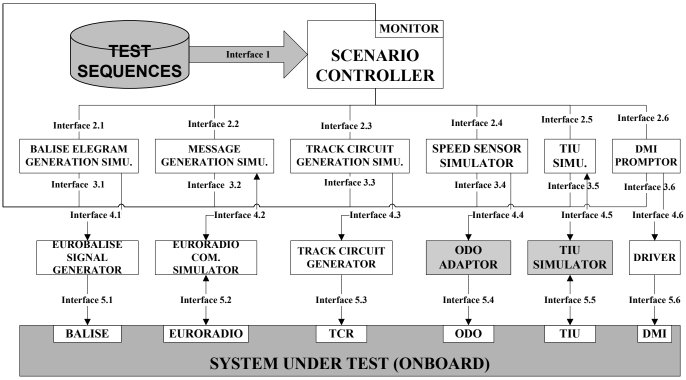

Figure 2: Structure of the test bench.

According to the on-board equipment specifications of the train control system, the structure of the interoperability test bench includes a test sequence database, a Scenario Controller module (SC), online executive modules and interface modules. During the execution of the test, the test bench will send test data to the SUT in real-time, according to test sequences data, thus creating an external running environment for the on-board system, so that the tested device's functions can be executed. The test sequences stored in the database are formed by concatenation of the set of test cases according to the test specification. The Scenario Controller is the main module of test execution, which is responsible for reading all the test data from the test sequence database, configuring other online executive modules, controlling the start and the end of the test, and monitoring the entire testing process. The online executive modules' responsibility is to generate messages according to the configuring data by SC, determining the proper time and occasion for sending messages to the on-board equipment. The Speed Sensor Simulator (SSS) simulates the dynamics behavior of the train and calculates the speed and the position of the train in real time. It provides not only speed information to the on-board equipment, but also position information of the train to other modules of the test bench. The interface modules are used to connect the test bench and the SUT, as the equipment to be tested may come from different suppliers; therefore, it needs to adapt the interfaces between the test bench and the real SUT, to ensure the communication between the test bench and the SUT.

Considering the functional decomposition of the test bench and the improvement of the maintainability and reusability, the test bench is divided into several functional modules, which can run on different computers. As the test bench uses the data-driven approach, all the modules work together to drive the on-board system in real-time through the test sequence data, so it is essential for there to be synchronization between the modules' internal states. To ensure functional correctness of the test bench, a formal method is used to model and analyze the test bench at the beginning of the design. In the analysis process of using Petri Nets for modeling, a hierarchical modeling approach is used, that is, the modeling process was divided into three levels: system level, module level and process level. This is not only to decompose the complex model, and simplify the problem, but also to model with Petri Nets throughout the whole process of the test bench's development. As the development of the software and hardware of the test bench is in accordance with the preliminary design, detailed design, module interface design, sub-module design, coding and debugging, modeling should be hierarchical to meet the phase of the development of the test bench. The hierarchy of the modules is:

(1) System level. This level includes all the components of the test bench. The components are taken as transitions and the information exchanged between modules are taken as places. The relationship and the information flow between the components are considered in this level.

(2) Module level. The transitions in the system level were refined in this level. The states and the transition of the states are considered in the modules. The internal states of the component are taken as places, and the events that triggered the state transitions are taken as transitions.

(3) Process level. The transitions in the module level were refined further here. The modules of this level are similar to the program function design of the components of the test bench.

Considering the network's hierarchy, the lower level network is actually the refinement of the higher level Petri Nets model. That is, if the total Petri Nets is N = (P, T, F) and the set Y is a transitions boundary set, N[Y] = (P[Y], T[Y], F[Y]) is a higher level module and the subnet, which just contains the elements of the set Y, is the lower level sub-model. In this way, the internal behavior of the subsystem has been further described in lower level models (Girault and Valk [3]). During the process of the development of the test bench, the module of the system level was designed to check whether the system design and the interface definition are correct. In addition, the structure and the functional partitioning of a component were focused in the module level modeling and the implementation of the functions of the component was focused in the process level modeling. By refining the transitions of a Petri Nets, we can gradually get the hierarchical Petri Nets model of the interoperability test bench to show the internal logic of reasoning and operation mechanism within the bench. Using the top-down modeling method, we can reduce the complexity of a system; make the model intuitive and easy to control and so on.

The sub-net model of a hierarchical Petri Nets model is actually the operational analysis of each subsystem, and the necessary parts of the whole Petri Nets model. To ensure the properties of the total Petri Nets model of the system, such as activity, boundedness and consistency, each sub-Petri Net model must satisfy the following conditions:

(1) If the places are removed, the structure of the network should be non-circular.

(2) Subnet models should be marking graphs.

2.2 System level models

Using the top-down hierarchical approach to develop the interoperability test bench, we should discompose the organizational structure and identify the important sub-modules and then describe the entire structure of the sub-modules, as well as the relationships of the sub-modules model, and finally refine the sub-models according to the requirements of the system.

In this level, we focused on modeling the interface between the various component modules of the test bench. With modeling, we defined the interface relationships and the data flow between the test bench and the SUT at different stages. The test bench module is divided into four sections: scenario controller, executive modules, interface modules and the SUT. The information exchanged between modules is also divided into the offline data and the online data. The offline data mainly refers to the configuration data of the test bench before the test. The online data mainly refers to the dynamic data of the modules in the process of the test. By modeling the system level, we can make a clear understanding of each module relationship in the configuration stage before the start of the test, and of the interaction relationship of each module in the process of testing.

As shown in Figure 3, the transitions in the shadow corresponded to the application logic of each module and the places in the shadow corresponded to the information exchange between the modules. In the model of this level, the messages, which were received from and sent to the external modules, are merged into a place; this means that we do not have to be concerned with specific information and can just focus on the source, the destination and the type of the data (i.e. offline or online).

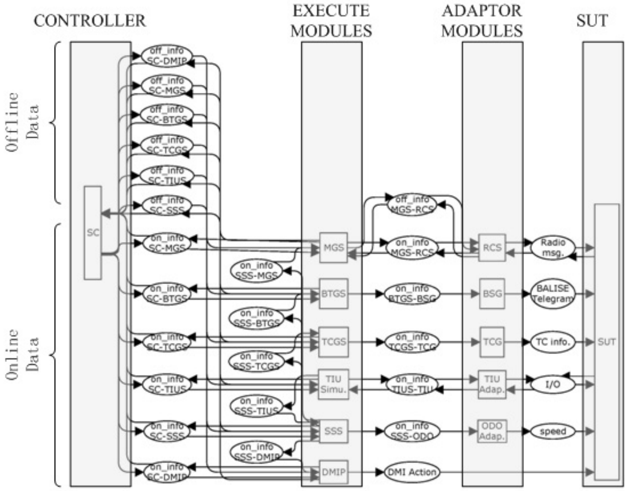

Figure 3: The system level model of the test bench.

The right part is the simulator of the on-board equipment. This simulator is a part of the test bench when we debug the test bench and the test sequences. So the test bench for the modeling process should be considered as the SUT behavior.

2.3 Module level models

Modeling in this level, the functions of each component were refined. The external interface of this component is refined to a message or information of train location. According to the state division of the test bench, there are initial state, waiting state, ready state, operational state and implementing state in the component, the name of which is MGS (MESSAGE GENERATION SIMULATOR), as shown in Figure 4. Each state corresponded to a place. The state transition was triggered by the Scenario Controller. The operation of the state corresponded to a transition. In this model, the transition will be refined in the next level to describe the detail functions. Figure 4 shows an example of the module of MGS.

2.4 Process level models

In the modeling of this level, the function of each module is refined in further detail. This model refines the upper network in order to refine the inner function of the component. The nets of this level do not consider the external interface, because the external interface level has been considered in the previous level model and the function, which is external interface communication, is ensured by the previous level model. The example of the model shown in Figure 5 is for the execution of the test sequence.

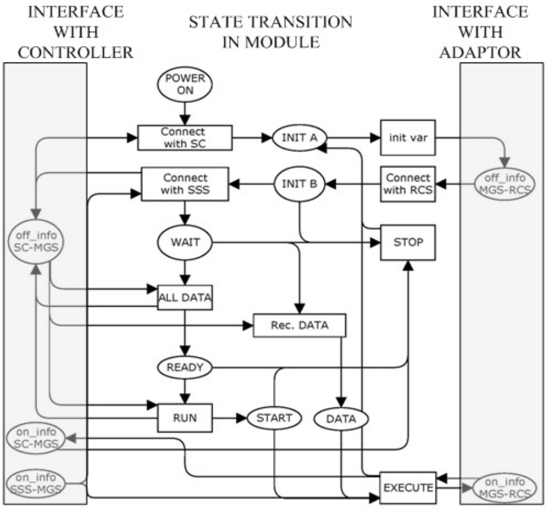

Figure 4: The model of the module level (MGS).

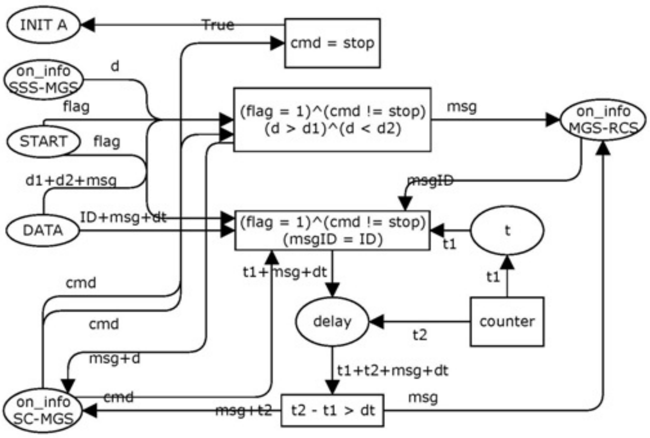

Figure 5: An example of the model of the process level.

3 State reduction and model analysis of the test bench

State space analysis is also known as Occurrence Graphs (OG) or Reach ability Graphs/Trees (RG/RT). State space analysis researches the accessibility of Petri Nets models to construct a directed reachability graph. The reachability graph included each state node, which is the reachability state place, and each arc that binds each element. State space analysis of the Colored Petri Net can be done through constructing the reachability graph of the reduced nets. In addition, for modeling with the hierarchical Petri Nets Model, we can analyze the reachability in the form of a reachability graph.

The correctness of the test bench design can be verified by researching the reachability of the model states. The reachability graph was constructed through the research of the transition path from any initial state and the research of all possible replacements. We can also construct state space by a fully automated tool, such as CPN TOOLS. The state space can explain many analysis and verification issues related to the system behavior. The state space analysis can validate whether the system owns the expected properties through the node, path and subnet forms. For example, there had been a design error in the early stage of the test bench design, which was a lack of an essential state, and the error was found through the deadlock of the state. Once the problem was identified, a new state was added into the model of the test bench design and the deadlock was eliminated, so the test bench design was improved.

The major restriction for the state space analysis is the dimension of the state space. The increase of Petri Nets model (such as the increasing of transitions) may lead to the exponential growth of the state space. To complete the model analysis, we should consider the reduction method to simplify the net model. First, the reduction rules were defined. Then, the places and the transition of the model were merged and eliminated through applying the rules to the Petri Nets model to simplify it. Reasonable reduction rules should be made so that the resulting model is still in keeping with the properties of the original model. The reduction rules included are as follows: merging continuous places and continuous transition, eliminating equivalent places and equivalent transition, eliminating self-circular places and self-circular transition, etc.

The model reduction is essentially a transform process to apply repeatedly the set of the reduction rules. Repeated application of reduction rules can maintain the feature we considered, until the system becomes irreducible. Reducing the model can mask some details that are irrelevant to the designer. The initial model of the test bench was reduced through applying these rules and this made it possible for the analysis to be performed using CPN-TOOLS.

After describing the test bench using formal model, the model checking must be done to verify the model. Assuming the model has a finite state space, model checking confirmed that the system will not execute against the state rules through detecting all the possible routes of the system state space. The system state of a reliable test bench should be unique at any time; therefore, the correctness and completeness of the state transition must be validated. For the test bench, the boundedness of the model and the equality of each state transition were mainly considered. The boundedness of the model shows that the resource of places was limited to avoid the system exception. In the reduced model, there is no specific description about the trigger conditions of the transition, because the model should describe abstractly the state transition of the test bench. Actually, each transition was bound with certain conditions and the sequence and the frequency of the transitions will affect the whole system. This will be shown in the status report as the fairness of the occurrence of the transitions.

The experience of the implementation of the test bench has shown that the model-based design and analysis method supported effectively the development of the test bench. The bug of the design was found in the early stage and the efficiency of the development was achieved.

Figure 6: The application for monitoring the execution of the test sequence.

4 Conclusion

The interoperability test bench is used to test the on-board equipment, which is from different manufacturers, and to validate whether the system meets the specifications. In this paper, a structured CPN model was proposed to describe and analyze the test bench during the development of the test bench. The model showed the behaviors of the test bench in the aspect of system, module and process. The model was simulated and verified with CPN-TOOLS to check the accessibility of the states and analyze the deterministic of the state transition; therefore, the correctness of the different designing stages was validated. Modeling provides the basis for the development and debugging of the test bench, and ultimately promoted the realization of the test bench. It also showed that this formal method can be used effectively for interactive system design. The models can be used for the system quality assurance and the system certification.

Acknowledgements

We wish to acknowledge the support of the National High-Technology Research and Development Program ("863" Program) of China No. 2009AA11Z221, National Science & Technology Pillar Program of China No. 2009BAG12A08.

References

[1] Jensen, K. Coloured Petri Nets. *Basic Concepts, Analysis Method and Practical Use (Vol.1-3).* Monographs in Theoretical Computer Science, Second Edition, Springer-Verlag, 1997.

[2] David V., Didier R., Morm B. A Petri Net based model for assessing OH&S risks in industrial processes: modelling qualitative aspects [J]. *Risk Analysis*, 2004, 24(6): 1719-1735.

[3] Girault C., Valk R. *Petri Nets for systems engineering: a guide to modeling, verification, and applications.* Publishing House of Electronics Industry: Beijing, 2005.

[4] Ma M., Chen G. Stochastic Petri-Net of auto-test system and performance evaluation. *Measurement & Control Techniques Journal*, Vol.25, No.10, pp. 19-2l, 2006.

[5] Cai J., Wang D., Li B. Extended hierarchical color petri net-based test case generation for composite services. *Journal of southeast university (Natural science Edition),* Vol.38, No.4, pp. 598-604, 2008.

[6] Hu J., Li H. Design & implementation of Petri-net-based coordinator in industrial hierarchical control scheme. *Computer Integrated Manufacturing Systems Journal,* Vol.13, No.12, pp. 2316-232l, 2007.

[7] Pan X., Li T., Lui Q. A Hierarchical model of Petri Net and a modelling tool for its design. *Computer Applications and Software Journal,* Vol.25, No.8, pp. 33-35, 2008.