SAFETY ANALYSIS OF ON - BOARD EQUIPMENT RBC HANDOVER FUNCTION BASED ON MULTI - AGENT AND HAZOP

GUIHENG HE, YADONG ZHANG, JIN GUO, SHUO WANG
The School of Information Science and Technology
Southwest Jiaotong University, Chengdu 610031, China

Abstract: *In CTCS-3 train control system, the RBC handover function of the on-board equipment seriously affects the operation efficiency and the safety of the train. In this paper, the on-board equipment RBC handover function is modeled by Multi-Agent, and then the formal analysis of the model is established. Finally, the HAZOP method is used to analyze the formal analysis and the hazard log table, which could provide a basis for risk management in full life cycle, is obtained.*

Keywords: *RBC handover, Safety analysis, Multi-Agent, HAZOP.*

I. INTRODUCTION

CTCS-3 (China Train Control System) uses GSM-R wireless communication to transfer information between on-board and wayside, and RBC (Radio Block Center) generates a MA to control the safe operation of the train according to the track circuit occupancy, route status, temporary speed limit and other information. In CTCS-3, lines are divided into several RBC sections. Therefore, the train should be able to achieve RBC handover function efficiently, safely and reliably when it running to the RBC boundary. The train in CTCS-3 should be controlled by RBC all the time. The RBC handover process will affect the efficiency and even the safety of the train, if the train does not be controlled by RBC or be controlled by multiple RBC. And the handover process of adjacent RBC control right is the weak link of RBC to train control. The RBC handover function of the on-board equipment in the RBC boundary is very important to the safety of the train.

In the process of RBC handover function, the information may be delayed, lost, errors, etc which will affect will affect the efficiency of RBC handover function and even affect traffic safety. Therefore, it is necessary to find out the security risk of the RBC handover through establishing a model of the on-board equipment RBC handover function and analyzing the security of the model.

II. MULTI-AGENT MODEL AND HAZOP

2.1. Multi-Agent Model

CTCS-3 is a complex system, it has complex structure and the internal subsystems in

This work was supported by the Key Research Projects of China Railway Corporation (2015X007-J and 2015X009-D) and the Fundamental Research Funds for the Central Universities (82014BR059)

CTCS-3 interact frequently, and the interaction process is complicated. Traditional modeling and simulation method are influenced by the theory of reduction and determinism obviously, whether the process simulation method based on the deterministic model or the statistical simulation method based on the probability model, the modeling and Simulation of the dynamic characteristics of the CTCS-3 level train control system can not be solved well. The modeling method based on Agent can combine the whole attribute of complex system with the individual behavior in complex system. This modeling method is an effective way to solve complex system and complexity.

Multi-Agent Model is based on signal Agent, this method abstracts the complex system into several Agent combinations, defines and describes the behaviors, relationships and interactions of these Agent, so that it can accurately complete the function and behavior of complex systems. This method provides an effective way for modeling and Simulation of complex systems. The method has a stronger ability of modeling and simulation of complex systems, a more abstract, which reducing the complexity of modeling structure and logic modeling compared with the traditional simulation technique[3].

Agent is a kind of subject which can continuously perceive the environment and act on the environment. The Agent can be in an environment and as a part of the environment. In order to achieve the design goal, the Agent can carry out the flexible behavior. The model structure of Agent is shown in the following figure.

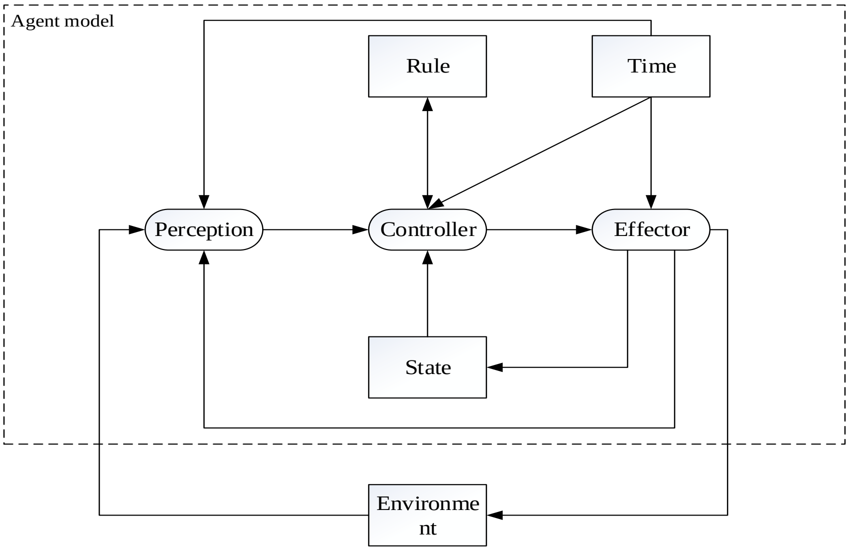

Fig 1. *The model structure of Agent*

In the model structure, the perception perceives the input information from the environment, and then transfers it to the controller. According to the information and the Agent internal state, the controller can select the appropriate rules and send the operation that Agent need to perform to the effector. According to the command given by the controller, the effector modifies the state of the Agent and outputs information to the environment. State is responsible for storing the state of Agent at this time. Time is the logical clock of Agent.

The structure of a single Agent can be divided into the careful thinking type, the reactivity type and the mixed type. The reactivity type Agent work through the reaction behavior of the external stimuli, it does not need to express and reason about the environment. The reactivity type Agent has good robustness and fault tolerance and simple interaction behavior of the reactivity type Agent can be combined into complex behavior. In addition, the reactivity type Agent does not have complex logical reasoning and the speed of execution is fast. According to the characteristics of reactivity type Agent, modeling the on-board equipment RBC handover function through reactivity type Agent is discussed in this paper.

2.2. HAZOP

HAZOP (Hazard and Operability Analysis) is a structured approach to identify design defects, process hazards and operational problems. Because it is able to analyze the system completely, HAZOP has become one of the most popular analysis methods in the field of risk analysis and is widely used in railway industry. In this paper, we mainly use HAZOP to identify the hazard identification of the RBC handover function model of the on-board equipment[4].

III. THE ON-BOARD EQUIPMENT RBC HANDOVER FUNCTION MULTI-AGENT MODELING AND IMPLEMENTATION

3.1. Multi-Agent model of the on-board equipment

The CTCS-3 level system is mainly composed of wayside equipment and on-board equipment. The on-board equipment adopts distributed structure, mainly including: C3-Kernel, C2-Kernel, DMI, TIU, BTM, RTM, GSM-R, SDU, TCR, JRU. The reference model of the on-board equipment is shown below.

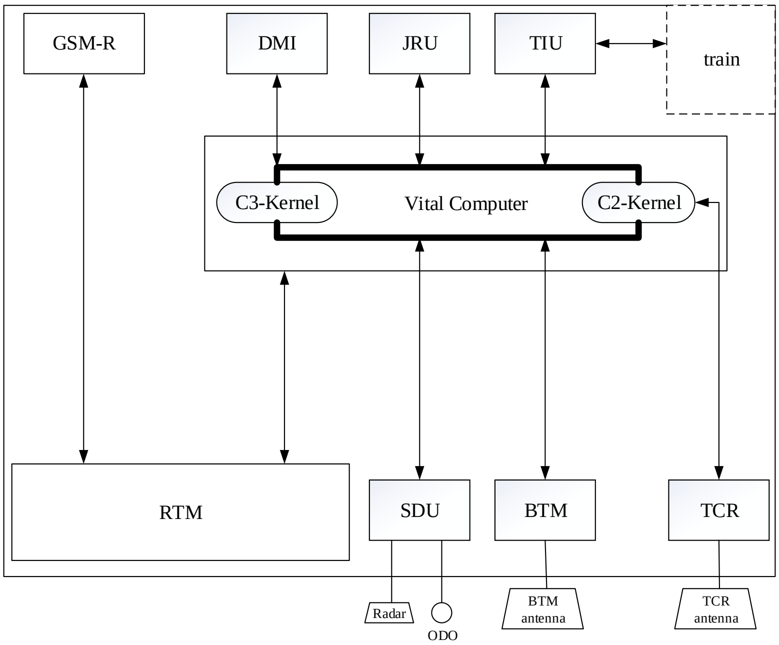

Fig 2. *The reference model of the on-board equipment*

According to the reference model, as well as the function and interaction behavior of the subsystem of the on-board equipment executing the RBC handover function. The on-board equipment can be abstracted into a Multi-Agent model, as shown in the following figure.

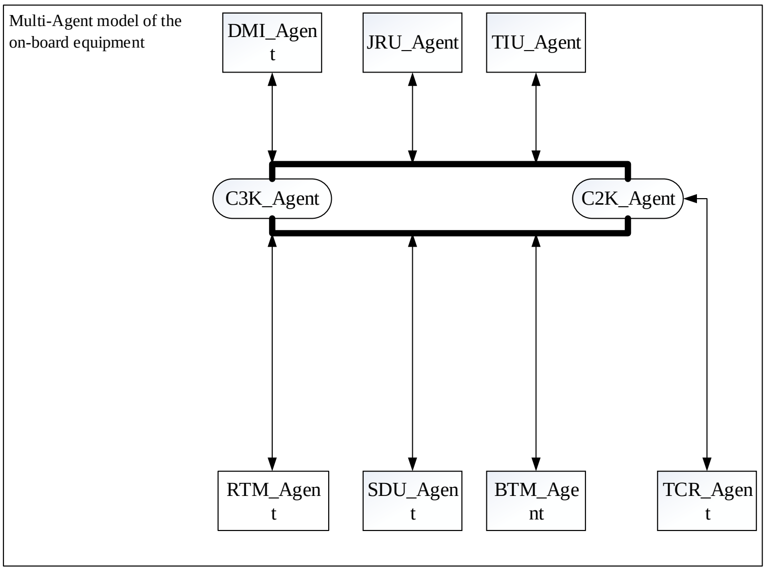

Fig 3. *Multi-Agent model of the on-board equipment*

The main functions of the main modules in the on-board equipment RBC handover function are as follows.

C3K_Agent: The core function module in the on-board equipment RBC handover function. C3K_Agent is mainly responsible for processing the information receiving from other Agent and controlling the on-board equipment to perform different actions.

C2K_Agent: In the on-board equipment RBC handover function, C2K_Agent is mainly responsible for receiving the track circuit code from the TCR_Agent, and then sending the track circuit code to the C3K_Agent.

RTM_Agent: In the on-board equipment RBC handover function, RTM_Agent is mainly responsible for the transmission of information from RBC to C3K_Agent and the transmission of information from C3K_Agent to RBC.

SDU_Agent: In the on-board equipment RBC handover function, SDU_Agent calculates the real-time speed and the distance of the train based on the pulse information generated by the speed sensor, and then reports to the C3K_Agent.

BTM_Agent: In the on-board equipment RBC handover function, BTM_Agent is mainly responsible for receiving information from balise and then sending the information to the C3K_Agent.

DMI_Agent: In the on-board equipment RBC handover function, DMI_Agent is mainly responsible for receiving train control information from C3K_Agent and then showing the information to the driver.

JUR_Agent: In the on-board equipment RBC handover function, JUR_Agent is mainly responsible for receiving the information from C3K_Agent and recording the information.

TIU_Agent: In the on-board equipment RBC handover function, TIU_Agent is mainly responsible for receiving the information from C3K_Agent and controlling the speed of the train according to the information.

From the above functional analysis, C2K_Agent, RTM_Agent, SDU_Agent, BTM_Agent, DMI_Agent, JUR_Agent and TIU_Agent play the same role in the on-board equipment RBC handover function. Their main function is to receive and send the information. As for the Multi-Agent model of the on-board equipment RBC handover function, we do not need to pay attention to the forms of information interaction and the process of encoding and decoding of information in the process of interaction. Therefore, we only need to analyze the content of information interaction among the Agents. Therefore, due to space limitations, in this paper, taking C3K_Agent and Agent RTM_ as an example, we can make formal analysis on the multi-Agent model of the on-board equipment RBC handover function.

3.2. Formal analysis on the RTM_Agent

The function of RTM_Agent in the multi-Agent model of the on-board equipment RBC handover function is transferring information between RBC and RTM_Agent, therefore, the RTM_Agent can be regarded as an information transceiver. The rule of RTM_Agent can be set to send what it has received to the corresponding target. The formal description of RTM_Agent can be defined as the following eight elements:

RTM_Agent==< RTM_time, RTM_input, RTM_output, RTM_state, RTM_Rule, RTM_GetENV, RTM_Control, RTM_action>

RTM_input represents the collection of RTM_Agent input messages, and the formal description of the RTM_input is shown below:

RTM_input=={MA, acknowledgement, position report, general message, initiation of communication session, system configuration, communication established, train data, train data confirmed, end of communication session, end of communication session confirmed}

RTM_output represents the collection of RTM_Agent output messages, and the formal description of the RTM_output is shown below:

RTM_output=={MA, acknowledgement, position report, general message, initiation of communication session, system configuration, communication established, train data, train data confirmed, end of communication session, end of communication session confirmed}

RTM_state represents the RTM_Agent state collection, and the formal description of the RTM_state is shown below:

RTM_state=={work, error}

Using RTM_GetENV to indicate the RTM_Agent perception, which is used to receive and process input messages from the environment, and then send the message to the RTM_Agent controller. The formal description of the RTM_ GetENV is shown below:

RTM_GetENV==[env?: Environment
Perceiver: Environment→RTM_input
result!: RTM_input|
∀ per: Perceiver · ran(per)=dom(per)
result!=ran(per)]

Using RTM_Control to indicate the RTM_Agent controller, it is a rule processing system. According to the input message and the internal state of the RTM_Agent, RTM_Control can select the appropriate rule and determine the effect of the effector. In the Multi-Agent model of the on-board equipment RBC handover function, we only need to pay attention to the content of information interaction among the Agents, therefore, the RTM_Control can be set to send the information same with the information it has received. The formal description of the RTM_ Control is shown below:

RTM_Control==[in?: RTM_input
out!: RTM_output
S:RTM_State
RTM_Rule:(in?×S) →(out!×S)|
∀rule: RTM_Rule · ran(ran Rule)=ran(dom Rule) ∧dom(dom Rule)=dom(ran Rule)
out!=dom(rule(in? α S)]

Using RTM_action to indicate the RTM_Agent effector, it can determine the output of the RTM_Agent and the changes of the RTM_Agent state according to the rule given by the controller of the RTM_Agent. The formal description of the RTM_ action is shown below:

RTM_action==[ΔRTM_Agent
env?: Environment
in: RTM_input
out!: RTM_output|
in = GetENV(env?)
in∈RTM_input
out!= Control(in)
RTM_state′=ran(ran RTM_Rule (in α RTM_state))]

RTM_action can also be regarded as the action function of RTM_Agent, which can represent the whole execution process of RTM_Agent from receiving information to sending information.

3.3. Formal analysis on the C3K_Agent

C3K_Agent is the most important and most complex part of the Multi-Agent model of the on-board equipment RBC handover function. It needs to send the information selectively according to the different information received, so it has a relatively complex rule definition. C3K_Agent can also be formalized as the following eight elements:

C3K_Agent==< C3K_time, C3K_input, C3K_output, C3K_state, C3K_Rule, C3K_GetENV, C3K_Control, C3K_action>

C3K_input represents the collection of C3K_Agent input messages, and the formal description of the C3K_input is shown below:

C3K_input=={Preview information, MA, BTM report, position report, general message, call RBC, system configuration, train data confirmed, end of communication session, end of communication session confirmed}

C3K_output represents the collection of C3K _Agent output messages, and the formal description of the C3K _output is shown below:

C3K_output=={MA, acknowledgement, position report, general message, initiation a communication session, communication established, train data, end of communication session, end of communication session confirmed}

C3K_state represents the C3K_Agent state collection, and the formal description of the C3K _state is shown below:

C3K_state=={work, error}

C3K_Rule represents the C3K_Agent rule collection. Depending on the input message, the output message is different. The formal description of the C3K _Rule is shown below:

C3K_Rule=={(MA α work)→(work α acknowledgement)
(BTM report α work)→ (work α position report)
(call RBC α work)→ (work α initiation a communication session)
(system configuration α work)→ (work α communication established)
(position report α work)→ (work α position report)
(general message α work)→ (work α acknowledgement)
(train data α work)→ (work α train data confirmed)
(session management α work)→(work α end of communication session)}

Using C3K_GetENV to indicate the C3K_Agent perception, which is used to receive and process input messages from the environment, and then send the message to the C3K_Agent controller. The formal description of the C3K_ GetENV is shown below:

C3K_GetENV==[input?: Environment
Perceiver: Environment→C3K_input
result!: C3K_input|
∀per: Perceiver · ran(per)=dom(per)
result!=ran(per)]

Using C3K_Control to indicate the C3K_Agent controller, it is a rule processing system. According to the input message and the internal state of the C3K_Agent, C3K_Control can select the appropriate rule and determine the effect of the effector. The formal description of the C3K_ Control is shown below:

C3K_Control==[in?: C3K_input
out!: C3K_output
S:C3K_State
C3K_Rule:(in?×S) →(S×out!)|
∀rule: C3K_Rule · ran(ran Rule)=ran(dom Rule)
out!=dom(rule(in? α S)]

Using C3K_action to indicate the C3K_Agent effector, it can determine the output of the C3K_Agent and the changes of the C3K_Agent state according to the rule given by the controller of the C3K_Agent. The formal description of the C3K_ action is shown below:

C3K_action==[ΔC3K_Agent
env?: Environment
in: C3K_input
out!: C3K_output|
in = GetENV_C3K (env?)
in∈C3K_input

out!= Control_C3K (in)

C3K_state′=ran(ran Control(in))]

C3K_action can also be regarded as the action function of C3K_Agent, which can represent the whole execution process of C3K_Agent from receiving information to sending information.

The following is a brief description of the previous use of the symbol. ? represents the input variable; ! represents the output variable; → represents the corresponding relationship; Δ represents the change of state; ran is said to get the range; dom is said to get domain of definition[5].

3.4. Formal analysis on the implementation process of the on-board equipment RBC handover function

According to the process of the on-board equipment RBC handover function, the whole process can be divided into 3 steps to carry out the formal description. First, we can have a formal analysis on the process of the train passing through LAT. After the train pass through LAT, BTM_Agent and SDU_Agent collect the position information of the train and then report to the C3K_Agent. After receiving the position information, C3K_Agent sends the position information to RBC1 through RTM_Agent. Then, RBC1 sends a position reference point adjusted MA to C3K_Agent through RTM_Agent. After receiving the MA, C3K_Agent sends a acknowledgement to RBC1 through the RTM_Agent. The formal description is shown below. Among them, BTM_action represents the action function of BTM_Agent, SUD_action represents the action function of SDU_Agent and so on.

<Agent_process>::=(Do(position report))

|INFORM

def position report{

∃ env1, env2: Environment· (env1=BTM report)&&(env2 =position report) →

(BTM.BTM_action(env1)&&SDU.SDU_action(env2)) → (C3K.C3K_action(BTM report)&& C3K.C3K_action(position report)) →RTM.RTM_action(position)

∃ env3: Environment· (evn3 in RTM.RTM_input)

∃ input: RTM_input · input=RTM.RTM_action(evn3)

if(input=MA)→C3K.C3K_action(input) →RTM.RTM_action (acknowledgement)}

Second, according to the telephone number provided by RBC1, C3K_Agent sends initiation a communication session to RBC2 through RTM_Agent. Then, RTM_Agent receives system configuration from RBC2 and sends it to C3K_Agent. Then, C3K_Agent sends communication established to RBC2 through RTM_Agent. At the same time, RTM_Agent receives the train data confirmed sent by the RBC2, and sends to the C3K_Agent. The formal description is shown below:

<Agent_process>::=(Do(call RBC))

|INFORM

def call RBC {

if (∃env4: Environment· (env4=call RBC)) → input: BTM_input· input=BTM.BTM_action(env4)

if (input=call RBC) →C3K.C3K_action(input) →RTM.RTM_action(initiation a communication session) →RTM.RTM_action(communication established) →C3K.C3K_action(System configure)

→RTM.RTM_action(communication established) →RTM.RTM_action(general message) →C3K.C3K_action(general message)→RTM.RTM_action(acknowledgement)→RTM.RTM_action(train data confirmed) →C3K.C3K_action(train data confirmed) →RTM.RTM_action(general message) →C3K.C3K_action(general message)}

Finally, after the max safe front end passes RN, BTM_Agent and SDU_Agent receives position report from the environment and then sends it to C3K_Agent. C3K_Agent sends position report to RBC1 and RBC2 through RTM_Agent. Then, RBC2 sends MA to C3K_Agent through RTM_Agent. After the min safe rear end goes through RN, C3K_Agent sends position report to RBC1 and RBC2 through RTM_Agent. And then, C3K_Agent receives session management from RBC1 through RTM_Agent. Then, C3K_Agent sends end of communication session to RBC1 through RTM_Agent and receives end of communication session confirmed from RBC1 through RTM_Agent. The formal description is shown below:

<Agent_process>::=(Do(RBC handover))

|INFORM

def RBC handover{

∃env7, env8: Environment·(env7= balise switching report)&&(env8 =position report) →(BTM.BTM_action(env7)&&SDU.SDU_action(env8)) →(C3K.C3K_action(train head over balise switching report) →RTM.RTM_action(position report) →RTM.RTM_action(MA)→C3K.C3K_action(MA)→RTM.RTM_action(acknowledgement)

∃env9, env10: Environment·(env9= balise switching report)&&(env10 =position report)→(BTM.BTM_action(env9)&&SDU.SDU_action(env10))→(C3K.C3K_action(train tail over balise switching report)→RTM.RTM_action(position report) →RTM.RTM_action(session management)→C3K.C3K_action(session management)→RTM.RTM_action(end of communication session)→RTM.RTM_action(end of communication session confirmed)}

IV. RISK ANALYSIS OF THE MULTI-AGENT MODEL FOR THE ON-BOARD EQUIPMENT RBC HANDOVER FUNCTION

The safety of the system is closely related to the hazard identification, the more comprehensive the hazard identification, the more it can ensure the safety of the system. Therefore, the analysis of the on-board equipment handover function model in different views can fully identify hazards of the system. In this paper, the system is analyzed from 3 aspects of hardware structure, function and implementation process. Taking RTM_Agent as an example, we choose RTM communication board as the element and combine the guide word with the element. The results are shown in table 1.

***Table 1**. Hazard log table of RTM_Agent structure*

Number	Node	Guide Word	Element	Deviation	Cause	Result	Recommended Measure
1	RTM_Agent	In error	RTM communication board	RTM communication board error	Poor reliability of RTM communication board	Unable to receive MA and other information, affecting traffic safety	Select the higher reliability communication board

We choose RTM_GetENV as the element and combine the guide word with the element. The results are shown in table 2.

***Table 2**. Hazard log table of RTM_GetENV*

Number	Node	Guide Word	Element	Deviation	Cause	Result	Recommended Measure
1	RTM_GetENV	No	RTM_GetENV	Unable to get input information	Program vulnerability	RTM_Agent could not get information, affecting traffic safety	Standardized program design
2	RTM_GetENV	In error	RTM_GetENV	Input information error	Program vulnerability	RTM_Agent receives wrong input information, affecting traffic safety	Standardized program design
3	RTM_GetENV	Part of	RTM_GetENV	Input information incomplete	Program vulnerability	The input information is not complete affecting the traffic safety	Standardized program design
4	RTM_GetENV	Later	RTM_GetENV	Input information delay	Program vulnerability	Input information delay, the train may be speeding	Standardized program design

We choose RTM.RTM_action(position report) in implementation process as the element combine the guide word with the element. The results are shown in table 3.

***Table 3**. Hazard log table of RTM.RTM_action(position report)*

Number	Node	Guide Word	Element	Deviation	Cause	Result	Recommended Measure
1	RTM.RTM_action(position report)	No	RTM.RTM_action(position report)	RTM_Agent is unable to report position to the RBC	RTM transmission channel error	Can not perform RBC handover function, affecting traffic safety	regular maintenance
2	RTM.RTM_action(position report)	In error	RTM.RTM_action(position report)	The position report is error	Program vulnerability	Can not perform RBC handover function affecting normally, causing emergency stop	Standardized program design
3	RTM.RTM_action(position report)	Part of	RTM.RTM_action(position report)	The position report is not complete	Program vulnerability	RBC could not determine the location of the train, causing emergency stop	Standardized program design
4	RTM.RTM_action(position report)	Later	RTM.RTM_action(position report)	RTM_Agent reports position to RBC too late	Program vulnerability	Can not perform RBC handover function in RN, affecting traffic safety	Standardized program design

V. CONCLUSION

In conclusion, this paper presents the formal description of the on-board equipment RBC handover function Multi-Agent model, performs risk analysis of the model by HAZOP method, and finally carries out a comprehensive and reliable hazard identification of on-board equipment RBC handover function model.

References

[1]. *Zhang Shu-guang*. Overall Technical Scheme of CTCS-3 Level Train Control System [M]. Beijing: China Railway Press, 2008.

[2]. *Zhang Xin-ming, Yu Zhi-yang, Yuan Huan-jing*. The Function Requirements Analysis of CTCS-3 Train Control System Based on the Train Operation Scenarios [J]. Railway Signalling & Communication, 2010, 46(4): 17-21.

[3]. *Liao Shou-yi, Wang Shi-cheng, Zhang Jin-sheng*. Agent-based Modeling and Simulation for Complex Systems[M]. Beijing: National Defense Industry Press,2015.

[4]. *Cui Jun-fei, Zhang Ya-dong*. Research on Hazard Identification of Train Control Center [J]. Railway Signalling & Communication, 2013, 49(7): 11-13.

[5]. *Feng Mei*. Behavior Research and Application of Agent[D]. Master's degree thesis of Shandong Normal University. 2001♦