Abstract

Recently, many critical control systems are developed using formal methods. When software applied to such systems is developed, the employment of formal methods in the software requirements specification and verification will provide increased assurance for such applications. Earlier error of overlooked requirement specification can be detected using formal specification method. Also the testing and full verification to examine all reachable states using model checking to undertake formal verification are able to be completed. In this paper, we propose an eclectic approach to incorporate Z(Zed) formal language and ‘Statemate MAGNUM’ which is formal method tools using statechart for applying to the train control systems.

1. Introduction

Recently, in accordance with the development in computer and telecommunication technologies, the existing mechanical and electric train control systems in the railroad system are being changed to electronic ones. As train control systems were being computerized like this, it became very difficult to secure the safety of system. Especially, the train control systems are required to keep high level of safety as they are very vital equipment responsible for the safe operation of train. As for the plan to solve these problems, the relative international standards recommend to apply formal methods in specifying development specifications and design for train control systems [1, 2], and the research and efforts to apply this formal method are being progressed by centering on the advanced countries in railroad industry, especially Europe [3]-[6].

Among them, the case of applying formal specification of train control systems to the #14 subway line of Paris, France is introduced as the first case of applying formal method to the actual commercial system [3]. The studies to apply various formal methods such as Z(Zed), VDM(Vienna Development Method), Statechart, etc. as well as B method in [3] to train control systems had been progressed already or are in progress now, but it is the unprepared in practice situation yet to utilize them for the development of actual railroad sites. Because, the application plan of formal method presented as a result of studies now is based on the excessively complex mathematical formulas or theories, and therefore, it has problems that the specialists in that field are needed and very long development time is required, etc.

Moreover, in Korea, the research stage to apply formal methods for the development of train control systems is very far behind in comparison with that of advanced countries, and though there are some cases applying formal methods to the design of railroad signaling protocol [7], there is no actual situation of application for the train control system yet. Currently in Korea, although it is already recognized at research level that the formal method is essential method for securing the stability of software, it is required to conduct researches on realistic and material procedures and methods enabling to apply it to the railroad field since the application plan for it has not been raised as issue in the industry actually. Accordingly, this thesis presented the specific plan and procedure to apply formal methods in practice to the train control systems in railroad field for the first time in Korea.

For this purpose, this thesis proposed the approach applying Z which is based on the mathematical form only for the essential part while using the Statechart which is one of the languages for the graphical formal specification, rather than depending on the excessive mathematical formula which is one of the drawbacks of formal method application. The order of this thesis is as follows: After describing the explanation and needs for the formal specification and formal verification composing of the formal method in chapter 2, it will present the application plan for domestic train control systems using the compromised plan of Z and Statechart concretely for which we would like to propose in chapter 3. Besides, after showing the results of formal specification using the method of our proposal, we would like to conclude in chapter 4.
2. Formal Specification and Formal Verification

The train control systems are in charge of the speed control and route control, etc. of train, and they are vital systems responsible for the final safe operation especially by taking in charge of functions preventing head-on/rear-end collisions of train. Recently, train control devices to be used in railway signaling systems are being changed from past mechanical/electrical types to electronic types thanks to the development of electronics, computer and communication technologies. Accordingly, the composition of logic circuit in the railway field is being moved from the one using traditional relay to that using microprocessor. The improvement plan on the safety, reliability by utilizing hardware can be applicable in part in the composition using microprocessor, but for the most cases, it is to be relied on the software. In accordance with the increased use of computerized train control systems, large scale human casualties or economic losses can be occurred through the failure of systems resulted from the error due to the complexity of software.

Therefore, the safety-critical system such as railroad field studies the formal method as one method for guaranteeing the accuracy and reliability of software. The formal method is a kind of the software engineering, and its purpose is to enhance the reliability of system by designing errorless system. Since this formal method uses mathematical symbols, it can reduce the vagueness or uncertainty which might be caused by natural language in preparing the details of system to the minimum, and it can be verified by using its mathematical nature whether it is identical to the designed requirement of user. And it can be divided largely into the formal specification and the formal verification. The purpose of formal specification is to depict the requirement to be accomplished by the system and the design satisfying such requirement, and the formal verification means a series of process testing whether the design is satisfying such requirement.

![Figure 1: Procedure of formal methods](image)

The formal specification refers to the method eliminating the ambiguity and uncertainty of natural language specification by specifying it through using the mathematical logic symbol formula or formal diagram with definite meaning, etc. As for the formal specification languages, the language using mathematical symbols and letters such as logical or process algebra, and the schematic specification language such as state-diagram can be given largely as examples.

The formal verification refers to the test on whether the system design and requirement which were prepared by using formal specification technique as shown in Figure 1 are being satisfied or not by using mathematical and logic verification methods. There are largely two types of formal verification techniques; one is the model checking method and another is the theorem proving method. Model checking is the technique to verify parallel systems in infinite status automatically and to prove mathematically that the desired results can be achieved at all times. Besides, the theorem proving method is the technique to prove mathematically by using logical mathematics formula, and since it has the demerits such as time-consuming for use and need for specialists, the model checking method which is easier to use than theorem proving method is being used more preferably in general.

3. Formal Method Application Plan for Domestic Train Control Systems

Every formal method has its unique applicable area, and the maximum effects can be achieved if only used to suit for this area. In addition, as mentioned in the foregoing section also, the convenience of using shall be considered for the actual applicability, and various considerations such as selection of reliable tool supporting the formal method, etc. In this study, based on the formal specification
languages shown in Figure 2 and the tool selection basis to support it, we had compared and analyzed various formal methods to adopt any formal method suitable for the vital software of domestic train control systems and announced some of its results [9].

As results of comparison and analysis on the basis of these selecting criteria for the formal method, this thesis selected Z and Statechart as a method for the formal specification, and selected ‘Statemate’ which is based on the Statechart for the formal verification. The compromised plan of Z and Statechart proposed in this thesis can accomplish very high integrity when preparing the requirement specifications such as system definition at the formal specification stage compared with other existing methods, and at the same time, it has great advantages to shorten the development time through convenience of using due to the formal specification language based on graphics.

3.1 Analysis on the compromised plan of Z and Statechart

While Z has great merits of high accuracy in the formal specification work, it has demerits of difficulty in representing data flows by including time concepts in the works such as modeling and simulation. Therefore, we considered that the method of using Statechart based on the state transition possible to complement it compositely was effective. Accordingly, we presented the method like Figure 3 which applies both Z and Statemate together, which is the most widely known tool as the Statechart tool supporting up to the formal specification and verification.
First of all, it shall pass through the formal specification stage showing the system requirements and development specifications. In the formal specification stage, the accuracy can be enhanced since the specification of written specifications composed of natural languages is prepared by formal specification languages which use graphic expression such as mathematical logic symbol formula or formal diagram with definite meaning, etc. Z proposed for application in this thesis makes very high integrity accomplished in case of preparing the specification for requirements such as system definition since it uses mathematical logic and sets [8][10]. Z which has merits of enabling expression in definite meaning usually checks the data type of system to be defined, and enables systems to be controlled by expressing such data flows effectively.

The schema in Figure 4 is the thing to represent characteristics of Z language specification very well. The schema makes Z specification distinguished from various other expressions, and it has the architecture introducing variables into the system to be specified and describing the relationship among variables. As it can be identified from the first schema box in Figure 4, the architecture is composed of; firstly the schema name, the schema signature composing of the system status at the top part of center line, and the schema predicate representing restrictions on the values of those variables at the bottom part of line. The bottom part of Figure 4 shows a simple example for preparing schema modeling the ‘letter inserting function of text editor’.

![Figure 4: Schema expression of Z language specification & example](image)

Generally, this motion is being executed at the editor if only pressing any one of the character or number located in the keyboard, but it can be applied only for the printable character, not for the control character. In Figure 4, the schema name was declared as ‘Insert’, and the schema signature within the ‘Insert’ shows that the ‘Insert’ is the motion changing the status of schema ‘Editor’, and declared the input variable to be inserted as ‘CHAR’ (ch?:CHAR). In Z, the input variable is always attached with a question mark (?) at the end of variable, and it is only for the purpose of expression to distinguish variables. The predicate informs us of how the status of editor is changed. The first line of predicate (ch? printing) shows the precondition, and it describes the condition which must become to be true before the next motion is occurred. The precondition informs us of the fact that the ‘character insertion’ motion can be occurred only for the input of printable character. The remaining part of predicate is the postcondition and it expresses the status after completion of motion. That is, ‘left’ = left ch?’ means that the input of new character is being added to the left side of cursor, and ‘right’ = right’ means that there is no change in the right part of cursor before and after motion.

Like this, Z language can express definitions and checking on the data type of system to be specified and its data flow effectively and definitely, but it has demerits requiring accurate knowledge and experience about the set theory and logic. Besides, since data flows include the concept of time, the existence of limitation in the expression is the problem in works such as concrete system modeling and simulation in the development specification using Z.

Since Statechart is expressed schematically through state-based visual specification language, and therefore, since it is easy to understand and it has merits to express behavioral aspects of system, it is being applied a lot to the design and implementation of embedded system [11]. By utilizing Statemate which supports both formal specification and formal verification based on the Statechart that is one of the formal specification languages, this thesis displays the architecture of targeted
system visually by using the Activity-chart supported by the tool, and it makes concrete behavior of system easy to understand through simulation of graphic model. Since Activity-chart has the function to express system status, it can check the operation between interfaces among data flow and module connected with various layers.

Figure 5: Specification instance of a flashlight operation using Activity-chart & statechart in Satemate

Statechart expresses all the design of system schematically, expressing the input and output of system as input and output of the signal and data, and the behavior of system as the transition of phase diagram. Figure 5 is the thing representing an example which specifies the motion of flashlight by using the Activity-chart and statechart; firstly at the top figure, it shows the interface architecture between the whole flashlight motion structure and exterior (user: ‘USER’) by using the Activity-chart on the flashlight. As you may check at the Activity-chart, the external ‘USER’ controls ‘@LIGHTER_CON’ module through ‘ON/OFF’ control flow indicated in dotted line. This ‘@LIGHTER_CON’ module includes the statechart diagram at its bottom as shown in the figure.

Figure 6: Application of formal specification
Like this, it indicates the architecture of system and input/output interface by using the Activity-chart, and expresses the function and motion status of system through statechart which is the sub-module of Activity-chart. Figure 6 is the figure which explains the formal specification method proposed in this thesis, and it verifies its relevancy, etc. by specifying definitions on data type, etc. by Z, and then it explains the procedures of specifying the architecture of system and interface by Activity-chart, and the function of system and system behavior such as data flow by statechart. Especially, by using the Statemate tool which supports both Activity-chart and statechart, it enables us to check visibly how the graphic-based specification modeled as Figure 6 operates system through simulation.

Since system requirement shall verify through simulation at the Statemate, and for the formal verification, it shall identify that there are no states like those in Figure 7 (Deadlock, Livelock, Reachability, Liveness) existed within the modeling, the items for formal verification module are included in the Statemate for this purpose. In comparison with other formal specification languages, it is difficult to find the tool supporting both formal specification and formal verification like Statemate. That is, the biggest merit of Statemate is to enable whole things up to preparation of specification, design and verification of complex system by consolidating the visual modeling, creation of simulation code, creation of documents, test plan, etc. into one.

3.2 Results of formal specification in accordance with proposed methods

This thesis proposed procedures and means of formal method to be applied for the design of vital train control systems in consideration of various factors. To verify the feasibility of proposed methods, we carried out the study on applicability to make requirements specified in formal specification form for the train route control algorithm which is a representative vital train control system. The train route control system is the vital control system by setting and protecting routes necessary for the train entering into the station yard so that the entering train can not collide with other train or derail from the branch section.

Figure 7: Formal verification items

Figure 8: System architecture using Activity-chart of Statemate
Figure 8 is the very result of preparation for the architecture of targeted train control system to meet the basic requirements of this train route control system. Figure 8 was prepared by using the Activity-chart of Statemate, and as shown in the result of preparation, it can be identified that it displayed complicated systems based on the equipment or software as graphic models effectively. Furthermore, it is possible to decompose them into smaller units which can design the targeted system by using the Statemate.

Based on the Activity-chart of Figure 8, Z made data form used by each status verified, which is because of the fact that Z is suitable for describing the input/output interface and input/output data type of the system. By using this Z, this thesis carried out the formal specification work with the object of point machine in train control systems as follows. In the point machine, it processes control on the point machine on the basis of status table of point machine. The status table of point machine shall store the information on the station name where any point machine was installed, ID of corresponding point machine, control command, control status, display status and the connected block, etc. The following Figure 9 is the result of specification on the corresponding contents by applying Z.

```
[ABCD] = {TransID, SegmentID}
OutputType = YES, NO
TABLE = TransID x P SegmentID

System

check
name: AB; name = YES \land \text{output} = YES \lor \text{output} = NO

registering

System

\(t_{\text{ref}}: \text{TransID}
\)
\(\text{ID}_{\text{ref}}: \text{P SegmentID}
\)
\(t: \text{TABLE}
\)
\(t = (t_{\text{ref}}, \text{ID}_{\text{ref}})
\)
\(\text{tableList} = \text{tableList} \cup \{t\}
\)

checkConflict

System

\(t: \text{TABLE}
\)
\(\text{output} = \text{OutputType}
\)
\(\text{if} \text{checkConflict}(t) \text{then}
\)
\(\text{output} = \text{M3}
\)
```

Figure 9: Formal specification results using Z language

In the Statemate, we made system control and algorithm verified by using the data type verified through Z. This is because the statechart of Statemate is suitable for expressing the behavior about which reaction the system will make under the condition determined as state-based language. This thesis tried to prepare the behavior for the interlocking software which is in charge of the basic route control function of train control systems by using the statechart of Statemate. This thesis obtained statechart results such as Figure 10 which make respective control behavior easy to be recognized by decomposing it into the part searching the route through meeting the control flow of train route control system, the part creating the status table of route, and the part controlling point machine and setting the route.

This thesis could identity the definite input/output data flow and functional motion of route control system through simulation by Statemate on the basis of definitions on data type by this Z and the graphic-based formal specification by using the Activity-chart and statechart. The requirements made of natural languages and prepared at the early stage of research through this formal specification process were re-prepared more concretely and definitely, and it is anticipated that the requirements
passed through formal specification will bring the shortening of time necessary for the detailed design and manufacturing of system and enhancement in safety.

Figure 10: Statechart result of interlocking S/W

Conclusion

When developing the vital software to be used for important systems, the safety of control systems developed will be increased effectively, if any formal specification and formal verification corresponding to the formal method are used. However, the results of research are still in the unsatisfactory situation for formal methods to be utilized in actual industrial fields, and especially in Korea, it is essential to have concrete procedures on formal methods for train control systems since there is no actual case applied in the railroad field. Like most of the overseas research cases for formal methods, if we utilize only mathematical logic formulas such as B method or VDM for the formal specification and formal verification of train control system, it will be stopped at the laboratory level of research instead of being applied to the actual railroad system like today since high-level of expertise is required due to the complexity of these formal specification languages. However, the application approach of formal methods applied Z and graphic-based Statechart proposed in this thesis complicatedly may be the practical alternative possible to overcome these problems. Especially, in the method proposed through the applicability of formal specification through this thesis, it is anticipated to enhance the actual applicability since the necessity of high-level of expertise on the formal methods will be reduced due to the application of formal specification languages in the form of graphics to most of functions and data flows. Accordingly, in case of applying the methods proposed in this thesis, it is anticipated that the securement of higher safety in the vital train control system will be possible since we may make much efforts in analyzing train control logic itself rather than professional knowledge on the formal specification languages conducted by existing studies. In addition, it is anticipated to meet the requirements set forth under international standards in relation to the railroad RAMS (Reliability, Availability, Maintainability and Safety) together with it.

References