Promotion of Local to Global Operation in Train Control System

Sher Afzal Khan
National University of Computer and Emerging Sciences
Islamabad
Pakistan
sher.afzal@nu.edu.pk

Nazir A. Zafar
Department of Computer and information Sciences
Pakistan Institute of Engineering and Applied Sciences
Islamabad
Pakistan
nazafar@pieas.edu.pk

Journal of Digital Information Management

ABSTRACT: *Railway interlocking system is a safety critical system. Its failure can cause the loss of human life, severe injuries and loss of money. Therefore the complication of this type of system requires advanced methodologies, which provide complete security and quality of a system. One way of achieving this goal is by using formal methods, which are mathematically based languages, techniques and tools used for specifying and verifying such systems. This paper provides the control of trains in a sector of moving block interlocking system using the approach of promotion. The promotion is the approach used to link the local state with a global state in Z specifications. The control comprises three components, i.e. sector, trains and security of a train in a sector.*

Categories and Subject Descriptors
D.3.2 [Language Classifications]; *Specialized application languages* **F.4.3** [**Formal Languages**]

General Terms
Railway interlocking systems, Formal mathematical languages, Z Specifications

Keywords: Z Notation, Predicate logic, Schema calculus, Formal specification

Received 17 December 2006; Revised and Accepted 12 February 2007

1. Introduction

Today software or hardware, are becoming pervasive in every field of human life, for example banking and trading sectors, aircraft control to mission critical satellite launchers. The quality and behavior of these systems directly affect the quality of human life. Many of these applications are quite complex and challenging in making the development of high quality software or hardware systems. Some of these applications like computer based control systems used in nuclear reactors, space avionics, process-control and robotics are safety-critical in the sense that failure could lead to financial or human catastrophes [13]. One way of achieving this goal is by using formal methods. Formal methods are increasingly being used in the development of software and hardware systems. Formal specification is a valuable tool in this process as it can highlight errors early at the design level in software life cycle when their correction is relatively easy and inexpensive. Formal methods are more accepted in both academia and industry because of their ability to improve the quality of both software and hardware system [2]. It is also quite clear that railway system is a safety critical system.

Now initially the task of railway interlocking system is to guarantee safety, i.e., preventing trains from collisions and derailing and on the other hand allow the train for normal movement. There exists a lot of work related to modeling of railway interlocking system. A list [1] of about 300 publications, addressing various issues of interlocking system, proves the importance of the system. The work [23] of A. Simpson is the specification of moving block interlocking system and it does specify some good schemas in Z but it does not present the safety of the system and the security functions of control system. The work in [27] of Zafar is the specification of moving block interlocking system using VDM-SL. Haxthausen et al. [14] describes elegant formal specification of a distributed railway control using RAISE. J.Hoenicke [15] has integrated CSP, OZ [5] and DC [28] in modeling component railway crossing of interlocking system. Morley. [21,22] develops a method, based on theorem proving higher order logic [12], to establish correctness of the signaling rules embedded in the geographic database of the solid state interlocking [4] developed by British Rail. Eriksson et al. [8,9,10,11] has formalized railway-interlocking requirement, which is of quite interest because of requirements analysis using formal approaches. In this paper we investigate the requirements for the safety control of train in moving block interlocking system. We will use the approach of promotion to link the train system with the corresponding control system associated in every sector using Z notations. In moving block system the open block (where open block is the front distance linked with a train for safe stop) is associated with every train in the railway network depending on the speed of train, as shown in Fig. 1.

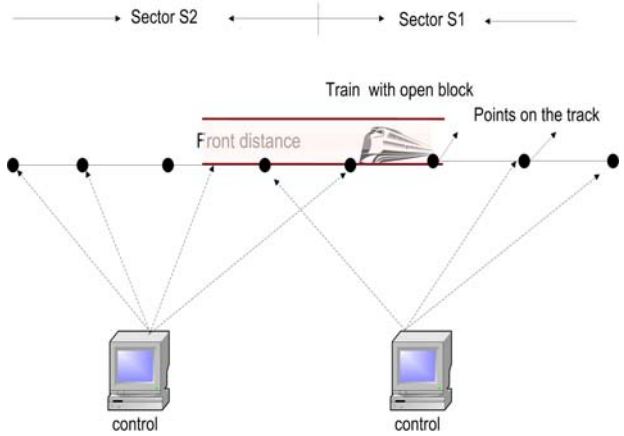

Figure 1. Moving Block Railway Interlocking System

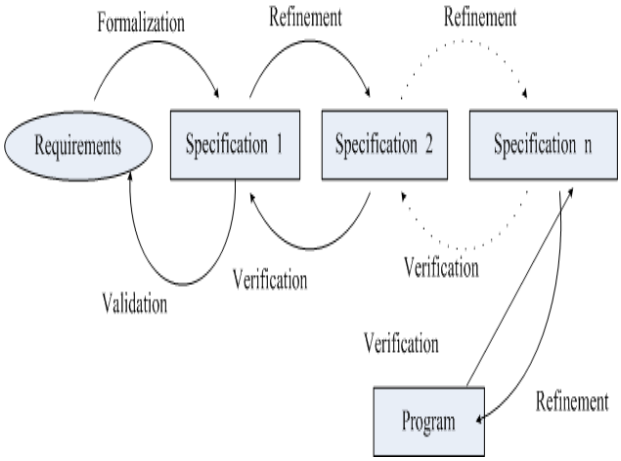

Figure 2. The Process of Software Development using Formal Methods

2. Formal Methods

Formal methods are based on mathematical techniques and notations, for describing and analyzing properties of software systems [3,6,7,19,25]. These mathematical techniques are typically based on discrete mathematics such as predicate logic, set theory, relations, functions, and graph theory. The precise mathematical notation eliminates the ambiguity, inconsistencies and incompleteness from any software and hardware system. The process [20] of developing systems using formal methods is shown in Fig.2. Based on the model of software life cycle; the "Requirements" are the result of requirements analysis and are normally described in informal language. 'Specification1' corresponds to the stage of transformation of requirements into formal methods so that it fulfills the requirements, and the process from 'Specification 2' to 'Specification n' corresponds to the stage of design. The stage from 'Specification n' to 'Program' corresponds to the stage of implementation or coding. Validation and verification are the two basic principles that arise in system development. Validation address whether the produced system fulfills the requirements, and verification check whether the software meets the requirements established in the previous phase. The aim of this approach is to demonstrate the process of development of a system from requirement to coding. This approach identifies errors and oversights early in the design life cycle, which are then easy to remove, with consequent high quality and cost saving. This improves the informal form of the system, but doesn't replace them. The Z notation [16,17,18,26], is model-oriented approach, and is based on discrete mathematics such as predicate logic, set theory, functions and relations. It is also used for specifying the behavior of abstract data type and sequential programs. Z specification divides the specification of complex system in different states called schemas. Schema is an important object of Z specification, which will be use in the specification of this paper. Schema consists of three parts; the first one is the schema name, which is in the top line, the second part between the first and second line is the schema signature, which is the set of names and types of entities introduced in a schema. The third part under the middle line is called the schema predicate, which is used for the set of properties and shows the relationships between the entities and the variables defined in the schema signature. These schemas can be combined to produce the overall description of the system. Z specification cannot typically be executed by computers, but the standard tools are available which are used for checking syntax and proof of the specification, lead to quality and correctness and this allows mistakes to be detected and corrected sooner in the design life cycle.

3. Moving Block Interlocking System

In this paper we will discuss the fundamental concept of moving block system. In this system the railway track is divided into sectors and then every sector is further divided into small segments where the segment is the distance between two consecutive points. The sector may have the network components (switches, crossing or level crossings). A fixed number of trains can move in a sector. Each sector is controlled by a computerized control system. Every train has onboard-computerized system which is used for storing its position, speed, sector identification and also for sending and receiving the signals from control system. The sector control system is required to inform continuously all the train operating in the sector for the safe travel. This information can be obtained from the state of the train, from the occupied sectors and from the occupied segments in network topology. For example a sector containing network components, like crossing and switch and three trains T1, T2 and T3. In Fig.3. Train T3 is moving along the right branch of switch and it has a clear status. While the open block of train T2 occupies the crossing, which changes the status of crossing including the surrounding segments connected to the four sides of the crossing into the red zone. The control system will inform train T1 about the status of train T2. As the open block of train T1 intersects the crossing segment the control section will inform the train T1 about the red zone and command the train T1 to stop before reaching to the crossing. In the meantime if the status of the crossing is clear the control system will allow the train T1 to move. In this system an open block is associated with every train depending on the speed of the train. Moving block is the red zone for any other trains with in the sector for the safe operation of the system. When a train enters from one sector to another then its moving block will cover some segments of both the sectors. Then the train will be observed by both the control systems of corresponding sectors as shown in Fig. 1. It is also defined that the moving block interlocking system does not require any physical signals.

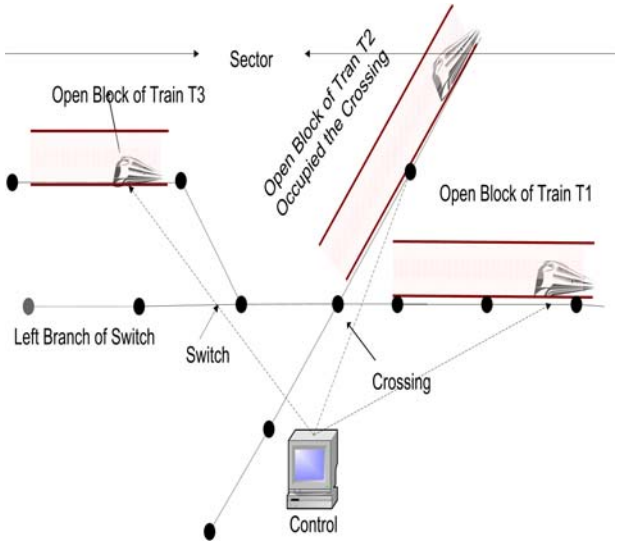

Figure 3. A Moving Block Network

4. Modeling of Control System

In this section we use the Z-notation to describe the control system of moving block interlocking system.

4.1. Fundamentals of the Systems

First we start with the identification of trains, we used *[ID, Trains]* where '*Trains*' is the set of all trains and '*ID'* is the set of identity numbers associated with the existing trains in the system. We also define *[Speed, openBlock]* where the set *Speed* is of the type natural numbers and shows the speed of the trains. The *openBlock* is the set of track segments, which depends on the speed of a train. The segment is the set of ordered pair of two consecutive points. Where points are the identification spot on the railway track and the set of all points on the railway track can be represented by the set *[Points].* Another given type is the set *[Sectors]* which is the set of segments defined as *sector* of type $\mathbb{P}$ (*Points x Points)*. The railway track in moving block interlocking system is divided into sectors. Every sector is controlled by a computer based control system. We use in this paper the approach of promotion, which is used to link the schema *TrainSystem* with schema *ControlRoom*. The schema *Promote* continuously updates the *ControlRoom* about any train in the corresponding sector and also about status of every segment in the sector.

5. Specification of the System

In this section we specify the control system for moving block interlocking system using the schema calculus of Z notation. The syntax and specification are checked and verified by using the toolkit Z/EVES 2.1. This includes the declaration of all the constants of the standard mathematical toolkit [24].

5.1. Train System

The specification of schema TrainSystem required the following sets:[ID, Trains, Speed, openBlock, Points, Sectors]. The specification is defined the injective functions trainIdentifier between the set ID and Trains. This shows the unique identification of the train in the system.

TrainSystem

$$trainIdentifier: ID \rightarrowtail Trains$$
$$trainSpeed: Trains \rightarrow Speed$$
$$blockSize: Speed \rightarrow openBlock$$
$$location: openBlock \rightarrow \mathbb{P} Points \times Points$$
$$distribution: \mathbb{P} Points \times Points \rightarrow \mathbb{P} Sectors$$
$$control: \mathbb{P} Sectors \rightarrow ControlRoom$$

The function *trainSpeed* is defined from the set *Trains* and the set *Speed* which gives the corresponding speed of any train moving in the sector under the control of corresponding control system. The function *blockSize* is the function between the set *Speed* and the set *openBlock.* It describes the size of open block associated with the train depending on the speed of the train. The next function *location* represents the division of the open block into the identified set of segments. This shows the identification of the location of the train in the corresponding sector. The *distribution* is a function from the set of segments and the set of sectors. This provides information about the sectors having the occupied segments. At the end *control* is the function from the set of sectors to the *ControlRoom*. This continuously informs the control room about the status of the related sector.

5.2 Control System

The state schema *ControlRoom* represents the control system. The variables, *sectorFree* and *sectorOccupied* are used for the state of sector in the specification. *sectorOccupiedby* is of type $\mathbb{P}$(*ID*x*Trains).* This provides the identification of sector, occupied by the corresponding train. The variable *segmentsOccupied* is of type $\mathbb{P}$(*Points*x*Points).* This shows the set of all segments occupied by a particular train in the corresponding sector. The *segmentOccupiedby* is declared as of type *ID*x*Trains* which provides the identification of train occupied the set of segments. The declaration *traininRedZone* represents the set of segments unsafe for the other trains.

RedZoneduetoTrain is the identification of train occupied the red zone. *DirectionofRedZone* represents the direction of red zone. *StopOppDirecTrain, StopSameDirecTrain, ContinueGreenZone* are the signals of the type *Reports* used for information to the existing trains in the corresponding sector to prevent them from collision and allow them for safe operations.

ControlRoom

$$sectorFree: Sectors$$
$$sectorOccupied: Sectors$$
$$sectorOccupiedby: \mathbb{P} ID \times Trains$$
$$segmentFree: Points \times Points$$
$$segmentOccupied: \mathbb{P} Points \times Points$$
$$segmentOccupiedby: ID \times Trains$$
$$traininRedZone: \mathbb{P} Points \times Points$$
$$RedZoneduetoTrain: ID \times Trains$$
$$directionofRedZone: Directions$$
$$StopOppDirecTrain: Reports$$
$$StopSameDirecTrain: Reports$$
$$ContinueGreenZone: Reports$$

5.3. Promotion

Promotion allows us to compose and factor the specifications. It has also been called framing, because it is evocative of placing a frame around part of a specification: only what is inside the frame may change; what is outside must remain unaffected as in [17] The *train?, id?* and *speed?* are the input for the system associated with a train entered into a related sector having the *openblock?* in certain direction. Where the question mark '?' with the variables represent the input to the system. The *directionofTrain* is of the type set *Directions* shows the direction of the incoming train in sector. Since the moving block interlocking system need continuously the information about the occupied sectors, occupied segments and the identification of a trains. This information can be obtained to the system by the input variable *sectorsOccupied?*, *segmentsOccupied?* and by the above defined variables *train?* and *id?.* All this functionality is operated continuously by the schema signature of schema *Promote*. The schemas predicate expressing the relationship between the local state *TrainSystem* and global state *ControRoom*. The change in *ControRoom* and a change in *TrainSystem* are linked by the identification of train with occupied segments contained in set of sectors. The schema predicate of *Promote* checks the identification of each sector and then linked the schema *TrainSystem* with schema *ControlRoom* by the relation.

θ *ControlRoom=control(sectorsOccupied?)*

Where *sectorOccupied?* is the set of sectors occupied by the open block of train is depending upon the train's speed, direction and segment occupied. All the above functionalities are described in the given operations of the schema *promote.*

Promote

$\Delta TrainSystem$
$\Delta ControlRoom$
$id?: ID$
$train?: Trains$
$speed?: Speed$
$openblock?: openBlock$
$directionofTrain?: Directions$
$trainLocation?: \mathbb{P}\ Sectors$
$segmentsOccupied?: \mathbb{P}\ Points \times Points$
$sectorsOccupied?: \mathbb{P}\ Sectors$

$(id?, train?) \in trainIdentifier$
$trainIdentifier' = trainIdentifier$
$(train?, speed?) \in trainSpeed$
$trainSpeed' = trainSpeed$
$(speed?, openblock?) \in blockSize$
$blockSize' = blockSize$
$(openblock?, segmentsOccupied?) \in location$
$location' = location$
$(segmentsOccupied?, sectorsOccupied?) \in distribution$
$distribution' = distribution$
$\theta ControlRoom = control\ sectorsOccupied?$
$control' = control \oplus \{(sectorsOccupied? \mapsto \theta ControlRoom')\}$

ated with any train in any direction other than the open block ob? in the corresponding sector. If there does not exist any open block in the controlled sector other than the *ob?.* Then control system will allow the train having open block *ob?* to move safely in this sector by giving the signal *ContinueGreenZone.*

ReceiveAndSendSignals

$\Delta Promote$
$ob?: openBlock$
$direction: openBlock \rightarrow Directions$
$segOccupied?: \mathbb{P}Points \times Points$
$segOccupiedby?: ID \times Trains$
$ContinueYouInGreenZone:Reports$
$StopOppDirecTrain: Reports$
$StopSameDirecTrain: Reports$
$segmentsStatus!: Reports$

$(ob?, segOccupied?) \in location$
$\wedge\ (\forall openblock: openBlock$
$\bullet ((openblock, segOccupied?) \notin \{ob?\} \lhd location$
$\Rightarrow segmentsStatus! = ContinueGreenZone))$
$\vee\ (\exists openblock: openBlock$
$\bullet\ ((openblock, segOccupied?) \in \{ob?\} \lhd location$
$\wedge\ (direction\ ob? \neq direction\ openblock$
$\Rightarrow segmentsStatus! = StopOppDirecTrain)$
$\vee\ (direction\ ob? = direction\ openblock$
$\Rightarrow segmentsStatus! = StopSameDirecTrain)))$

5.4 Receive and Send Signals by Control System

The schema *ReceiveAndSendSignals* need as input the open block *ob?* associated with a particular train occupying the set of segments *segOccupied?* arrived in the corresponding sector in certain direction giving by the function *direction* of the open block *ob?*. The predicate part of this specification plays an important role by providing security to all the trains moving in the sector controlled by the corresponding computer based control system. The specification involved in the predicate part check all the other open blocks associated with any train in any direction other than the open block

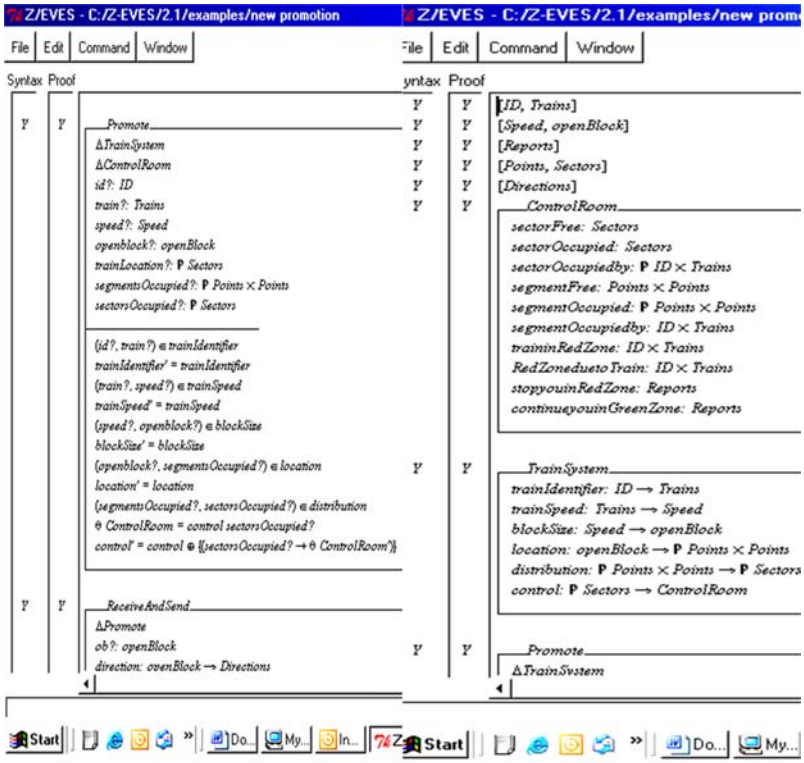

Figure 4. Syntax and Proof of the Specification

ob? in the corresponding sector. If there exists any other open block in the opposite direction of ob?, the control system will inform to stop the train by giving the following information to both the trains that is *StopOppDirecTrain*. This is the process of continuous signaling and both the trains will receive the above signal as both the open blocks associated with different trains intersect with each other. And hence there will be no problem for the train to stop safely. Similarly when their exists open block of any train in the same direction as of the open block *ob?*, the control system will inform both the trains with the signal *StopSameDirecTrain.* If we conjoin the *ReceiveAndSendSignals* and Promotes then we obtain a schema that describes operation upon the *ControlRoom.*

$$\exists \Delta\ ControlRoom \bullet ReceiveAndSendSignals \wedge Promote$$

5.5 Specification Analysis

Z is one of the formal methods, which is widely used in system development. Our experiences of applying this formal approach in modeling the system are the following. (1) This specification supported us to take out ambiguities and inconsistencies from the system by using the approach of promotion. (2) We were able to do systematic testing of syntax and proof of the specification, using Z-EVES 2.1 which is mathematical toolkit developed by ORA Canada used for checking syntax and proof of the specification. The verified snapshots of our specification are given in Fig.4, which provide the accuracy of syntax and proof of the specification given in the paper.

6. Conclusion

In this specification, computer based control system was introduced using the approach of promotion, for preventing collision and allowing normal trains movement. Two benefits were achieved. First, it was easy to develop a formal safety analysis of the system by using this type approach. Second and most important, benefit of using this approach is that complexity of the system is reduced. Development from abstraction to refinement made it easy to propose a simple and understandable formal model for moving block control system. The objective of this paper was to give the formal model of train system and its control in moving block railway interlocking system. Secondly to apply formal approaches in modeling and safety analysis of train system and its control, by using the approach of promotion. This links the train system with control system. Formal analysis of safety properties preventing collisions is given in the specification of this paper. Since the model of control of moving block interlocking system is not for a particular system, we believe that our model is useful for further research, that is to connect switches, and crossing in a sector of a network topology. Moreover the connection of two switches or the crossing in a sector will enhance this research.

References

[1] BjOrner, D (1998). The FME Rail Bibliography, Department of Information Technology, Technical University of Denmark.

[2] Bouali, A., Gnesi, S., Larosa, S (1994): The Integration Project for the JACK Environment, *Bulletin of the EATCS*, 207-223.

[3] Heitmayer, Constance (2005). Developing safety-critical systems: the role of formal methods and tools, *In*: Proceedings of the 10th Australian workshop on Safety critical systems and software, p.95-99, August 25, 2005, Sydney, Australia.

[4] Cribben, A. H (1987). Solid State Interlocking (SSI): An Integrated Electronic Signaling System for mainline Railways, *Proc. Of IEEE*, 134 (3) 148-158.

[5] Duke, R., King, P., Rose, R., Smith, G (1991). The Object-Z specification Language, Technical Report 91-1, Software Verification Centre, University of Queensland.

[6] Strunk, Elisabeth., Yin,Xiang., Knight, John (2005). Echo: A practical Approach to Formal Verification, *In*: International Workshop on Formal Methods for Industrial Critical Systems (FMICS), Lisbon, Portugal. p. 44-53..

[7] Edmund, M. Clarke., Winggie, Jeennette M (1996). Formal Methods: State of the Art and Future Direction, Carnegie Mellon University.

[8] Eriksson, L.H (1997).Formalising Railway Interlocking Requirements, Technical Report 3, Swedish National Rail Administration.

[9] Eriksson,L.H (1997).Formal Verification of Railway Interlocking, Technical Report 1997:4, Swedish National Rail Administration, (1997).

[10] Eriksson, L.H (1999). Some Technical Aspects of an Interlocking Specification Language, FMERail Workshop 4.

[11] Eriksson, L.H.,Fahlen, M.:An Interlocking Specification Language, Proc. of Int'l Conference on the Institution of Railway Signaling Engineers,(1999).

[12] Gordon, M.J.C., Melham, T.F (1993). Introduction of Higher Order Logic, Cambridge University Press, (1993).

[13] Smith, Green (2004). A Formal Framework for Modeling and Analyzing Mobile System. *In:* Australasian Computer Science Conference (ACSC 2004), Australian Computer Society.

[14] Haxthausen, A.E., Pleska, J (1994).Formal Development and Verification of a distributed Railway Control System, Proc. of the Ist FME Rail Seminar,(1994).

[15] Hoenicke, J (2001). specification of Radio Based Crossing with the Combination of CSP, OZ and DC, FBT.

[16] FAD VDM-SL Tools Group. "VDMTool" Austria, Vienna (1996).

[17] Woodcock, Jim., Davies, Jim (1996). Using Z: Specification, Refinement, and Proof. Prentice Hall International.

[18] Jonathan P., Hinchey, Bowen & Michael G. (1997). Ten Commandments of Formal Methods, *PC World Russia*, 56–63.

[19] Khan, S.A., Zafar, N.A (2006). Modeling of Information System using Z-Notation, *In*: 7th International Pure Mathematics Conference.

[20] Liu, S., Adams, R (1995). Limitations of Formal Methods and an Approach to Implement, Technical Report ,Hiroshima City University.

[21] Morley, M.J (1993). Safety in Railway Signaling Data: A Behavioral Analysis, *In*: Proc. of 6th Annual Workshop on Higher Order Logic Theorem Proving and Its Application, Vol.780 of LNCS, Springer-Verlag.

[22] Morley, M.J (1996). Safety Assurance in Interlocking Design, PhD Thesis, University of Edinburgh, (1996).

[23] Simpson, A (1997). Towards the Mechanical Verification of Moving Block Signaling Systems, Technical Report CMS-TR-99-06,School of Computing and Mathematical Sciences, Oxford Brookes University.

[24] Spivey, J.M (1992). The Z Notation: A reference Manual 2nd Ed. Programming Research Group University of Oxford.

[25] Wing, J. M (1990). A Specifier' Introduction to Formal Methods, *IEEE Computer* 23 (9) 8-24.

[26] Woodcock, J, Davies, J (1996). Using Z, Prentice Hall, 1996.

[27] Zafar, N.A (2004). Modeling of Moving Block Railway Interlocking System using Formal Methods, Ph. D. Thesis, Kyushu University, Japan.

Dr.Sher Afzal Khan is an assistant professor and Head of the Department of Sciences at the National University of Computer and Emerging Sciences (FAST), teaching Discrete Structures, Algorithm analysis, Numerical Computing and Formal Methods. His recent research has centered on the development of formal intelligent process for the safety of a railway interlocking system, using the approach of a formal Z-specification. His future research would cover the integration of formal Z-specification with fuzzy theory. This approach would broaden the use of formal method for the inelegant systems.

[28] Zhou, C., Hoare, C.A.R., Anders, P.R (1991). A Calculus of Durations, Information Processing Letters,40 (5) 269-276.