

Multi-Agent Communication Based Train Control System for Indian Railways: The Structural Design

Anshul Verma*, K. K. Pattanaik

Atal Bihari Vajpayee - Indian Institute of Information Technology and Management, Gwalior, India.
A-117, ABV-IIITM, Gwalior, M.P., India.

* Corresponding author. Tel.: +91-9826074618; email: anshulverma87@gmail.com

Manuscript submitted August 30, 2014; accepted March 12, 2015.

doi: 10.17706/jsw.10.3.250-259

Abstract: Indian Railways (IR) work on static block signaling (SBS) system to control mixed train traffic. Current state of operation can be improved by using communication based train control (CBTC) system with moving block signaling (MBS). The mixed traffic conditions in IR restrict the direct use of CBTC systems used elsewhere and thus first necessitated study of existing CBTC systems followed by development of CBTC system for IR. Operations research and constraint programming technologies do not cope well in complex, dynamic, and highly interactive systems like railway operations. Multi-agent systems are event based and have been used to express many complex real systems. Developments of multi-agent systems are based on logical structure design phase and implementation phase. Lack of sufficient standards and tools makes the design of complex agent based applications difficult thus most of the work do not describe the design phase in detail. This paper uses methodology for engineering systems of software agents (MESSAGE) for the analysis and design of multi-agent based CBTC system for IR. The paper describes the detail design of sub-goal *moving authority given* for the block section while considering various run time scenarios that might interrupt normal running of trains. A good structural design of the system shall however help for the study of behavioral aspects of the system through simulation or deployment.

Key words: CBTC, MESSAGE, multi-agents, railway signaling, structural design.

1. Introduction

Static block signaling (SBS) system to control mixed train traffic in Indian Railways (IR) has several limitations related to existing line capacity utilization and safe train operations. These limitations can be improved through communication based train control (CBTC) system with moving block signaling (MBS). Most countries, that use CBTC system, have homogeneous train traffic which makes its implementation easier. CBTC system with MBS is a safety critical system because its incorrect functioning may generate serious consequences. Therefore, in-depth study of various organizational and operating processes in IR is necessary to evolve with a software framework that provides increased flexibility in terms of further refinement as and when necessary.

CBTC system is a distributed, complex, dynamic, and highly interactive system and agent based computing is one of the powerful technologies for the development of such systems [1]. Agents represent the most important new paradigm for software development since object-oriented design [2]. Agent-based computing system has several advantages over traditional computing system as its inherent distribution property allows for a natural decomposition of the system into multiple agents that interact with each other to achieve a desired global goal. This approach significantly enhances the design and analysis of problem domains under the following three conditions [3]: The problem domain is geographically distributed; the subsystems exist in a dynamic

environment; and the subsystems need to interact with each other more flexibly. The operating scenarios of railway transportation system overlaps with the above three conditions thus making the multi-agent technology most suitable [4]-[6].

The most appealing characteristics of agents in any dynamic, complex, and highly interactive scenario are their autonomy, mobility, collaboration, and reactivity. The collaborative property of multi-agent system enables agents communicate with each other to achieve a global goal. On the other hand, their ability to perceive about the environment where they operate and react timely to environmental changes is another important property [7].

In spite of several interesting properties, however, a very few deployable multi-agent systems have been developed. Development of multi-agent systems involves: *logical structure design* and *implementation*. One of the major factors delaying widespread adoption of multi-agent systems is due to the lack of standard software development techniques focusing on the structural analysis and design of any deployable system [8]. Existing object oriented analysis and design techniques do not adequately model agent paradigm's properties. In such situations agent oriented analysis and design techniques to provide developers with a higher degree of expressiveness and flexibility to represent agent based systems is required. Several methodologies such as societies in open and distributed agent spaces (SODA) [9], Tropos [10], multi-agent systems engineering (MaSE) [11], methodology for engineering systems of software agents (MESSAGE) [12], and a process for agent societies specification and implementation (PASSI) [13] etc. have been developed for the analysis and design of agent oriented systems. Out of them, MESSAGE provides a powerful set of views in comparison to other methodologies for graphically describing structural aspects of agent oriented systems. Furthermore, it provides a process and a number of heuristic rules for carrying out the analysis and design of multi-agent systems. Several agent implementation tools [14] such as java agent development framework [15], zeus agent building tool-kit [16], jack intelligent agents framework [17] offer a strong support for implementation of multi-agent systems. All these tools, however, lack full support for the analysis and design phase that form the central in any software development life cycle.

This paper presents MESSAGE based multi-agent CBTC framework for IR named Indian Railways Management System (IRMS) and in particular the design of sub-goal *moving authority given* for the block section is discussed in detail. The rest of the paper is organized as follow: Section 2 covers the related work and Section 3 presents background and overview of the proposed system. Structural design of the system is covered in Section 4 and analysis of the system is presented in Section 5. Finally Section 6 concludes the paper.

2. Related Work

Several approaches for automatic train control based on mathematical models, simulations, multi-agents, and soft computing techniques corresponding to SBS and MBS systems have been dealt in the literature. To the best of knowledge our work is the first to explore the issues specific to IR that might be important to consider before moving IR to CBTC on MBS based infrastructure. IR system operates under its General Rule book [18] and defines numerous functionalities and operational constraints. For the sake of simplicity in this section we limit our discussion to a few of them and assess them from the view point of a CBTC system. Generally speaking, the entire operational aspect can be divided into two major components: first, the *scenario* in which the trains run and the second, how the *functional component(s)* react to such scenario(s). In the following, we discuss related research in the direction of multi-agent based train control system.

In our earlier work [19], a multi-agent based CBTC framework with MBS using MESSAGE methodology was presented for IR. It was a work-in-progress paper and a simplified model of sub-goal *moving authority given* was implemented using CPN to evaluate the behavioral aspects.

Implementation of a completely autonomous multi-agents based train control system is presented in [20]. Authors describe several issues related to stability and controllability, provided relevant safety and operational requirements by using communication methods and various sensors. However, the multi-agent architecture

based on any standard agent based software modeling technique was missing. Further, there was little information about the structure's behavioral analysis. Absence of a agent oriented standard methodology for structural design does not bring much clarity about the software system.

MARCS - multi-agent railway control system is presented in [21] is a multi-layered system comprises three distinct agent types, i.e., supervisor, train and station. The learning layer creates rules by learning from the situations accumulated by the control layer. These rules were used to improve traffic control processes, minimize wait time and stop orders sent for each train. However, the detailed structural design using a standard agent oriented methodology was missing.

A hybrid software agent model for designing Chinese Train Control System (CTCS) simulation software using deliberative and reactive agents is developed in [22]. The reactive memory used in the model appropriates the agents' use in real-time. Authors concluded the efficacy of their hybrid software agent to distributed supervising and control systems. UML was used to describe the structure of CTCS in sufficient detail. UML does not support to fully describe agents' characteristics and thus use of agent oriented structural design methodology for describing agent based systems will be more beneficial.

Multi-agent system for the open railway access market [23] is developed to model the negotiation process between a train service provider (TSP) and an infrastructure provider (IP) for track access permissions. The multi-agent used TSP agents and IP agents to interact with each other to enable optimal track access according to the requirements given by TSP agent.

A multi-agent system given in [24] works on contract-net protocol and is capable of handling planning and plan execution monitoring in a real-time railway transportation scenario. Outcomes of their work provide further insight into the route optimization issue in shared track infrastructure.

To the best of our understanding none of the above works focused on the detailed structural analysis and design of multi-agent based system. The absence of detailed structural model of any system makes the understanding difficult for its analysis and further structural modifications to enhance the functionality if required. We felt it is important to follow a methodology that builds upon best practice methods in software engineering for the analysis and design of agent based systems.

3. Proposed IRMS

In this section we first discuss the background of the present signaling system used in IR and discuss the MBS based CBTC system used elsewhere. Further the major components of the proposed IRMS and their interaction is presented.

3.1. Background

IR covering a total rail length of 65,000 km is one of the largest railway networks in the world. The signaling system followed in IR is SBS and has six systems of working such as absolute block signaling (ABS) system, automatic block signaling (ABSi) system [25], following train system, pilot guard system, train-staff and ticket system, and one-train only system [18]. According to chapter VII of General Rule Book of IR [18] "ABS and ABSi" alone shall be used while other working may be used under special circumstances. IR operates several different categories of trains, widely varies in terms of their technology, speed, braking performance, number of stoppages etc. and termed as mixed traffic. The following paragraphs highlight various aspects of SBS and MBS system. Further, we mention on the need for MBS system for IR.

ABS system consists of *block section* - the entire section between two stations and *station section*. In long block sections ABS results in severe underutilization of track section for both slow and fast trains due to the reason of one train in one block section in any given time. ABSi improved track utilization by further sectionalizing the block section into multiple smaller sized block sections (as small as 1 km) each controlled by automatic block signals. Smaller block sections does improve the utilization but not optimally. Moreover, when enforced on mixed traffic, the considerably large safety distance required by fast trains are imposed also on slower trains. This makes marginal improvement in utilization of track capacity which can be further improved. On the other

hand smaller block section lengths is better for slow trains but not for fast trains due to safety aspects. Cab signaling [26] (not used in IR) is another enhancement to ABSi where signal aspects are transmitted to train driver's console through wireless communication. Cab signaling eventually leads to CBTC system [27].

In MBS [28] the blocks are decided in real time by computers as safe zones around each train. This requires the knowledge of the exact location, speed, direction of trains and continuous radio between the zone controller (ZC) and the train's on-board system. Track side active and passive markers, and train-borne tachometers along with underlying communication system are the principal enablers for the real time determination and communication of several train operation control information. The tests [28] indicated that CBTC system based on MBS (For example European Rail Traffic Management System (ERTMS) level-3 [29]) greatly improved service quality and transportation efficiency in comparison with SBS.

3.2. Overview of IRMS

The software agent based CBTC rail track infrastructure is divided into areas or regions, each area under the control of a ZC and each with its own radio trans-receiving system with reliable and continuous radio link. Fig. 1 shows the high level architecture of IRMS which describes the important functional components and their corresponding agents. It consists of four principal components: ZC, Station Controller (SC), Trackside Device Control System (TSDCS), and On board Device Control System (OBDCS). Working of these functional components is outlined here.

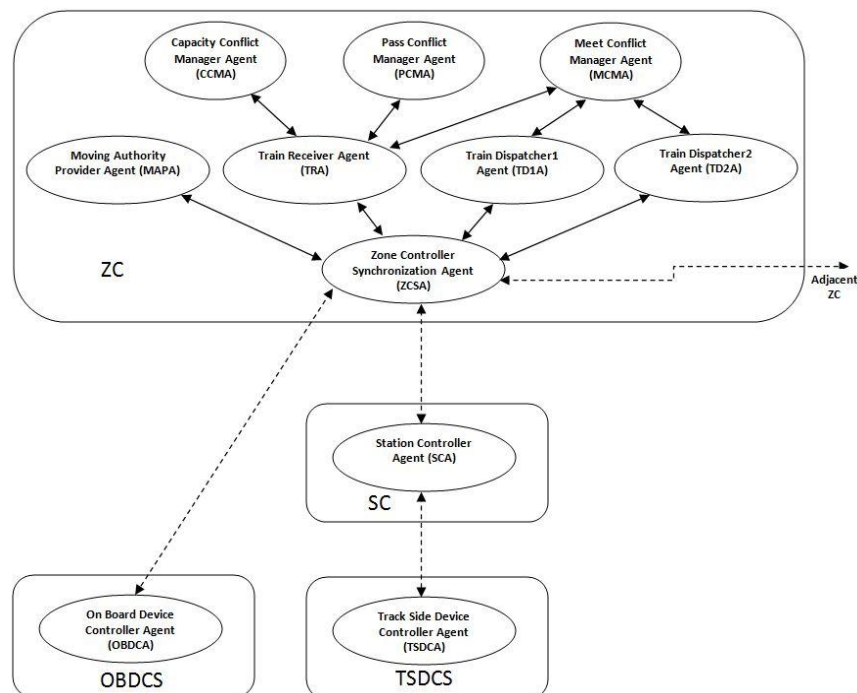


Fig. 1. Abstract level diagram of IRMS.

Each train has an OBDCS that includes vehicle onboard controller and several train borne equipments, such as GSM-R trans-receiver antenna, RFID reader, speedometer, accelerator, braking unit, driver display screen, and train integrity monitor etc. Train borne equipments collect the relevant information, such as train number, speed, location identifier, direction etc., and periodically send to their respective ZC. Location identifier is the location reference read by train borne RFID reader through track side RFID tags.

On the reception of this information, respective ZC computes safe moving authority on the basis of ahead track status and train characteristics, and communicates to respective train. Further, ZC gives the instructions to SC to create the route by interlocking for the arrival or departure of trains in station section. Train handover between two adjacent ZCs occurs when a train passes the overlapped ZC boundaries.

Each station section has a single SC responsible for interlocking (establishing or releasing a route) for the

arrival or departure of trains. The route is created by fixing all switches presented on the path in a required position. SC receives a route from ZC, i.e. sequence of switches and their positions, and gives instruction one by one to TSDCS to fix each switch in a required position. The SC on receipt of positive acknowledgement for all required switches from TSDCS, it gives route establishment acknowledgement to ZC. If SC receives negative acknowledgement for any required switch, it releases all switches and gives negative route establishment acknowledgement to ZC.

Switches falling under a particular station section are controlled by the respective TSDCS. It is responsible to fix each individual switch in a required position. It receives a switch identifier and its position information from SC. If switch is fixed in desired position it transmits positive acknowledgement to SC otherwise negative acknowledgement. TSDCS also continuously monitors health and status of all its switches, and reports to respective SC.

4. Structural Analysis and Design of IRMS

This section discusses MESSAGE methodology based on which the IRMS is designed. Further, the structural design of IRMS is presented.

4.1. Overview of Message

MESSAGE [12] is an agent oriented software engineering methodology which builds upon current software engineering best practices covering analysis and design of multi-agent systems. It extends the UML by adding new concepts such as agent, goal, and task etc. required to analyze and design agent oriented systems. A number of predefined guidelines and rules help the developer to analyze and design agent based systems in accurate way. It uses a rich set of notations and proper textual labels for easy and intuitive representation of several view diagrams of the agent oriented system. It provides a set of views each emphasize on a limited but consistent aspects of a system, however together they represent a complete view of the whole system. The views/diagrams that it describes:

- *Organisation diagram*: Organization is a group of agents working together for a common purpose. Organization diagram shows concrete entities (agents, organisations, roles, resources) of the system and relationships between them (aggregation, power, and acquaintance relationships).
- *Goal/Task diagram*: Goal is an intension of an agent to achieve some desired state. Goal diagram defines the goals of the composite system and their decomposition into sub-goals. Task is the basic unit of activities preformed by an agent to achieve a particular goal. Task workflow diagram captures how a number of tasks can be performed in a sequence to achieve a particular goal.
- *Agent/Role diagram*: Agent is an atomic autonomous entity capable to perform some useful function. It can play several roles each describing its external characteristics. Agent diagram focuses on an individual agent and its roles. It describes agent's characteristics such as what goals it is responsible for, what resources it controls, what tasks it knows, how to perform a task, etc. Roles can also be used as indirect references to agents.
- *Interaction diagram*: The interaction diagram is concerned with capturing the way in which agents (or roles) exchange information among each other (as well as with their environment).
- *Domain diagram*: Domain works as a repository of relevant information about the problem domain. The domain diagram shows the domain-specific classes, their structures and relations with others. MESSAGE uses UML class diagrams for this purpose.

The various concept symbols and relation symbols used in MESSAGE are presented in Fig. 2.

4.2. Structural Design

MESSAGE provides above described five different views to the system designer for the analysis and design of agent oriented systems. We used MESSAGE to design all these views to capture different aspects of the proposed IRMS. These views of IRMS are as follows:

4.2.1. Organization diagram

Fig. 3 describes the level 0 organization diagram of IRMS representing the structural relationships. It has four organizations OBDCS, TSDCS, SC, and ZC, each represent a set of agents working in unison for a common goal. Further, it shows the relation between system and its external resources. External resources include databases for maintaining train related information, train schedule, track, and track side devices, and deals with three types of physical entities such as train, track, and track side devices. The working and responsibility of each organization is same as described in section 3.2. Organizations OBDCS, SC, and ZC are associated with roles driver, station master, and zone control master respectively responsible for monitoring and supervisory control. Each organization can be further analyzed at level 1 for its detailed view.

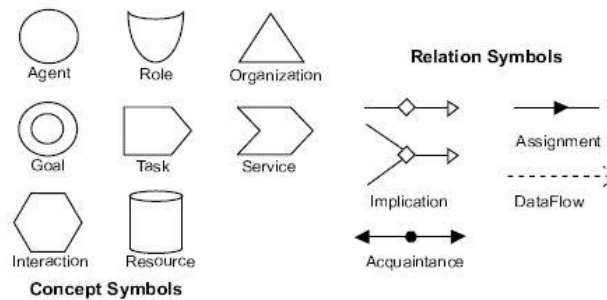


Fig. 2. Concept symbols and relation symbols in MESSAGE [11].

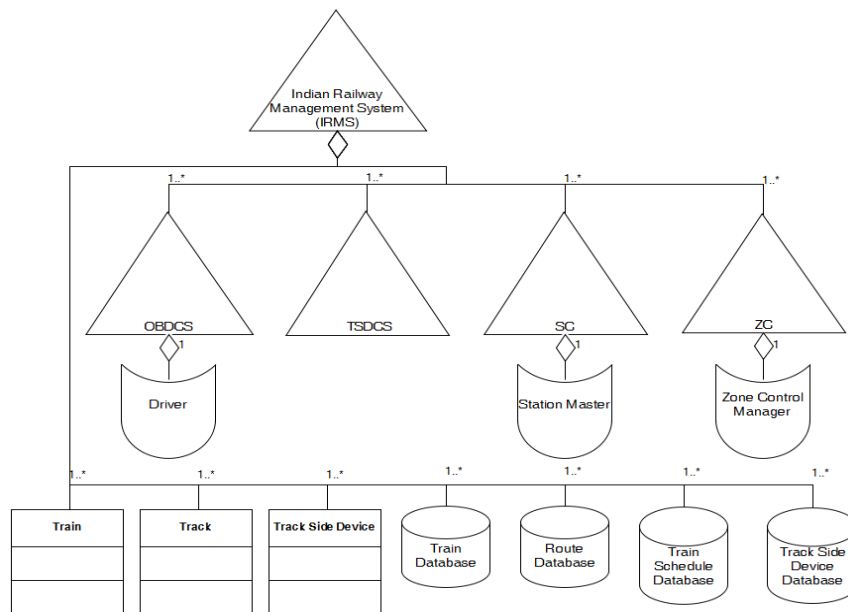


Fig. 3. Organization diagram of IRMS

4.2.2. Goal diagram

The main goal *Train Moved* safely is decomposed according to the goal decomposition diagram of MESSAGE. The goal view (see Fig. 4) represents various sub-goals to be attained by the system for safe movement of trains both in station section and block section. Further every station section has different scenarios like single track or double track. Certain scenarios cannot coexist for example a train at a given time cannot be both in station section and block section. Such scenario is represented by Or-decomposition. Whereas certain operations can coexist in a station section for example operations like *train received*, *train dispatched*, and *conflict detected and resolved*. Such scenarios are represented by And-decomposition. Similarly, the block section can have only sub-goal *moving authority given*. As the operational rules for block sections are different from station section, and the work focuses on designing of block section only, in the following we discuss design related to sub-goal *moving authority given* for block section. However, design related to the station section is a work under progress.

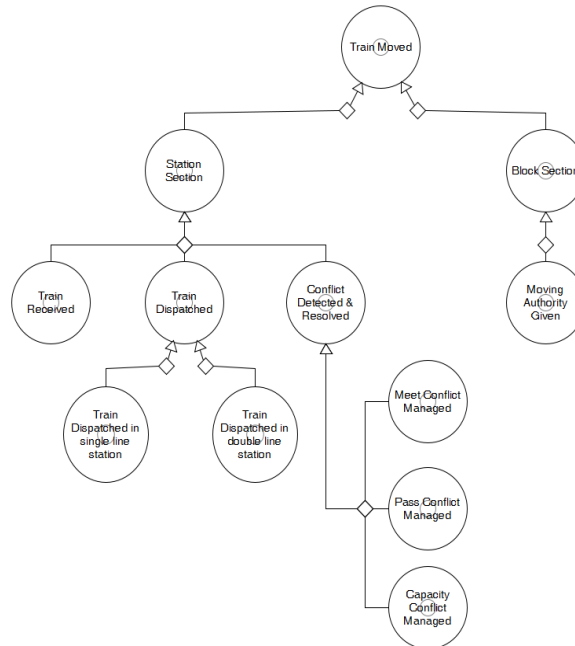


Fig. 4. Goal diagram of IRMS.

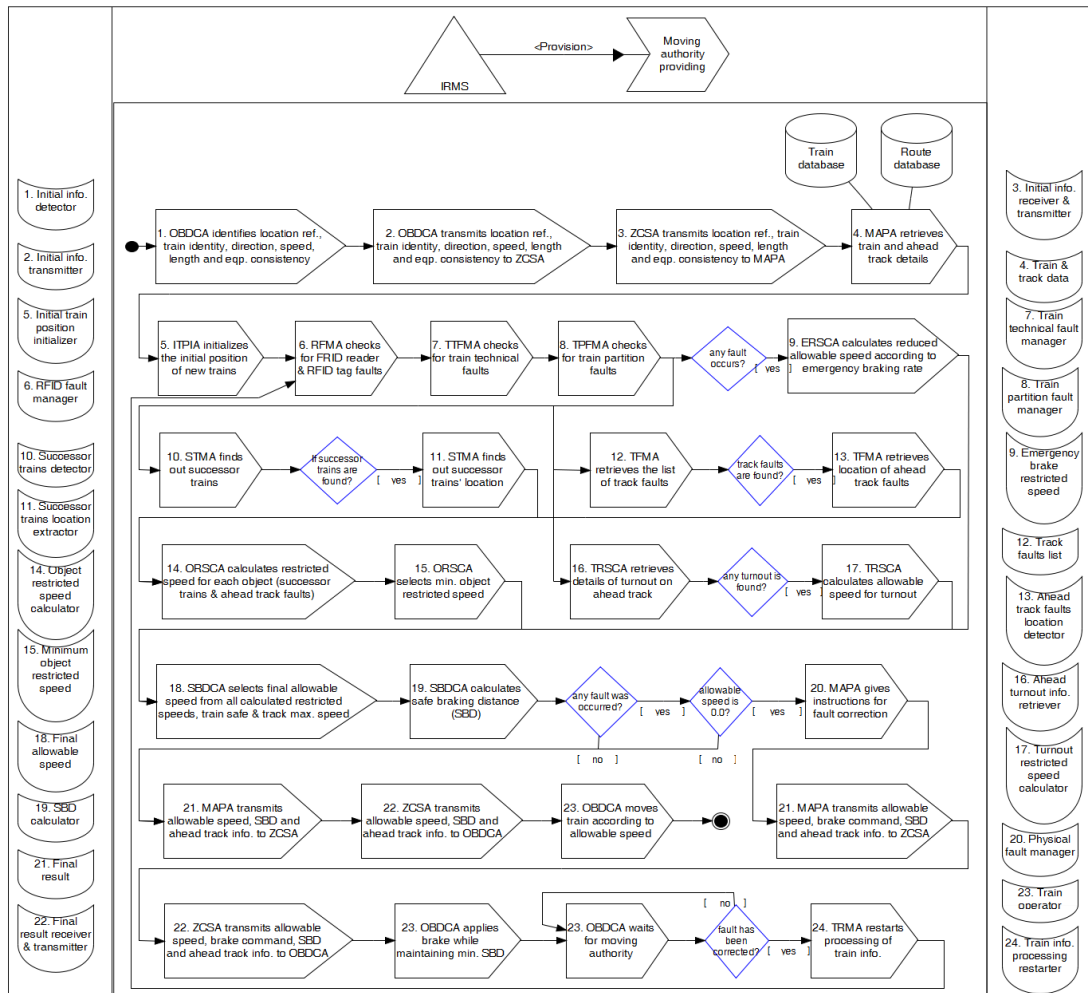


Fig. 5. Task workflow diagram of sub-goal moving authority given.

4.2.3. Sub-goal moving authority given

Here we furnish further details like workflow diagram, agent diagrams, and interaction diagrams of the

sub-goal *moving authority given*.

- a) **Task Workflow diagram:** Each sub-system in Fig. 1 contains one or more identified software agents to carry specific task(s). These agents follow a predefined workflow according to IRMS and operate in a collaborative manner to perform the overall goal of each sub-system. Fig. 5 shows a sufficiently detail diagram showing every task performed by respective agents during the course of *moving authority given* decision process. Agents can be of three types: reactive, proactive, and hybrid. In this work agents are of hybrid category. For the expansion of certain abbreviations used in Fig. 5 the reader may refer to Fig. 1. According to MESSAGE's task workflow designing rules, each task is connected with a role, responsible to perform the task, through a line labeled <<performance>>. However, due to space restrictions the lines connecting the role and task have not been shown here. To make the representation simple, each task is numbered corresponding to their role.
- b) **Agent diagram:** Fig. 6 explains MAPA and its relationship with other entities, i.e. roles, goals, services, resources, and other agents. The functionality of MAPA is divided into four roles. *Initial info. receiver* role receives train information from ZCSA, *Track & train data retriever* role retrieves specific track and train details from database, **MAPA's sub-agents communication manager** role helps exchange information between MAPA and its underlying sub-agents, and *Final result transmitter* role transmits allowable speed and safe braking distance determined by underlying sub-agents with ahead track information to ZCSA. The combined effort of all these roles and sub-agents helps MAPA to providing moving authority service and to achieve *moving authority given* goal. Each sub-agent is responsible for achieving its specific goal. The description of MAPA's sub-agents is given in Fig. 7.
 - **Initial trains position initializer agent (ITPIA):** It performs only one role i.e., *Initial train position initializer* shown in Fig. 7(a), that initializes the initial start position of all trains within the respective ZC to the corresponding MAPA.
 - **RFID fault manager agent (RFMA):** RFID fault may occur either due to train borne RFID reader or track side RFID tags malfunctioning. The functionality of RFMA is divided into three roles as shown in Fig. 7(b). *Train occupied RFID list generator* role generates the list of RFIDs occupied by the train representing the length of train, *Trains occupied RFID lists manager* role manages lists of RFIDs occupied by all trains running within the same ZC, and *RFID fault manager* role detects the RFID faults and instructs to apply emergency brake if more than one consecutive RFIDs are missed due to either train borne RFID reader or track side RFID tag malfunctioning. The combined effort of all these roles helps achieve RFID fault resolved goal.
 - **Train technical fault manager agent (TTFMA):** Train's consistency report defines the overall health status of its equipments. Fig. 7(c) presents structure of TTFMA. It contains a single role *Train technical fault manager* that checks consistency report of train's equipments sent by OBDCa to detect technical fault. If a fault is detected, it instructs to apply emergency brake.
 - **Train partition fault manager agent (TPFMA):** A train is partitioned due to brake in coupling and such situations referred as train partitioning. Fig. 7(d) describes the structure of TPFMA. The functionality of TPFMA is divided into three roles. *Train partition fault manager* role checks whether train has been parted or not, if yes it instructs to apply emergency brake and *Parted trains list manager* role appends the last RFID occupied by train's rear end when train was not parted, and **Successor parted trains location estimator** role extracts the locations of last RFIDs occupied by the rear ends of successor parted trains.
 - **Successor trains manager agent (STMA):** Safe operation of successor trains depends on the ahead train *location* information. Fig. 7(e) presents the structure of STMA. It performs the single role *Successor trains location extractor* that extracts the locations of last RFIDs occupied by the rear ends of ahead

trains and achieves Successor trains detected goal.

- **Track fault manager agent (TFMA):** Track faults refer to restricted speed of operation in a given region or not allowed to proceed at all until fault is rectified. The functionality of TFMA is divided into two *roles* as shown in Fig. 7(f). *Track faults list manager* role maintains the list of track faults detected by underlying track side equipments, and *Ahead track faults detector* role extracts the location and restricted speed information on ahead track.

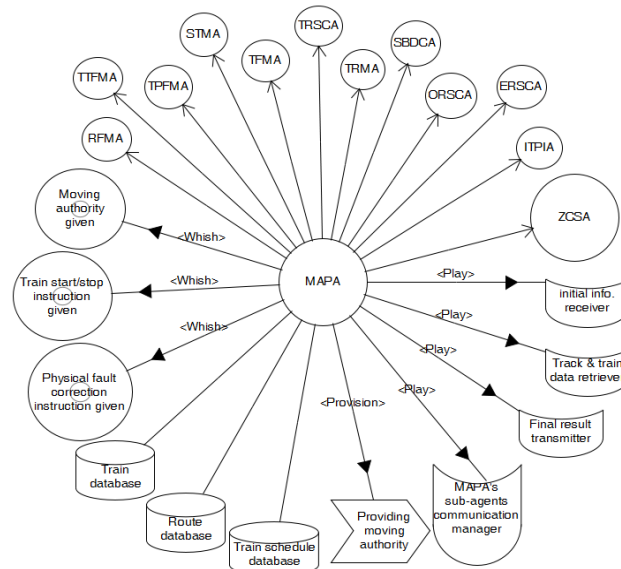
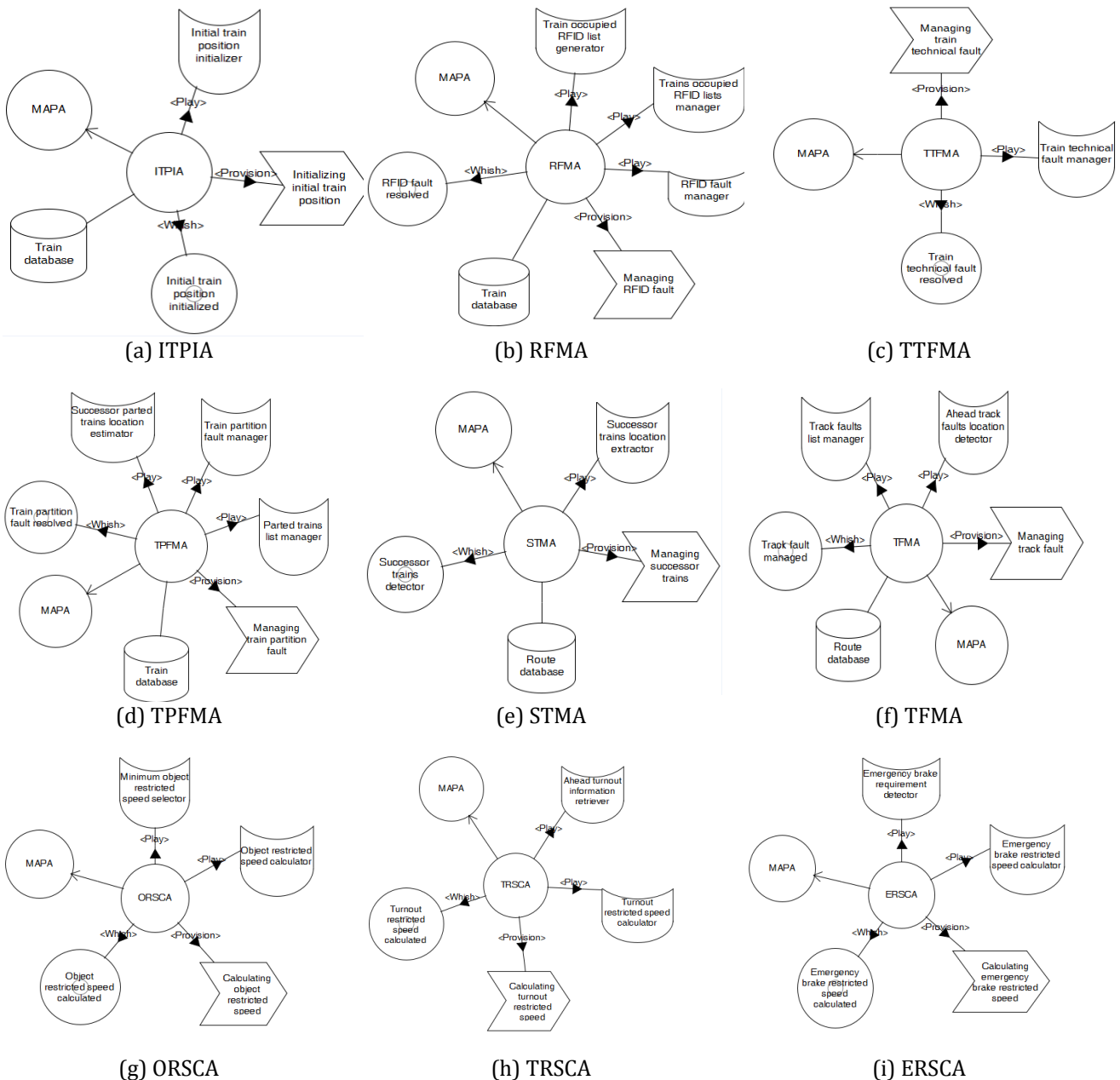


Fig. 6. Agent diagram of MAPA.

- **Object restricted speed calculator agent (ORSCA):** The safe operation of train depends on object(s) on the track ahead. Objects can represent train, parted train, a faulty **track**, or a track with speed restriction. The ORSCA performs two roles shown in Fig. 7(g). *Object restricted speed calculator* role takes the *locations* of the ahead trains, ahead parted trains and ahead track faults as input to determine the speeds of train in question. *Minimum object restricted speed selector* role selects the minimal speed of the calculated object restricted speeds.
- **Turnout restricted speed calculator agent (TRSCA):** Turnouts on track enables trains to be guided from one track to another. Such portions have speed restrictions and the running trains must follow the speed restrictions. Working of TRSCA is divided into two roles as shown in Fig. 7(h). *Ahead turnout info retriever* role **retrieves** the information of the turnout on ahead track and *Turnout restricted speed calculator* role determines the restricted speed of the train for that turnout on ahead track.
- **Emergency brake restricted speed calculator agent (ERSKA):** Due to variety of reasons train may receive emergency **braking** instruction from ZC. The functioning of ERSKA is divided into two roles as shown in Fig. 7(i). *Emergency brake requirement detector* role checks the status of various faults, if faults occur *Emergency brake restricted speed calculator* role determines the desired speed of the train according to emergency braking rate for that train.
- **Safe braking distance calculator agent (SBDCA):** Safe braking distance defines the distance at which the train stops according to its braking characteristics. SBDCA performs two roles as in Fig. 7(j). *Final allowable speed selector* role selects the smallest value of the speed among the turnout restricted speed, **emergency** braking reduced speed, object restricted speed etc., and *SBD calculator* role calculates the safe braking distance of the train according to the selected final allowable speed and train's braking characteristics.
- **Train restart manager agent (TRMA):** Train restarts after normalcy is restored. TRMA performs the two roles as in Fig. 7(k). *Physical fault manager* role is responsible for providing instructions to correct

physical faults related to train and infrastructure that require human intervention, and *Train information processing restarter* role checks the status of stopped trains, those were stopped due to some faults, if they have elapsed their assigned waiting time and respective faults have been corrected, the role *restarts the processing of train information* is used to provide moving authority.

- c) **Interaction diagram:** ZCSA decides what operations have to perform with the information received from OBDCA on the basis of train's current location and transmits this information to respective agents for further processing. Because trains are running within block section, ZCSA transmits the received information to MAPA for the moving authority determination. Fig. 8 represents the ZCSA-MAPA interaction and the reason for their interaction. Each **unique** interaction between two agents is represented by an individual interaction diagram. Whereas, the format of a message exchanged during the interaction is illustrated by domain diagram (same as UML class diagram). The interaction diagram of each interaction between MAPA and its sub-agents, and domain diagram of each message have not been presented here for the sake of simplicity. Fig. 9 illustrates a UML message sequence chart (MSC) to represent the interaction and message sequence between MAPA and its sub-agents.



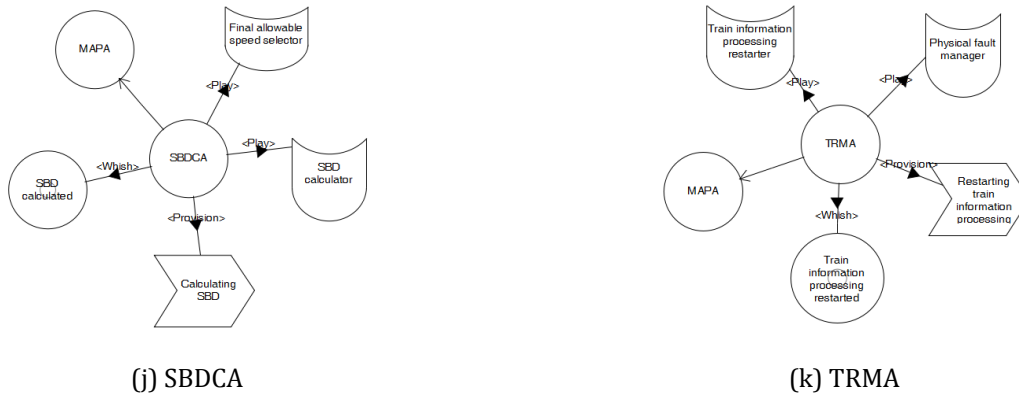


Fig. 7. Agent diagram of MAPA's sub-agents.

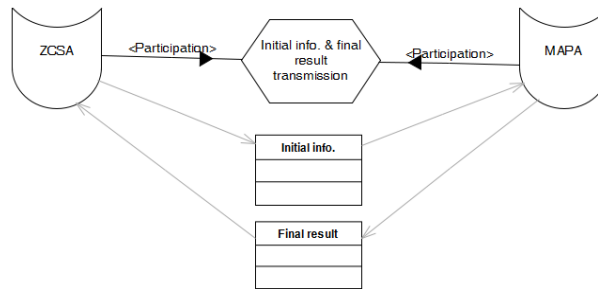


Fig. 8. Interaction diagram between ZCSA and MAPA.

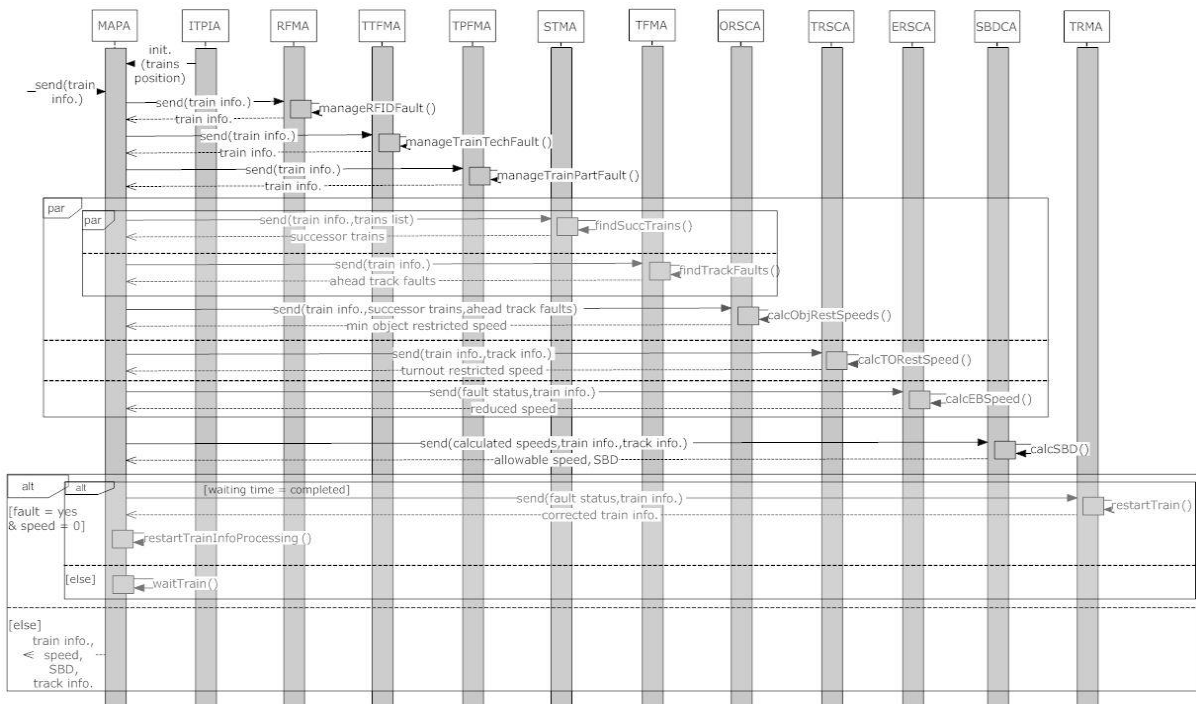


Fig. 9. UML-MSC of MAPA and its sub-agents.

5. Conclusion

The paper focused on the design aspects of multi-agent based IRMS and proposed the structural analysis and design using MESSAGE methodology due to its high degree of expressiveness and flexibility. In addition, the work attempted at providing the system design based on software best practices in sufficient detail which was missing or insufficient in most of the multi-agent system related research. The design approach, while serving as a case of IR, however, would provide better understanding of the relation among the various organizational

components and processes that in general is foundation for any complex, dynamic, and highly interactive multi-agent system design. In the following work we will be reporting about the formal modeling of the designed system to analyze the behavioral aspects.

Acknowledgment

The work is a part of project named "Multi Mobile Agent based Train Operation in Moving Block Setup" sponsored by Department of Information Technology (DIT), Ministry of Communications and Information Technology, Government of India, vide grant number 2(6)/2010-EC dated 21/03/2011.

References

- [1] Zambonelli, F., & Parunak, H. V. D. (2003). Signs of a revolution in computer science and software engineering, *Agents World, Lecture Notes in Computer Science*, 2577, 13–28.
- [2] Luck, M. (2004). Guest editorial: Challenges for agent-based computing. *Auton. Agents Multi-Agent Syst.*, 9(3), 199–201.
- [3] Adler, J. L., & Blue, V. J. (2002). A cooperative multi-agent transportation management and route guidance system. *Transp. Res. Part C: Emerging Technol.*, 10(5-6), 433–454.
- [4] Wang, F. Y. (2005). Agent-based control for networked traffic management systems. *IEEE Intell. Syst.*, 20(5), 92–96.
- [5] Wang, F. Y. (2008). Toward a revolution in transportation operations: AI for complex systems. *IEEE Intell. Syst.*, 23(6), 8–13.
- [6] Verma, A., & Pattanaik, K. K. (2014). Mobile agent based train control system for mitigating meet conflict at turnout. *Procedia Computer Science, Elsevier*, 32, 317–324.
- [7] Macal, C. M., & North, M. J. (2010). Tutorial on agent-based modelling and simulation. *Journal of Simulation*, 151–162.
- [8] Iglesias, C., Garrijo, M., & Gonzalez, J. (1998). A survey of agent-oriented methodologies. *Intelligent Agents: Agent Theories, Architectures and Languages, Lecture Notes in Computer Science*, 317–330.
- [9] Omicini, A. (2001). SODA: Societies and infrastructures in the analysis and design of agent-based systems. *Agent-Oriented Software Engineering, Lecture Notes in Computer Science*, Springer New York, 1957, 185–193.
- [10] Giunchiglia, F., Mylopoulos, J. & Perini, A. (2002). The TROPOS software development methodology: Processes, models and diagrams. *Proceedings of the International Joint Conference on Autonomous Agents and Multi Agent Systems AAMAS02* (pp. 96–100). ACM Press New York.
- [11] Wood, M. F., & DeLoach, S. A. A. (2000). An overview of the multi-agent systems engineering methodology. *Proceedings of the International Workshop on Agent-Oriented Software Engineering* (pp. 127–141). LNCS, New York: Springer, 1957,
- [12] G. Caire *et al.*, (2001). Agent oriented analysis using MESSAGE/UML. *Agent-Oriented Software Engineering II*, LNCS, New York: Springer, 119–135.
- [13] Cossentino, M., & Potts, C. (2002). A case tool supported methodology for the design of multi-agent systems. *Proceedings of the International Conference on Software Engineering Research and Practice (SERP02)* (pp. 102–109).
- [14] EURESCOM Project P907 - PIR5.1. (2001). *Analysis of Existing Agent-Oriented Tools*.
- [15] Bellifemine, F., Poggi, A., & Rimassa. G. (1999). JADE - A FIPA-compliant agent framework. *The Practical Application Company Ltd.*, 97–108.
- [16] Nwana, H. S., Ndumu, D. T., Lee, L. C., & Collis, J. C. (1999). ZEUS: A toolkit for building distributed multi-agent systems. *Applied Artificial Intelligence Journal*, 1(13), 129–185.
- [17] Busetta, P., Ronnquist, R., & Hodgson, A., & Lucas, A. (1999). JACK intelligent agents components for

intelligent agents in Java. *Agentlink News*, 2, pp. 2-5.

- [18] *Indian Government Railways General Rules for all Open Lines of RAILWAYS in the Republic of India*, 2008, Administered by the Government.
- [19] Verma, A., Pattanaik, K. K., & Goel, P. P. (2014). Mobile Agent based CBTC system with moving block signalling for Indian railways. *Proceedings of the International Conference on Railway Technology: Research, Development and Maintenance*, Civil-Comp Press, Stirlingshire, UK.
- [20] Khan, M. S., & Benkrid, K. (2010). Development techniques of multi-agents based autonomous railway vehicles control systems. *Word Academy of Science, Engineering and Technology*, 558-568.
- [21] Proenca, H., & Oliveira, E. (2004). MARCS multi-agent railway control system. *Proceedings of the 9th Ibero-American Conference on AI, Puebla, Mexico Lecture Notes in Computer Science, Advances in Artificial Intelligence - IBERAMIA* (pp. 1-10).
- [22] Hu, X., Zhou, X., & Dang, J. (2006). A designing method of simulation software for Chinese train control system based on hybrid software agent model. *Proceedings of the International Conference on Machine Learning and Cybernetics* (pp. 148-153).
- [23] Tsang, C. W., & Ho, T. K. (2008). Optimal track access rights allocation for agent negotiation in an open railway market. *IEEE Trans. Intell. Transp. Syst.*, 9(1), 68-82.
- [24] Lind, J., & Fischer, K., Bocker, J., & Zirkler, B. (1999). Transportation scheduling and simulation in a railroad scenario: A multi-agent approach. *Proceedings of the 4th International Conference on Practical Appl. Intell. Agents Multi-Agent Technol* (pp. 325-344), Blackpool, U. K.
- [25] Goundan, P. R., & Jhunhunwala, A. (1999). Axle counter based block signalling for safe and efficient train operations. *Proceedings of the Conference on Vehicular Technology* (pp. 824-828).
- [26] Allison, L. R. (1951). A modern cab signalling and train control system for railroads. *American Institute of Electrical Engineers*, 70(1), 232-239.
- [27] Yong, Z., Ming, Z., & Xishi, W. (1999). Research on a train operation simulation system under moving automatic block conditions. *Journal of system simulation*, 11(3).
- [28] Zhi, Z., & Yinhang, C. (1999). The comparison of several moving block systems. *Railway signal and communication*, 35(12).
- [29] Frerichs, C., Detering, R., & Wiemann, M. (2000). The ERTMS/ETCSS specifications on Interoperability and their implementation in the Berlin-Halle/Leipzig pilot project. *ZEVI/DET Glasers Annalen*, 467-473.



Anshul Verma is a Ph.D. scholar of information and communication technology Department at ABV-Indian Institute of Information Technology and Management, Gwalior, India. He obtained master of technology degree in computer science and engineering with specialization in digital communication from ABV-Indian Institute of Information Technology and Management, Gwalior, India, in 2011. His research interests include opportunistic network, wireless ad-hoc network, formal modelling and verification, mobile-agents based computing, communication based train control system.



K. K. Pattanaik received his diploma in electronics and telecommunications from State Council of Technical Education (Orissa) in 1992, He received his bachelor of engineering in electrical and electronics engineering from Kuvempu University, Shimoga, in 1997, He received his master of technology in computer science and engineering from Motilal Nehru National Institute of Technology Allahabad (formerly MNREC) and subsequently, PhD in engineering with computer science as a major form Birla Institute of Technology-Mesra, Ranchi in the year 2010. Currently, he is serving as an assistant professor (Department of Information and Communication Technology) at ABV-Indian Institute of Information Technology & Management Gwalior, MP, India. His research interests include network traffic issues, wireless sensor network, manets, data centre network, grid computing, mobile agent based computing.