A Distributed Multi-Agent Control Model for Railway Transportation System

Ali Pouyan
School of Computer and IT Engineering
Shahrood University of Technology.
Shahrood, Iran.
apouyan@shahroodut.ac.ir

Momeneh Taban
School of Computer and IT Engineering
Shahrood University of Technology.
Shahrood, Iran.
m.taban@comp.tus.ac.ir

Sadegh Ekrami
School of Computer and IT Engineering
Shahrood University of Technology.
Shahrood, Iran.
s.ekrami@comp.tus.ac.ir

Abstract **–Management and control of railway system is a serious and complex task. A railway system is consisted of network of tracks, stations, safety tools, and trains. In order to cope with increasing train frequencies a method for organizing an appropriate schedule in distributed parties is needed. However this paper concerns the application of distributed multi-agent for presenting an appropriate model consisted of interacted distributed layers for railway transportation system. The proposed layered-based model reduces the workload of the system, which in turn leads to a real-time approach. In this method the system's efficiency increases in comparison with the centralized approach.**

Keywords–distributed; mobile agent; distributed systems; layered-base model.

I. INTROUDUCTION

During the last decade human life has changed by automation revolution. Railway system is one of the systems whose development depends inevitably on automation. Increasing use of railway transportation systems has resulted in a huge complexity system which can not be managed and controlled by human by itself. Various versions of Automatic Train Control (ATC), Automatic Train Operation (ATO), and Automatic Train Protection (ATP) are the paradigms of automation being used in current railway industries [1]; in order to have a complete interoperable railway system, more rigorous research is needed. Despite different technologies being used to implement automation in railway system, for large complex systems, distributed applications are more feasible. Moreover, because of its inherent characteristics, agent technology is capable to design distributed systems [2].

Distributed systems can be implemented using multi-agent technology because it is able to decompose the system into a variety of agent systems that can interact with each other. Accurate interoperability helps these distributed multi-agent systems to reach the main goal of the whole system. Several approaches have been proposed in controlling system with the help of agent technology [3, 4].

This paper presents a multi-agent-based model for controlling and management of railway system. This model is compartmentalized into four layers including central office, local offices, subnets and blocks. All systems applications rely on the communication between these layers. The proposed model helps to reduce system's workload and enhances communication efficiency which leads to a real-time approach. Mobile Agent technology is used to carry on the interaction between layers.

The rest of this paper is organized as follows. Section 2 presents a brief background of the concepts previously being used. Section 3, presents a general distributed model for railway system. Section 4, surveys the system from an internal point of view and determines 4 scenarios which show the real-time functionality of Mobile Agents and Agent Systems. Finally, section 5 concludes the paper with a summary and a discussion of possible future work.

II. BASIC CONCEPTS

A *distributed system* includes a set of autonomous sub-systems linked to each other. Sub-systems are able to coordinate their activities and share the system's resources, information, and etc.

An *agent* is an autonomous computer program accomplishing the assignments on behalf of an agent system.

A *multi-agent system* is a system that consists of multiple interacting intelligent agents. Multi-agent technology usually is used for modeling distributed system in order to solve the problems that are difficult for an individual agent or monolithic system to solve. This kind of system usually uses both mobile and stationary agent to perform its tasks. Different from stationary agent, mobile agent is not bound to the system it begins the work from [5].

A *distributed multi agent system* is a multi-agent system in which individual agents act on different parts of the system.

III. SYSTEM GENERAL MODEL

Most of current traffic management systems are incumbent through only a central control unit (includes central scheduling system, centralized database architecture, operators, etc.) which can cover only a little scope accurately, and only a limited range of potential delays can be performed. The main advantage of the modeled approach is a decentralized planning that relies on distributed databases, distributed processors, distributed rescheduling systems, etc. A general model of the system is shown in Fig. 1.

As it can be seen, this design is consisting of different components whose some tasks are defined as follow:

AEE: Agent Execution Environment is a place for analyzing the operational data of both stationary and mobile agents, managing them, and controlling creation, suspension and immigration of each agent due to its operational essence.

Finder: this component routes and authenticates the messages between layers. It probes for the agent data (origin, destination, etc.) and identifies the agent owner; then transmits it to its destination.

Camera and sensor: The installed detectors wrap the current state of the environment into mobile agents and transmit them to their agencies where they will be unwrapped in order to be analyzed.

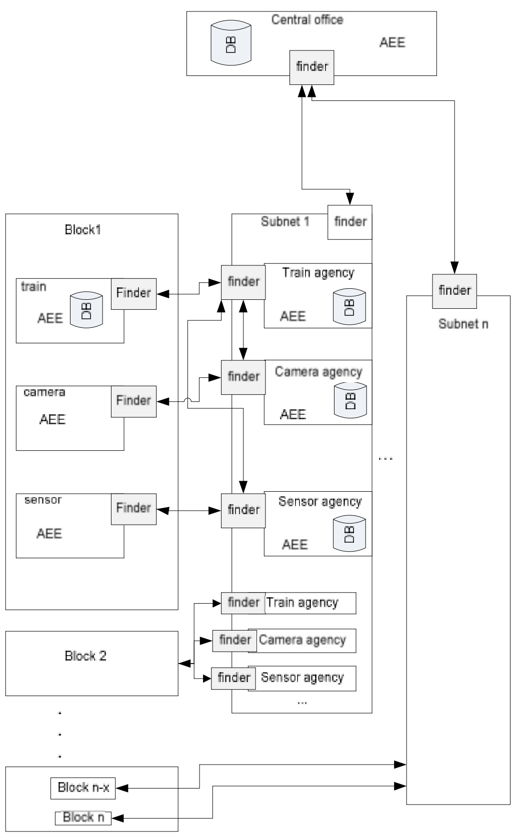

Figure 1. Presents a general model of railway system.

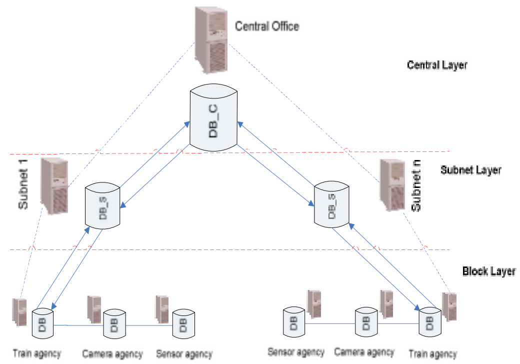

Figure 2. The schema of databases and processors.

Database (DB): Because of distributed modeling, decentralized DB method is used to balance the workload of immense amount of data and transportation task. It helps the system by reducing network traffic, having better availability and better response time. In this method each part controls its own data autonomy. As the Fig. 2 represents it clearly, our system is three layered in terms of database structure. The Central's DB at Central Office is supposed to be in charge of controlling whole databases by collecting integrated data and replicating them. In previous section, we introduced the main parts of our system; we will give a brief description of internal processes in the next section.

Block: Having the best control in an optimized way over rail roads, they are decomposed to some virtual blocks, in which different kinds of detectors (cameras, sensors, etc.) are located. The trains passing the predefined blocks are under block control. Once the trains enter the block, they are identified by that. Each train has a predefined scheduling and it is connected to its agencies in each pulses of the time to send its situation's parameters (speed, location, etc.) through mobile agents.

The whole system consists of blocks, subnets and central office, respectively. The second layer called “subnet” is a group of tripartite agencies. These three agencies (train agency, camera and sensor agency) are in charge of controlling one block. The head agency in each block is the train agency through which scheduling and rescheduling are served. After analyzing the data of camera agency and sensor agency, the output will be transmitted to the train agency.

If each train (for any reason, as mentioned in the following part), causes delay in pre-defined motion schedule, it can not be compensated only by itself; and the adjacent trains (forward, backward) also take part to fix the interval balance. This plan causes the delayed train to have more opportunities to compensate its delay so that the system would not face with dangerous condition.

IV. THE MODEL INFRASTRUCTURES

Trains are considered as equal interval chains along specified routes. If a train encounters an unusual condition and causes delay, the interval between the trains is reduced and it is a critical situation that must be avoided. In fact, upon such a situation, the compensating responsibility will be shared among the trains to balance influences on delayed train.

There are two points of view about the system, one is the system external view that is discussed in the last part, and the other one is the internal view of the system, which consists of all internal processes performed in a system. To perform internal design of a system by using agents, it is required to apply not only agents and their execution environments but also some special functions and mechanisms. This section defines some fundamental and basic processes of agents in connection with agencies trough the pseudo codes which have been showed in the Fig. 3. One of vital points in traffic management field is scheduling system which must be able to deal with problem specification, neglecting the type of technology that is used. Relating to the first point, the second essential requirement is finding an optimal solution depending on problems' type. The following part presents some probable scenarios in which the scheduling system at the train agency, subnet and Central Office are responsible for planning train traffic and finding optimized solutions. Fig. 3 describes brief functional tasks by agents and agent systems.

Trains, sensors and cameras scan environmental state and continuously send a list of data to their agencies within equal time intervals. Analysis information at the train agency takes place when information from trains, sensor agency and camera agency is received. If some potential delay is identified by train agency, different scenarios of system performance are put forward to system delayed state.

There are different critical scenarios which may happen through the train's move and affect secure movement curve (the train’s motion curve that Central Control System sends it to the train, so it can move by using this plan).

To avoid difficulty and complexity of data transmission, the control system is divided into 3 basic layers; the block layer which has the most effective role is Omni-present to deal with the trains. Subnet layer is responsible for interacting with the blocks’ layer in its authority and let them to solve their conflicts whenever the adjacent blocks need to be rescheduled; And Central Layer is designed to control not only the conflicts between subnet layers but also the issues of whole system.

```
Elem= list (train ID, train speed, train velocity, location(x,y), direction, time) // Elem is necessary information of the train
Agent1→ call ('insert', array(Elem)); // It can be sent by the train
Base=Agent2→ call ('retrieve', schedule); // It can be generated by the train agency
Function delay (Train (Elem), Base);
If (delay_result==0) {create Agent (type1) /*Agent type 1 is defined in order to inform the train for continuing its schedule/*} else {
Function curve_Train_Agency (delay_result, predefined compensating algorithms); /* Checks Block Situation, if there is a reschedule in the block calculates a new reschedule, if the reschedule plan is conflicting by another block the reschedule plan will sending its data in an agent form to the higher layer (Subnet) */
If (adjacent block conflict) {call curve_Subnet()} else{ create Agent (type2) /* Agent type 2 is defined in order to change the train's schedule/* } }
Function curve_Subnet (delay_result, predefined compensating algorithms, adjacent block situation); //calculate a reschedule for inter-block schedule.
```

Figure 3. Pseudo code of a conventional relationship between train, train agency and subnet.

To make it more clear, different scenarios are used to explain the model. Blocks are responsible to get the necessary data from trains and verify their characteristics such as ID, location, speed, etc. and compare them with the scheduled plan. The adjacent trains will keep moving even if a train delays for a couple of time intervals. However, in such cases, the plan will be rescheduled immediately by block processors and will be sent to the appropriate trains to compensate the delay. The Mobile Agent role is effectively seen in these communications. It transmits the data which have been reported by different parts such as sensors, cameras, trains. Moreover, agents interacts with each other in an environment called Block Layer in order to verify the errors caused by trains, sensors, etc.

There is a more complicated situation in which the trains in a block make a critical scenario leading to confliction with the adjacent block. In order to reduce the processing work in the block layers, the issue will be referred to the upper layer named subnet layer. The subnet has access to blocks’ databases and reschedules the system in the affected blocks and sends motion curves to the trains.

Consequently, the model will not cause any conflict with any parts, and the delayed time will be distributed among appropriate trains. The system is guaranteed to be encountered with an accident, because if so, the whole motion curve is going to be changed. In cases that problems remain unsolved and get to a critical point, the related blocks will stop the trains. In this paper we do not talk about the functions and details of curve algorithms or any compensating strategy. Our focus is mostly on communication phase, agents’ role, distributed processing, and distributed tasks.

As it is mentioned in the pseudo code, two important functions will be invoked hierarchically. If it is needed to refer to the upper layer and the agents are executed in order to find an accurate schedule. In such a case the upper processors which consist of some lower processors would not be in connection with any ordinary messages and data from the train, so these processors may run in bottlenecks. Hence, the system can save more executive

 ISBN: 978-1-61208-134-2

power. The following scenarios show how a schedule plan generates.

1) List of data which are sent by the train will be inserted into the train agency's DB. At the same time the actual schedule is invoked and is compared with the data list. If there is no difference between them an agent will be sent to the train indicating that the previous schedule should be followed.

2) Sometimes the trains available in delayed block can compensate the recognized delay through block rescheduling which is performed by train agency. And the new motion curves are sent to the trains.

3) A critical point is when the adjacent blocks cause delays, or the delayed block could not compensate the delay by rescheduling its block. In these cases, the subnet takes control of the delayed blocks. The subnet shall identify the number of operational risks, such as traffic conflicts, potential delays and bottlenecks, and then organize to avoid them by changing current time tables. Until now the subnet has performed a local planning process for its blocks.

4) As it is mentioned in Fig. 4, another probable situation is when two adjacent blocks from two adjacent subnets cause delay and the compensation process requires the subnets' coordination. In some conditions, probable problems in subnet coordination and subnet schedule can be solved via central office.

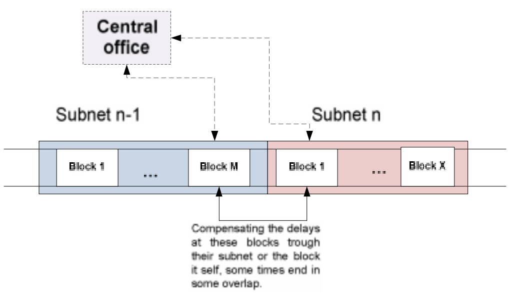

Figure 4. Role of Central Office in Controlling Subnets.

V. CONCLUSION AND FUTURE WORK

In summary, this paper presents a four-layer distributed designing to model the general behavior of railway transportation system which consists of different interacting parties. Railway is decomposed to some virtual blocks. We used detectors to control each block accurately and to avoid incidents. To model this distributed system, we used multi-agent system enjoying both mobile agents and stationary agents to reduce network traffic and enhance communication efficiency between distributed agent systems.

Future work is needed to classify and define current agents and agent systems in more details, for instance their exact framework and functions. A more detailed implementation model for future development is also required. Because, some of the aspects, such as authentication, security issue, agent system structures, etc. will become as important as our layered-based model. Finally, simulation fundamentals shall be extended via prospect experimentations.

REFERENCES

[1] K. Takeshi and H. Nakamura, "Automatic Train Protection System Depending on Data Oriented Control Method". Railroad Conference, 1993, pp. 51-57, DOI: 10.1109/RRCON.1993.292963.

[2] D.J. Mendes and M. De Assis Silva, "Mobile Agents in Autonomous Decentralized Systems". The Fourth International Symposium on Integration of Heterogeneous Systems.1999, pp. 258 – 260, DOI:10.1109/ISADS.1999.838444.

[3] B. Chena, H. H. Chengb, and J. Palenc "Integrating Mobile Agent Technology with Multi-Agent Systems for Distributed Traffic Detection and Management Systems". 2009, pp. 1-10.

[4] T.K. Ho, L. Ferreira, and K.H. Law, "Agent Applications in Rail Transportation". Proc of International Conference on Intelligent Agents Web Technologies and Internet Commerce, pp. 251-260, 2004, Vienna.

[5] A. Kavicka, V. Klima and N. Adamko, "Simulation of Transportation Logistic System Utilising Agent-Based Architecture". International journal of simulation modeling 6(2007) 1, pp. 13-24, DOI:10.2507/IJSIMM06(1)2.075.

[6] Z. Linda, B. Naadjib, and E. Aouaouche, "An Architectural Model for a Mobile Agents System Interoperability". 2007, pp. 555-566, DOI: 10.1007/978-1-4020-6270-4-46.

[7] L.M. Gambardella, A.E. Rizzoli, and P. Funk, "Agent-Based Planning and Simulation of Combined Rail/Road Transport". May 2002, Vol. 78, No. 5, 293303(2002). pp. 293-303, DOI: 10.1177/0037549702078005551.

[8] R. Leszczyna, "Anonymity Architecture for Mobile Agent Systems". Lecture Notes in Artificial Intelligence; Vol. 4659 Proceedings of the 3rd International Conference on Industrial Applications of Holonic and Multi-Agent Systems: Holonic and Multi-Agent Systems for Manufacturing. pp. 93 – 103, 2007.

[9] M. Miyoshi, M. Seki, Y. Fujiwara, Y. Sobu, and K. Toshiba Corp, "Development in Mass Transmit System", 1998, in Proceedings of the International Conference on Developments in

Mass Transit Systems (Conf. publ. No.453), pp. 196 – 201, ISBN: 0-85296-703-9.