International Journal of Safety and Security Engineering
Vol. 11, No. 6, December, 2021, pp. 671-682
Journal homepage: http://iieta.org/journals/ijsse

A Survey on Formal Specification and Verification of Smart Mass Transit Railway Interlocking System

Lokanna Kadakolmath[1*], Umesh D. Ramu[2]

[1] Acharya Institute of Technology, Bengaluru, Visvesvaraya Technological University, Belagavi 590018, India
[2] P.E.S College of Engineering, Mandya, Visvesvaraya Technological University, Belagavi 590018, India

Corresponding Author Email: lokanna@acharya.ac.in

https://doi.org/10.18280/ijsse.110607

Received: 16 August 2021
Accepted: 2 December 2021

Keywords:
computer-based interlocking system, formal specification, formal verification, modeling, rapid rail transit, reliability, safety-critical system, safety and security

ABSTRACT

Nowadays interest in Smart Mass Transit Rail has grown-up to a large extent in a metropolitan area as the need for urban mobility has increased steadily. The reliability of software being used in such mass transit rail is crucial for us, specifically when software crashes may lead to catastrophic loss of human life and assets. For example, when we travel by metro it is essential for us that the interlocking system software controlling the metros are accurate so collisions and derailment are prevented. The reliability and safety of such interlocking systems are made on the precise functional requirements specification and verification respectively. Therefore, the precise functional requirements specification and verification of such interlocking systems represent a challenge in an active research area, so in this paper, we survey various articles in this field and discuss their consequences.

1. INTRODUCTION

In a world of ever-increasing financial, environmental, and infrastructure requirements, the urban railway transportation system has become an essential part of every major city. Efficiently designed, operationally sustainable, and user-friendly urban transport systems are instrumental in urban mobility. The history of the urban railway transportation system dates back to the mid of the 19th century. In the year 1853, in London, the first and foremost rail-based metro track was launched. In the year 1867, in Mumbai, the first and foremost suburban line was launched. In the year 1868, in New York City the first and foremost elevated railway was launched. In the year 1900, in Paris, the first and foremost metro line of the Paris network was launched.

In the 1920s in all big cities of the globe, the only kind of mechanized transportation was the underground or elevated rail, and trams or surface train, and it was only for those peoples who could afford it. This happened when the big cities were enlarged to approximately 10 km distance with residents about one to two million. In those days, if any person resided away from their workplace, then it was necessary to stay near the rail tracks. In later days all factories, industrial units, and workshops were developed alongside the railway tracks. The urban railway transportation system changed the appearance of these 19th century big cities of the globe, especially in Europe and the US [1].

At the end time of the 20th century all over the world urbanization in big cities starts to grow up, and they faced the issues like the redevelopment of an existing area, the creation of newly urbanized areas, growing population, and pollution, global warming, increased traffic flow and road accidents, and roaming from one place to another within the city were very difficult. Road traffic accidents are one of the major issues in an urban area. After a comparative analysis of causes of accidents in an urban area, it is advised that urban railway transportation is one of the smart transports to avoid traffic congestion and accidents [2]. Therefore, nowadays public interest in Smart Mass Transit Rail has increased in metropolitan areas. At the start of the 21st century, the solution to all the above issues is given through the *'Smart Mass Transit System'*.

Rail Rapid Transit is surely the preferred mode for mass transport on high demand pathways in big and medium cities and leads to making growing cities more comfortable and sustainable. Rail Rapid Transit is an older and well-known part of the nationwide urban transport system. It transports a huge number of commuters from one place to another at high speed. The goal is to accomplish a high level of performance that must be escorted by a high level of safety and maximum comfort for the commuters. Rail Rapid Transit carriages run exclusively on fixed guideways in exclusive rights-of-way, and which is maybe tracked down in grade-separated tunnels, or subway, or elevated railroads.

Nowadays the technological advancement made the *'Rail Rapid Transit System'* completely dependent on Intelligent Transport System (ITS) technologies. ITS is a unified technique that carries out a wide range of transmission, self-control, detecting and tracking motor vehicles and microelectronic technologies to resolve and control traffic flow obstructions. ITS includes telematics and all types of communications in a train, between trains, and between trains and wayside locations. For the past two decades, ITS is being used in developed countries like Europe and the US. However, even so, it is a fresh conception, when developing countries like India, South Africa, Brazil, China, etc., are concerned. Therefore, nowadays Rail Rapid Transit Systems are also known as *'Smart Mass Transit Systems'*. Metro Rail, Subway Rail, Suburban Rail, Cable Car Rail, Monorail, Light Rail, and Elevated or High-Speed Rail are an example of Smart Mass

Transit Systems because nowadays these are completely dependent on the ITS technologies.

Smart Mass Transit Systems, especially metro rails are the key significance for social mobility, as societies are growing to be urbanized. Metro rails run exclusively on fixed guideways in exclusive rights-of-way, and which is maybe tracked down in grade-separated tunnels, or subway, or elevated railroads. Metro rails usually run at the normal speed of 20 to 35 km/h, and they transport 50,000 to 75,000 commuters per hour, for each direction. The headway between two metro rails is between 2 to 5 minutes. One of the advantages of the metro is decreased road traffic congestion, due to the commuters moving from motor vehicles transport mode to metro systems. This move also decreased air pollution and road traffic mishaps.

At present, there are more than 178 cities in 56 countries around the world that are constructing or planning metro lines, and there are already some lines that are started and being operated. Currently, our earth has 230 metro systems. As of 2020, according to the UITP data, 7% of installed total metro length worldwide is being automated, and the 178 metros are reported for an overall, yearly ridership of 53,768 million commuters. In a Tokyo city every year approximately 4,000 million commuters travel in metros, while in a New Delhi city every year approximately 2,000 million commuters travel in metros. In the past few years, yearly metro ridership raised universally by 8,716 million passengers (+19.5%). As of 2020, the 178 metros collectively built a fixed asset base of 642 lines for an overall distance of 13,903 km and 11,084 stations.

At present, Beijing station is the longest metro system in the globe having 699.3 km with 405 stations, while Shanghai station is the second-longest metro system in the globe having 639 km in length. Seoul station is the third-longest metro system in the globe having 466 km in length, and it is the most widely used Rail Rapid Transit system in the globe. The London Underground is the first-born metro system on the earth from 1890, and it is the fourth-longest metro system having 436 km. India also began as a strong marketplace for metro rail systems. In India, at present, 10 metropolitan areas run a metro system covering 536 km. Further, almost 750 km of the metro system and 373 km of the Rapid Rail Transit system are under development in many metropolises. A New Delhi station is the eighth-longest metro system having 389 km in length [3, 4].

Below Figure 1 shows the total number of metro systems and the location of systems inaugurated each decade from 1860-2019 [4]. These statistical data indicate the signification of research in the context of the Smart Mass Transit System.

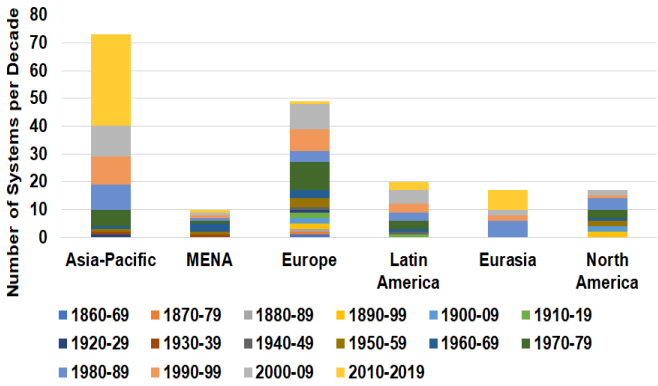

Figure 1. Metro system opening (per decade) 1860-2019

The remaining portion of our article is arranged into three sections. Section 2 illustrates the background details of the need for formal methods for railway interlocking, the challenges that were faced, while shifting to computer-based interlocking, and the modern signaling system CBTC. Section 3 describes the comparative assessment of the adoption of formal methods for interlocking systems. In section 4 we concluded the paper.

2. BACKGROUND

2.1 Formal methods for railway interlocking

In the olden times of the railway, there were no interlocking systems. Only workers at the railway stations physically observe trains and operating signals. To overcome human mistakes mechanical railway signaling was introduced. Mechanical railway signaling was quite easy to prove that railway signaling interlockings achieved what they were supposed to achieve. There were diagrams to analyze and a completed mechanical system that could be verified. The interlockings themselves were restricted in their application, possibly covering a junction, or a series of junctions such as at a station throat, on the other hand, it was all comprehensible.

Then alongside came computer-based interlocking systems. Unexpectedly, the problem was considerably more complicated. Each line of code could modify how the system performs and interlockings were growing to monitor larger areas, initiating possibilities of more communications. So how to verify it? With teams of computer professionals who were also signaling engineers going through the program line by line. To avoid this manual verification a sensible and standardized method called *'Formal Methods'* was introduced. Formal Methods are mathematical notations, that are used for functional and nonfunctional requirements specification and verification of a system [5].

Formal methods are classified into specification and verification languages. Usually, formal specification languages are used for unfolding the performance of an interlocking system as a model with specific formal semantics, and to evaluate these interlocking system models it presents their allied formal verification tools [6]. Formal methods have been in use within urban railway signaling systems and interlockings for well over 30 years. Generally, producing new or modified railway signaling and the interlocking system requires analysis of stakeholder requirements and generating a formal specification of the essential system. From this high-level specification, it is feasible to obtain certain safety requirements such as liveness properties. Later, a high-level design is developed after obtaining a specification. Once, a high-level design is get implemented, then we can verify the design with help of formal verification tools to check that the designed system will have the functionality expected by the formal specification. Therefore, the significance of formal methods of such systems relates to maximum levels of confidence, and the truthful working of the software systems contained to avoid collisions.

Modern Smart Mass Transit Systems such as metro signaling and an interlocking system are maintained by an innovative software control system called *'Communication-Based Train Control (CBTC) System'*. It focuses on means of reducing headway, confirming safety, and enhancing efficacy, and decrease in the cost of operation. These control systems

not only ensure train safety, but also integrate, interface, and automate areas of operation, driving passenger information, and examination of the same. So, the accurate performance and complete safety assurance of such control systems are foremost important. The main reasons for control system failures are improper requirements specification, design errors, incorrect implementations, and verification by human testers should be omitted with the high-level of assurance. Also, in modern-day Smart Mass Transit Systems, it is more essential to ensure not only safety requirements but data accuracy along with operative accuracy. The operative accuracy is ensured by expressing a given system in terms of state transitions by using state-based specification languages, and data accuracy is ensured by expressing how data is evolving or how they are related by using algebraic specification languages. For these reasons, formal methods for requirements specification and verification of modern-day Smart Mass Transit Rail Interlocking Systems are used in the industrial environment. Also, the importance of formal specification and verifications are rising as novel driverless or pilotless applications are developing.

The formal model of an interlocking system is designed by applying various formal specification languages. Several authors successfully used Z [7-10], B [11-14], Event-B [15, 16], VDM [8, 9, 17], CSP [11, 18], CPN [19-21] and ASM [22] for the design of formal model of an interlocking system.

Z, B, and Event-B are formal specification languages built on set theory and propositional logic; all are invented by a French computer scientist Jean-Raymond Abrial [23-25]. The *'Vienna Development Method (VDM)'* [26, 27] is a formal specification language developed at the IBM laboratory. It contains a set of tools and techniques for modeling computer programs at a very abstract level, but by using refinement techniques it can be translated into a detailed design. The *'Communicating Sequential Processes (CSP)'* [28] is a formal specification language for concurrent systems. It is built on process algebras, invented by T. Hoare. It is practically used in industrial applications for the specification and verification of concurrent systems. Colored Petri Nets (CPN) [29] is a formal specification technique for concurrent systems. It is an augmented version of Petri Nets and preserves all its properties. It has robust simulation and analysis methods. The strength and drawbacks of all these formal methods are described in the literature survey section.

2.2 Challenges

The most important challenges that are faced while adopting Computer-Based Interlocking systems are described below.

In the early 1990s, the safety requirements were the reserve of expert *'Signaling Engineers'*. What were assumed to be *'fair requirements'* were remained inaccurate. The deficiency of understanding of the need for accurate requirements and the toolset itself caused major challenges. The method or tool that they used for verification was not thought of for commercial use at the time. Also, such verification methods are expensive and time-consuming, and still, they offer only partial coverage.

The second major challenge faced in those early days is inaccurate, ambiguous, or inconsistent requirements. Throughout the development of the generic application, requirements need to be produced in natural language, so they can easily transform into the toolset code. This eliminates ambiguity and forces inconsistent requirements to be expanded to a clear form so the inconsistency can be eliminated [5].

For modern railway interlocking systems, it is extremely tough to get enough assurance by traditional verification methods. Stronger verification methods that offer superior assurance in railway interlocking software safety are desired. Also, modern railway interlocking systems have to fulfill CENELEC EN 50128 standards [30]. This standard highly suggests formal verification for verifying railway interlocking safety requirements. The third major challenge is the cost of change in modern railway interlocking systems, which occurs due to inconsistent requirements, physical development methods, outdated verification methods, and the need for expertise.

To overcome these challenges, we need to determine user requirements, so that they can be used with automation tools. This means that requirements need a high level of precision, to allow automatic processing with modern tools, and to shrink the need for physical expertise for interpretation of requirements. As we can see in Figure 2, formal requirements have high precision and need low physical expertise.

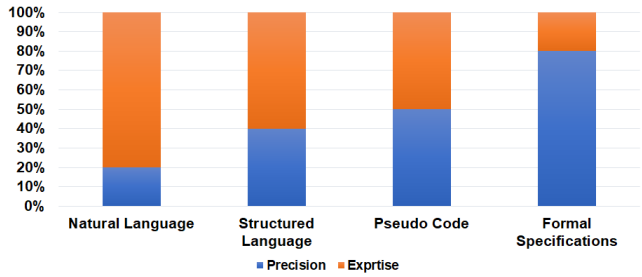

Figure 2. Precision Vs expertise required for specifications

The essential requirements are unambiguously partitioned into two parts. The first part of the requirements should be verified using functional testing and the second part should be used in safety verification. The goal of this partition is to allow verification of the different types of requirements using verification tools that are dedicated and specialized for safety assessment and functional testing, respectively.

The paper by Ferrari, et al. [31] also addressed the most important challenges that they faced while switching the development model to a model-based method from a code-based method, some of them are given below.

The software used within the railway interlocking system will adapt to precise safety specifications, and typically the railway firms utilize software coding rules to prevent the use of illegal hypotheses that could be unsafe from the safety viewpoint. Once modeling and auto-coding are accepted, the produced software will adapt to the same precise safety specifications.

A certified or proven-in-use translator is the foremost demand for the CENELEC EN 50128 standard railway interlocking system [30]. Nonexistence of such a tool, a policy must be well-defined to certify that the software code performance is entirely compatible with the model performance, and no extra incorrect events are included throughout the code synthesis stage. The verification actions should be performed at the same stage as the abstract model itself.

Railway interlocking systems are typically huge composite platforms with some cooperating entities and architectural layers. To control such complexity, their development is built on several stages of abstraction, and different models with different granularities are needed.

The paper by Knight [32] also focused on the most important challenges that they encountered while adopting safety-critical control systems. These challenges are described below.

A major cause of failures occurs because of breakdowns in the chemistry between software engineering and systems engineering. Complete methods on entire system modeling must be built so that the requirements of complete systems can be verified. Such methods should adopt a software code correctly and give high-reliability models of crucial software qualities.

Bad software specifications point to many serious failures, and they had difficulty saying correctly what software is supposed to perform. Past specification methods have not provided several aspects of the specification and even if specification methods be present, then there persists a deficiency of unification to allow an entire specification evaluation.

Systems that must function in ultra-dependable scope, for them it is not possible to perform verification by testing. Formal verification is a suitable method but is restricted in its pertinence. In the early days, high-level performance, speedy, comprehensive methods of verification was crucial.

Using past methodology development time and effort for safety-critical control systems are so risky that developing the systems that will be required in the upcoming days will be impossible. They needed a modern software methodology that must deal with both the cost and time disputes.

Security is becoming a more and more essential subject in the domain of safety-critical control systems, and it must be dealt with carefully. The CENELEC EN 50159 standard also defines safety-related communication in railway control systems [30]. It also includes several security aspects by describing cryptographic methods as well as cryptographic architectures needed for open network communication.

2.3 Communications-Based train control (CBTC) system

In the olden days, urban railway signaling and control systems have been controlled using track circuit-based signaling for more than five decades. In late 1980, the usage of a transmission-based also called cable loop system was initiated but it did not find many customers, further due to technology and environmental challenges. The failure of these technologies causes repeated train catastrophic accidents such as derailments or collisions in the last few decades.

In a few of these train catastrophic accidents, the major cause was too much speediness or the movements of the train driver who overran the stop line at the station, which occurred in Amagasaki city, Tokyo on April 22, 2005. This accident left 73 people killed and more than 456 others were injured [33].

Another worst world train accident called '*The Deadliest Accident in D.C. Metro History*', occurred on June 22, 2009. A subway train (Train 112) collides with the tail end of one more train (Train 214), causing 09 people to be killed and more than 80 others were injured [34]. The reason was a track circuit device that is considered to detect trains flopped in the zone where Train 214 was halted. This accident is shown in Figure 3. Figure 3 is a photograph by CNN.com [34].

As we have seen in the above accidents, most of the reason was not proper specification and verification of signaling interlocking system and not following standard operating rules. Also, the adoption of Computer-Based Interlocking systems in urban railways leads to automated and driverless trains, and the demand for very frequent trains with frequent stops was increased.

Figure 3. A collision of subway trains (train-on-train)

However, with technological advances in Information Technology (IT) and Telecommunication, the Rail Safety Improvement Act of 2008 suggests a '*Positive Train Control (PTC) System*' for urban rail. PTC includes a collection of state-of-the-art technologies intended to prevent accidents caused by human mistakes, track or equipment failure, and other kinds of train operator faults. PTC aims to avoid train-on-train collisions, diversions caused by too much speed, or early derailments on tracks.

A broad version of PTC is called *'Communication-Based Train Control (CBTC) System'*, which is a more complex signaling and train control system, and it is nowadays a preferred technology for Smart Mass Transit Systems like metros across the world. CBTC systems are normally operated to supervise driverless metro and suburban trains. CBTC needs train data to be directed to a central zone, which then distributes the data to all objects in the system. CBTC system makes use of a *'Global Positioning System (GPS)',* Balise, or transponder tags to track train position. Track-circuits are used as a secondary device to detect train positions. CBTC makes use of formal specifications to indicate system properties and prove them using model checking techniques. The reliability of the CBTC system is tested by using model checking techniques like *'Deterministic and Stochastic Petri nets (DSPNs)'*, which takes performance data (delay and packet loss) as a parameter to evaluate the system reliability [35].

3. COMPARATIVE ASSESSMENT

3.1 Literature review

The goal of our literature review is to identify various formal methods proposed earlier for the specification, design, and verification of an urban railway interlocking system. We surveyed relevant papers including journals, book chapters, conferences, and white papers from the railway industry. Some of them are discussed in the below sub-sections.

In the 20th century, authors focused on the usage of formal methods for safety requirements specification of solid-state interlocking systems and formal verification of these specifications. In those olden days, the relevance of formal methods in railway interlocking systems was recommended by Railway Industry Association Standard 23 (RIA 23). In those days only a few formal methods existed such as FOREST, ExSpect, JACK, Z, CSP, and VDM. The requirements

specified using these methods are somewhat imprecise and automated tools often fail, for the reason of a state explosion problem. Z notation is not suitable to express non-functional requirements, such as performance, reliability, size, usability, and also timed or concurrent behavior. These limitations are avoided by merging Z with other formal methods. FDR is not a symbolic model checker, it is a refinement checker, it cannot check non-reachable states. Also, CSP is not well suited for describing the railway control table because it's hard to understand and validate requirements by the practitioners. But these formal methods are the fundamental building blocks for the acceptance of formal methods to railway interlocking systems.

3.1.1 Literature review of journal papers

The paper by Basile et al. [36, 37] described their knowledge using the Uppaal SMC formal method for modeling, and statistical model checking of satellite-based ERTMS L3 moving block interlocking system. The usability and suitability of the Uppaal SMC formal method for urban railway interlocking systems are also demonstrated. Furthermore, they suggested a hopeful way for future enhancement, wherein they calculate Spatio-temporal study by Uppaal SMC.

The paper by Vanit-Anunchai [19, 21] described a method designed for building a CPN model of a railway interlocking system. Signaling Layout and Interlocking models are the two chunks derived from the CPN model. The CPN model imitates the signaling layout and stores data about how each part of the equipment attaches. The Interlocking part does not depend on the Signaling Layout and has the control table specifications written in twelve ML functions. They used XSLT for transforming the control table in XML into ML functions. They also demonstrated the analysis of three scenarios. Produced state spaces are verified against the required property that there is no train on two successive track circuits.

The paper by Keming et al. [15] used the Event-B formal method to construct a multilayer formal model and refinement strategy by analyzing the requirements, properties, and events of system function of railway interlocking system. They also refined their constructed formal model by using the theorem-proving technique and verified system properties. Finally, they tested the accuracy of the model by simulation.

The paper by Zafar et al. [7-9] described Z and VDM-SL formal methods on an abstract level for the specification of the moving block interlocking system. They used graph theory to model static parts of the moving block interlocking system, and these static parts are combined to define the complete interlocking system. Finally, specifications are analyzed using the Z EVS and VDM-SL Tools.

The paper by Haxthausen et al. [38, 39] described the development of a railway interlocking system through model-based and formal verification methods. The user formulates a depiction of the application-specific constraints in a domain-specific language for every railway interlocking control system to be produced. These descriptions are translated into a feasible systemC model and later is assembled into an object code. Formal verification is achieved with the help of the three foremost methods used in separate levels. In Level 1, using static analysis the systemC model is verified for reliability. In Level 2, using a bounded model checking the safety requirement specifications are verified. In Level 3, the object code is verified.

The paper by Kanso et al. [40] described a verification method for an urban railway interlocking system specified in ladder logic. In this method, they first developed a mathematical model and translated ladder logic into this mathematical model, later generated safety properties using the railway track layout, and finally, safety properties are verified using the SAT solver.

The paper by Janota [10] discussed the practical application of formal methods for safety requirements specification and verification of railway interlocking systems. For design, a small domestic railway network was used to express the use of concrete formal depictions using Z notation.

The paper by Atkinson and Cunningham [41] depict the FOREST method for safety requirements specification and validation of specifications using MAL prover. To form its proofs with rules the MAL prover utilizes a tableau technique. The authors clarified its usage invalidation via the prover to prove that a railway signaling safety specification has several required safety properties.

The paper by Basten et al. [42] depict the usage of the ExSpect tool for modeling and verifying railway interlocking specifications in the Interlocking Specification Language (ISL). ExSpect toolkit was built on the theory of colored Petri nets. A method for converting ISL into ExSpect was proposed. Also, some motivating themes for future research were recognized.

The paper by Bernardeschi et al. [43, 44] described several *'abstraction methods'* to resolve the drawbacks of safety requirements validation by using existing tools. These abstraction methods were defined within a verification method that was used to verify computer-based railway interlocking specifications. Also, they discussed how to resolve the *'state explosion problem'* using these abstraction methods. By using their ACTL abstract specification method they reduced the number of states in the state machine from one million to 77294 states.

The paper by Lukács et al. [45] described a framework for automated verification and specification of domain properties of railway interlocking systems. They focused on formalizing domain properties.

3.1.2 Literature review of book chapters

The paper by Laursen et al. [46] investigated a generic model for modeling and model checking of a distributed railway interlocking system using UPPAAL. The generic model has three variants. The first variant comprises the least essential operations such as moving a train, reserving a segment, and locking a point. The second variant uses an accurate operational order. The third variant expands the first variant by canceling reserving a segment and unlocking a point. To test their accuracy and contrast their performance, verification tests are conducted on instances of all variants. By varying the size of networks, the scalability of all three variants was also investigated.

The paper by Nazaruddin et al. [47] proposed a model checking method for verifying the safety properties of railway interlocking systems. Their proposed method is implemented on a timed automaton and they used computation tree logic formulations for specifying safety properties. The model checking is performed with the UPPAAL model checker, and verification simulation results are also presented.

The paper by Peleska et al. [48] described a well-organized method for data validation of geographical interlocking systems (IXLs). Their proposed method exposes defilements of configuration rules very quickly. The verification speed has

been accomplished by using classical global CTL model checking algorithms and converting LTL formulations specifying rule defilements to CTL formulations.

The paper by de Almeida Pereira et al. [13] described the usage of the B formal method by specifying the properties of a relay-based interlocking system. With the help of propositional logic rules, they described how system states are progressing. The properties are verified by using the ProB model checker.

The paper by Limbrée, and Pecheur [49] described a compositional verification method. Medium and large stations are controlled by a network of interlockings due to the limitations of the current technology and due to some availability constraints. So, in this article, they used a compositional verification method which is a perfect fit for the formal verification of such a network. This principle can be further used to split the interlocking entity thereby allowing to tackle the state space explosion problem. Their role is twofold. First, they proposed an algorithm that can automatically split an interlocking entity into smaller entities called components. Second, they proposed a catalog containing an exhaustive list of interfaces or contacts allowing the compositional verification of all our graphs of interlockings. They used the OCRA tool to perform the compositional verification of the system.

The paper by Macedo et al. [50] described how to avoid a state space explosion drawback by applying the compositional model checking approach. The goal of this method is to use linear cuts at the network modeling level to break down a model of an interlocking system. With a linear cut, to formally verify an interlocking system that controls multiple railway stations a compositional model checking method is used.

The paper by Haxthausen and Østergaard [51] described how to avoid a state space explosion drawback by using a static checker. Compared to model checker static checker consumes less execution time and memory for the RobustRailS verification toolset. Also, the static checker error messages are more useful than the counterexamples generated by traditional model checkers.

3.1.3 Literature review of conference papers

The paper by Stankaitis and Iliasov [52] described ongoing research that aims to enhance a current SafeCap verification method to verify heterogeneous railway interlocking systems. To improve the Event-B template formal model, they examined up-to-date technical contributions and stated the existing work. Permitting railway signal engineers to utilize formal methods to verify railway interlocking systems with no prior knowledge is the main motive of the SafeCap method. This research was carried out in a collaboration with *'Siemens Rail Automation'* to confirm that outcomes have likely to be suitable in the industrial setting.

The paper by Fantechi et al. [53] described the UMC framework for formally modeling and analyzing geographically distributed interlocking systems. By proving the models that obey safety requirements, the formal verification competencies of the UMC framework have been approved.

The paper by Rakesh and Kadakolmath [54] described the PyNuSMV method to verify the finite-state model of urban railway interlocking system. The properties are to be verified are specified by using CTL formulas.

The paper by Durmus et al. [20] used Petri Nets to decrease the possibility of logical errors. In this paper, they presented a supervisory control theory applied to a railway signalization and interlocking design using a Mathematica-based program (PetriBox) that produces Petri Net supervisors automatically for a given sample railway yard. An important drawback of using this method is the possibility of state explosion for large models.

The paper by Wang et al. [55] described a SCADE method for system design, specify safety requirements, and achieve verification. This method is recognized to make better quality and efficacy of interlocking software in practice.

The paper by James et al. [11] described a CSP||B formal model to graphically examine and verify scheme plans of railway interlocking systems. The models are generated using a range of abstraction methods that analyze the large-scale plans possible. They used the ProB tool to verify safety requirements to avoid collision and derailment.

The paper by James and Roggenbach [56] described the usage of SAT-based model checking methods for formal verification of urban railway interlocking systems. First, the propositional model was used to show how the performance of an interlocking system can be modeled using a finite machine. Later, they proposed algorithms to carry out an SAT-based model checking on a finite machine. To stay away from state-space explosion drawbacks they used the slicing method.

SAT-based model checkings are the best method to verify *'Trackguard Westrace Mk II'* interlocking systems because Westrace is programmed in Ladder Logic. The conversion from ladder logic into propositional logic, and applying SAT-based model checking methods reduce the complexity of the state-space explosion problem.

The paper by Kiss and Jánosi-Rancz [16] proposed a structured model approach and develop distributed railway interlocking system by using the Event-B formal method. Their proposed method combines the railway entity models specified by using Event-B with the communication models of session types. A prediction of session type specifications and verification techniques ensure global safety by using verification of local entities.

The paper by Winter et al. [57] described a model checking method to automatically check an interlocking design concerning safety. The major worries of this method are the problem size and the usefulness of existing tools. They examined both of these glitches to work with the least model of an interlocking design and to raise the efficacy of the model-checking method by utilizing domain knowledge of their precise application. The NuSMV model checker was used to verify the safety requirements of an interlocking design. To keep away from the state space explosion drawback and to raise the performance of the model checker for bigger case studies, they suggested several optimizations such as reduced OBDDs (ROBDDS).

The paper by Winter and Robinson [22] described a feasibility analysis on model checking of railway interlocking systems. The railway interlocking tables are modeled by using ASM (Abstract State Machines) formal notation, and then it is automatically transformed into NuSMV code for verifying safety requirements. To verify the safety requirements, they decomposed a well-built interlocking system into less significant ones without reducing the scope of the verification method. To complete the feasibility analysis, they used an automated theorem prover called NP-Tools to verify the properties and judge the outcomes against NuSMV outcomes. NuSMV is one of the best formal methods for urban railway interlocking systems because it is easy to write and understand

safety requirements, and counterexamples provided by the NuSMV tool are easy to analyze.

The paper by Winter [18] described how to model and verify railway interlocking system by using a formal modeling language called *'Communicating Sequential Processes (CSP)'* and their correlated model checker called *'Failures-Divergences Refinement (FDR)'*.

The paper by Bernardeschi et al. [58] described the safety requirements of a computer-based interlocking system using ACTL (action-based CTL) temporal logic. The state explosion problem of model checking is partially solved by zooming those parts of the system on which the safety requirements have to be verified. The verification is then performed on the subsystems focused by this zooming technique, and the results of the verification are then extended to the global system.

The paper by Bechina et al. [59] described why a Prolog programming language is a strong candidate for formal specification of safety requirements and interlocking behavior. The fairness of the Prolog programming language makes it a strong language to express complex railway interlocking systems.

The paper by Anselmi et al. [60] described their knowledge about formal specification and verification by validating the functional specifications of a computer-based interlocking system model project manufactured by Ansaldo Trasporti. They modeled the system using Labelled Transition Systems (state automata), and functional specifications are stated using process algebra (CCS), and certain safety requirements are stated using ACTL temporal logic. The specified properties are verified using JACK. The main problem with this method is the state explosion problem. They tried to minimize this problem by using some abstraction techniques.

The paper by Banci et al. [61] described a statechart method to generate detailed specifications of railway interlocking systems. They tackled the problem from a geographically distributed viewpoint. Specifically, their model was combined by models of distinct physical objects (points, signals, etc.) that all together fulfill the interlocking guidelines, with no centralized database. The foremost objective of this article was to examine the feasibility of the geographic method intended for the evolution of a distributed interlocking system.

The paper by Hansen [17] described the validation of the VDM model of a railway interlocking system by using simulation in ML. The model development shows how hypotheses are obtained for a non-trivial system. The stages from a predicative VDM model to an executable ML program are also summarized.

3.1.4 Literature review of white papers

In the white paper entitled *"Safety Verification Methods for Rail Control Software"*, [62] they outlined the safety verification practices usually applied in rail control projects in the US, Sweden, and France. Many of the signal engineering practices in the US originate from the design of relay-based systems, and interlocking software is also presented in ladder logic. The rail signaling systems in Sweden, for both metro and mainline, is a mix of traditional relay-based interlockings and computerized systems. The rail control community in France is more influenced by the modern software development practices used in other industries. It has also embraced formal methods to a higher degree, and it is a standard component in rail control software development. In all these three projects formal methods are introduced as a complement and partial replacement of the traditional testing and review safety verification methods.

In the white paper entitled *"Automated Verification and Validation of Signaling Systems in PTC and CBTC Environments"*, [63] to measure the safety of modern railway signaling systems, they described a state-of-the-art solution for automatic verification and validation tasks, and the same solution they applied to PTC and CBTC signaling systems all over the globe. The foundation for this solution is to have requirement specifications that capture the signaling rules in a formalized way, enabling automated processing by computer programs. It reduced the effort of verification to a simple configuration task.

In all the above literature survey papers, the authors put effort to reduce the '*state-space explosion problem'*, by using different techniques like efficient data structure, abstraction, bounded model checking, etc. The state-space explodes when the size of potential state space increases by a multiplicative factor for the increase in design size.

3.2 Usage of 'B Formal Method' in the urban railway industry

The paper by Woodcock et al. [12] and Lecomte [14] investigated the industrial usage of formal methods in safety-critical systems relevance. Railway transportation is the main domain of industrial usage of formal methods, it was proved from their survey of 62 projects. The B formal language is one of the most popular industrial usages of the formal method adopted by urban railway systems across the globe. The B formal language focuses on developing software systems from requirements specification, by modification through execution and automated code generation, in conjunction with verification at every phase. Some industrial usages of the formal method adopted by urban railway systems are given below.

The "*SACEM system of RER Line A in Paris*" was the first urban railway that adopted industrial usage of formal methods. Since 1989, it was in full operation and the speed of each train on the track was endlessly controlled by an automatic train protection system developed by GEC Alsthom Transport (Nowadays called Alstom Transport). Every day, it continuously ensured the safety of 0.8 million commuters. Formal specifications of the functional requirements were made by using the B formal method.

The "*Metro Line 14 in Paris*" was another urban railway that adopted industrial usage of formal methods. Since 1998, it was in full operation, and in those days, it is the only fully automatic metro line in Paris. It was controlled by an automatic train operation system developed by Matra Transport (Nowadays called Siemens Transport). Every hour, this 8.5 km line carries the traffic of 40,000 commuters with a gap of 85 seconds between trains during peak hours. This automatic train operation system was designed by using 86,000 lines of Ada, and 115,000 lines of B specifications.

The "*Roissy Charles de Gaulle Airport Shuttle in Paris*" was also another urban railway that adopted industrial usage of formal methods. Since 2007, it was in full operation, and it was the first fully automatic light train. This automatic train operation system was designed by using 158,000 lines of Ada, and 183,000 lines of B specifications.

The "*Singapore North-East Line (NEL) and Delhi Metro Line 8 (Magenta Line)*" are some more examples of fully automated driverless metro's they also used the B formal method for formal specifications of the functional requirements of an interlocking system.

The above success stories were witnessed about the adoption of the *'B formal method'* for formal specifications of the functional requirements and verification of an urban railway signaling interlocking system.

3.3 Usage of 'Prover Formal Method' in the urban railway industry

The paper by Borälv and Stålmarck [64, 65] described the industrializing Prover formal method in railways. Stålmarck approach is defined through an enormous number of nested assumptions in proof. It is also a patented normal deduction technique with a unique proof-theoretic concept of proof depth. Execution of this approach is known as 'Prover'. Since 1990 it was used as a proof engine in many commercial tools. Also, nowadays it is incorporated in a formal verification framework known as NP-Tools. Stålmarck's approach is an industrial usage of formal verification for railway interlocking and aircraft systems and some other industrial control systems.

In 1997, using a formal approach with NP-Tools and SVT (Sternol Verification Tool), ADtranz (ABB Daimler-Benz Transportation Signal) was given a task to supplement the verification stage of the Lago (Madrid metro station) interlocking system. The translation from a sternol interlocking system into a formal model in NP-Tools is ensured by SVT. According to the ADtranz report using the above formal methods reduce the verification stage by 90%.

After several years, for the reason of technology change, during the early 2000s, the Trackguard Westrace Mk II interlock was started to in use and the amount of configuration data increased ten-fold. As a result, it was a higher potential for error, and the opportunity to apply the new technology onto railway transportation through modular signaling, formal methods were investigated. By this period, accepting the need for accurate requirements had matured, and compared to the early days of the project, tool support had progressed significantly and formed a long track record in railway signaling. Also, the process had progressed from just using formal proof to also incorporating the production of the data, analysis of the data, and sign-off verification. To meet these issues, Prover Technology introduced Prover Trident which is based on the collective usage of the following three solutions: PiSPEC, Prover iLock, and Prover Certifier. PiSPEC is used to define generic requirement specifications for a particular railway interlocking system. Prover iLock specifies configuration and an automated generation of interlocking data, involving simulation and verification. Prover Certifier is a sign-off verification tool industrialized in agreement with SIL 4, building the safety mark for the location, using automated formal proof.

Siemens Rail Automation in joint partnership with Prover Technology verified the importance of these tools in terms of feasibility on UK infrastructure (The journey from Shrewsbury to Crew was reduced to less than 40 minutes). Some more examples of industrial usage of the Prover formal method adopted by urban railway systems across the globe are given below [66].

The *"New York City Transit (NYCT) subway system"* is one of the oldest and biggest transport systems in the globe. It involves more than 1,100 km of track and its 25 subway lines deliver service to 469 stations, and every day it carries more than 4 million commuters. In early 1999, it began to modernize its signaling system by switching to computerized solid-state interlockings and installing CBTC. Computerized interlocking systems from various vendors including MELLOCK, by Mitsubishi, Westrace Mk II, by Siemens, Microlok II, by Ansaldo STS, and iVPI, by Alstom, were used. Formal verification of NYCT's safety requirements is provided with Prover iLock Verifier.

The *"Paris Metro"* is a rapid transit system in the Paris urban zone, France. Thales developed PMI computerized interlocking systems for its Line 3 branch 3bis. In 2009 Prover Technology co-operated with RATP (Regie Autonome des Transports Parisiens) in building formal verification tools to encounter RATP's need for safety verification of interlocking software. In later days Line 12 South, Line 8, Line 12 North, and line 1 were also verified using Prover Technology.

In 2013 Ansaldo STS developed Microlok II interlocking systems for Roslagsbanan mainline railway in Roslagen, Stockholm County, Sweden. Ansaldo STS used Prover Technology for the development and safety approval of interlocking software. Now 40,000 commuters per day travel through this mainline and it runs at 80 km per hour.

3.4 Industrial experience of adopting formal methods to urban railway

The paper by Ferrari et al. [31] recap the experience of General Electric Transportation Systems (GETS) for adopting formal methods. GETS is a railway signaling company that agreed to accept model-based tools, such as Simulink or Stateflow, and SysML, for the development of their products. The company met many disputes mainly about the verification of the software and the incorporation of the tools inside the active process. Formal or semi-formal methods like semantic constraints, model-based testing, and abstract interpretation, and Structured development solutions were adopted to encounter disputes.

The paper by Fantechi et al. [67] investigated the motives behind adopting formal methods to the railway signaling domain, especially why industrial people trust B formal method for requirements specification by highlighting researchers and engineers experience in a Politecnico di Milano and Italian State Railway FS combined project. Also, they described the comparative analysis report of various formal methods.

The paper by Fantechi et al. [68] discussed in the context of Intelligent Transport Systems (ITS), how to use formal methods to model and examine railway control systems. Also, elaborated on the exclusive session of *'Intelligent Transport Systems'* organized within the ISOLA12 conference. This exclusive session stems from conversations organized within the *"ERCIM Working Group on Formal Methods for Industrial Critical Systems (FMICS)"*, which finally finished up in a direction focused on Intelligent Transport Systems. The influences to this exclusive section are further elaborations of the papers submitted at that conference track, concentrating on the usage of formal methods in railway signaling interlocking systems.

In this exclusive session, the paper by Ferrari et al. [69] suggest an innovative method by merging semi-formal modeling and methods described from product line engineering to adopt early formalization of requirements specification for a CBTC driver-less metro signaling and train control system. By using natural language processing (NLP) and rapid prototyping methods the evocation of requirements was done.

The *"European Shift2Rail"* project proposal is to the modernization of ERTMS / ETCS (European Rail Traffic Management System / European Train Control System). They

consist of automatic driving, moving block distancing, and satellite-based train positioning. The Shift2Rail project believes that formal methods are essential to the establishment of reliable, safe, and secure technological developments. The *'European Union's Horizon 2020'* framework suite sanctioned the fund for the Shift2Rail project [70].

The paper by Ferrari et al. [71] presented a systematic review on *'Formal Methods and Tools'* for the development of ASTRail, under the Shift2Rail project proposal. They analyzed different formal methods and tools used for the development of railway signaling and interlocking system during the last decades. They validated 114 research papers, 8 industrial railway projects and they consulted academics and practitioners from different railway companies to respond to their questionnaire. Based on their survey they short-listed 14 formal methods for modeling, specification, and verification of railway interlocking systems.

In the *"25th International Conference on Formal Methods (FMICS)"* held virtually in Vienna on 2-3 September 2020, they conducted a general survey of 130 international well-known specialists to collect their knowledge about Formal Methods. Based on their responses and comments were advised a precise analysis of the earlier, current, and upcoming usage of formal methods in industry, education, and research. According to a huge majority of the specialists, it is concluded that *'Formal Methods'* are basic building blocks for developing safety-critical software for railway signaling and control systems [72].

3.5 Strength of formal methods in urban railway

Formal methods are still an active research area in the field of railway signaling interlocking. Which is evidenced by the following Figure 4. In which the number of articles from both academic and industry that used various formal verification tools for verifying formal model of railway interlocking system is illustrated. The below Figure 4 is the search result of the DBLP Computer Science Bibliography [73], which comprises references to more than one million papers from journals and conferences.

The strength of formal methods for the urban railway is also certified by CENELEC / IEC Standards. The computer software for railway interlocking and control systems is defined by EN 50128 standard. This standard highly suggests formal specification languages such as Z, CSP, VDM, Temporal Logic, B, etc. for the specification of railway interlocking and control system components with SIL 4 (Safety Integrity Level). Formal verification tools such as NuSMV, SPIN, Prover, etc. are also highly suggested for verification of railway interlocking and control systems.

Similarly, the safety requirements specification for the sanction of microelectronic devices in the railway signaling field is defined by EN 50129 standard, and the requirements specification and validation of RAMS (Reliability, Availability, Maintainability, and Safety) for all railway signaling and control systems are defined by EN 50126 standard [30, 67].

The paper by Mazzanti et al. [74, 75] described comparative analyses of the performance of various formal verification tools for verifying the liveness property of CBTC based urban railway interlocking systems for avoiding deadlock. The following Table 1 shows their verification results in terms of execution time in seconds.

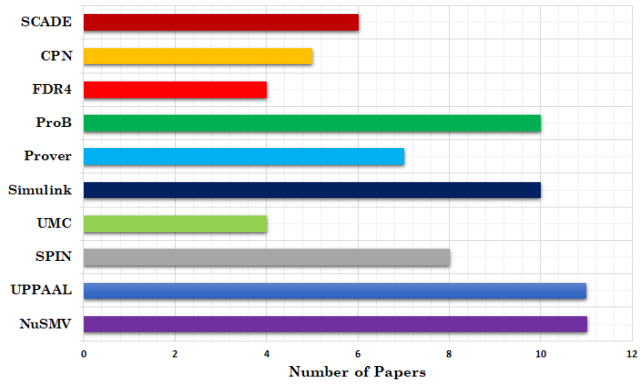

Figure 4. Active formal verification tools are used by both academics and practitioners

Table 1. Verification results

Verification Tool	Execution Time
NuSMV	2.9 to 43 seconds
SPIN	13 to 47 seconds
FDR4	15 seconds to 20 minutes
UPPAAL	16 seconds
CADP	29 seconds
UMC	38 to 86 seconds
mCRL2	2 minutes to 19 minutes
TLA+	3 minutes
ProB	32 minutes

4. CONCLUSION

In this survey article, we presented the history of successful application of formal methods for safety requirements specification and verification of Smart Mass Transit Railway Interlocking System. We also showed the challenges that were faced with improper specifications and shifting from traditional verification to formal verification, and how to overcome those challenges. Many success stories like B formal language and industrial usage of formal methods have risen the confidence about utilizing formal methods in the urban railway industry. Also, the intensity of adopting formal methods for safety requirements specification, and verification of Smart Mass Transit Railway Interlocking System was already witnessed by citations to related papers in a thirty-year-old survey. The latest reviews have focused on innovations of both the technologies of formal methods and the railway signaling interlocking. Finally, we conclude that formal methods are the most excellent approach to provide safety, reliability, and security of Smart Mass Transit Railway Signaling Interlocking Systems.

REFERENCES

[1] Mohan, D. (2008). Mythologies, metro rail systems, and future urban. Transport, Economic and Political Weekly, 43: 41-53. https://www.jstor.org/stable/40277079

[2] Suleiman, G.M., Younes, M.K., Ergun, M., Al Omari, K. (2021). Effect of transportation parameters on traffic accident in urban areas comparison study of ANFIS with statistical analysis. International Journal of Safety and Security Engineering, 11(2): 129-134. https://doi.org/10.18280/ijsse.110201

[3] Global Mass Transit. Information & analysis on the global mass transit industry. https://www.globalmasstransit.net/index.php, accessed on Dec. 20, 2020.
[4] UITP. Statistics Brief - World Metro Figures 2018. https://cms.uitp.org/wp/wp-content/uploads/2020/06/Statistics-Brief-World-metro-figures-2018V3_WEB.pdf, accessed on Dec. 10, 2020.
[5] Milne, A. Formal methods for signalling interlockings | Rail Engineer. https://www.railengineer.co.uk/formal-methods-for-signalling-interlockings/, accessed on Jan. 20, 2021.
[6] Almeida, J.B., Frade, M.J., Pinto, J.S., de Sousa, S.M. (2011). An overview of formal methods tools and techniques. Rigorous Software Development, 15-44. https://doi.org/10.1007/978-0-85729-018-2_2
[7] Zafar, N.A., Khan, S.A., Araki, K. (2012). Towards the safety properties of moving block railway interlocking system. Int. J. Innovative Comput., Info & Control, 8(7): 5677-5690.
[8] Zafar, N.A. (2006). Formal model for moving block railway interlocking system based on un-directed topology. In 2006 International Conference on Emerging Technologies, pp. 217-223. https://doi.org/10.1109/icet.2006.335983
[9] Zafar, N.A. (2006). Modeling and formal specification of automated train control system using Z notation. 2006 IEEE International Multitopic Conference, pp. 438-443. https://doi.org/10.1109/inmic.2006.358207
[10] Janota, A. (2000). Using Z specification for railway interlocking safety. Transport Engineering. 28(1): 39-53.
[11] James, P., Moller, F., Nguyen, H.N., Roggenbach, M., Schneider, S., Treharne, H., Williams, D. (2013). Verification of scheme plans using CSPB. In International Conference on Software Engineering and Formal Methods, 189-204. https://doi.org/10.1007/978-3-319-05032-4_15
[12] Woodcock, J., Larsen, P.G., Bicarregui, J., Fitzgerald, J. (2009). Formal methods: Practice and experience. ACM Computing Surveys, 41(4): 1-36. https://doi.org/10.1145/1592434.1592436
[13] de Almeida Pereira, D.I., Deharbe, D., Perin, M., Bon, P. (2019). B-specification of relay-based railway interlocking systems based on the propositional logic of the system state evolution. In International Conference on Reliability, Safety, and Security of Railway Systems, 11495: 242-258. https://doi.org/10.1007/978-3-030-18744-6_16
[14] Lecomte, T. (2009). Applying a formal method in industry: A 15-year trajectory. In International Workshop on Formal Methods for Industrial Critical Systems, Springer, Berlin, Heidelberg, 5825: 26-34. https://doi.org/10.1007/978-3-642-04570-7-3
[15] Keming, W., Zheng, W., Chuandong, Z. (2018). Formal modeling and data validation of general railway interlocking system. WIT Transactions on The Built Environment, 181: 527-538. https://doi.org/10.2495/CR180471
[16] Kiss, T., Jánosi-Rancz, K.T. (2016). Developing railway interlocking systems with session types and Event-B. In 2016 IEEE 11th International Symposium on Applied Computational Intelligence and Informatics (SACI), pp. 93-98. https://doi.org/10.1109/SACI.2016.7507347
[17] Hansen, K.M. (1994). Validation of a railway interlocking model. In International Symposium of Formal Methods Europe. Springer, Berlin, Heidelberg, pp. 582-601. https://doi.org/10.1007/3-540-58555-9_117
[18] Winter, K. (2002). Model checking railway interlocking systems. Australian Computer Science Communications, 24(1): 303-310.
[19] Vanit-Anunchai, S. (2018). Modelling and simulating a Thai railway signalling system using Coloured Petri Nets. International Journal on Software Tools for Technology Transfer, 20(3): 243-262. https://doi.org/10.1007/s10009-018-0482-9
[20] Durmuş, M.S., Yıldırım, U., Söylemez, M.T. (2012). Automatic generation of Petri net supervisors for railway interlocking design. In 2012 2nd Australian Control Conference, pp. 180-185.
[21] Vanit-Anunchai, S. (2009). Verification of railway interlocking tables using coloured Petri nets. In the tenth Workshop and Tutorial on Practical Use of Coloured Petri Nets and the CPN Tools, DAIMI PB, 590: 139-158.
[22] Winter, K., Robinson, N.J. (2003). Modelling large interlocking systems and model checking small ones. 26th Australasian Computer Science Conference (ACSC 2003), Melbourne, Australia, pp. 309-316.
[23] Diller, A. (1994). Z: An Introduction to Formal Methods. (2nd ed.), John Wiley and Sons.
[24] Abrial, J.R. (1996). The B-Book: Assigning Programs to Meanings. (1st ed.), Cambridge University Press. https://doi.org/10.1017/CBO9780511624162
[25] Abrial, J.R. (2010). Modeling in Event-B: System and Software Engineering. Cambridge University Press. https://doi.org/10.1017/S0956796812000081
[26] Jones, C.B. (1990). Systematic software development using VDM. Prentice Hall International Series in Computer Science.
[27] Bjorner, D. (1978). The Vienna Development Method: The Meta-Language. Lecture Notes in Computer Science. Springer-Verlag.
[28] Schneider, S. (2000). Concurrent and Real-time systems. John Wiley and Sons.
[29] Jensen, K. (1997). Coloured Petri nets: basic concepts, analysis methods and practical use. Springer Science & Business Media. https://doi.org/10.1007/978-3-642-60794-3
[30] CENELEC. Welcome to CENELEC – European Committee for Electrotechnical Standardization. https://www.cenelec.eu, accessed on Dec. 20, 2020.
[31] Ferrari, A., Fantechi, A., Gnesi, S., Magnani, G. (2013). Model-based development and formal methods in the railway industry. IEEE Software, 30(3): 28-34. https://doi.org/10.1109/MS.2013.44
[32] Knight, J.C. (2002). Safety-critical systems: challenges and directions. 24th International Conference on Software Engineering (ICSE 2002), Orlando, FL, USA, pp. 547-550.
[33] Times. At least 73 dead, 456 hurt in Japan train derailment. The New York Times. https://www.nytimes.com/2005/04/26/world/asia/at-least-73-dead-456-hurt-in-japan-train-derailment.html, accessed on Dec. 10, 2020.
[34] CNN. Six killed in the Washington-area Metro train collision. http://edition.cnn.com/2009/US/06/22/washington.subway.crash/index.html, accessed on Dec. 10, 2020.

[35] Zhu, L., Yao, D., Zhao, H. (2018). Reliability analysis of next-generation CBTC data communication systems. IEEE Transactions on Vehicular Technology, 68(3): 2024-2034. https://doi.org/10.1109/TVT.2018.2870053

[36] Basile, D., ter Beek, M.H., Ciancia, V. (2018). Statistical model checking of a moving block railway signalling scenario with Uppaal SMC. In International Symposium on Leveraging Applications of Formal Methods, pp. 372-391. https://doi.org/10.1007/978-3-030-03421-4_24

[37] Basile, D., ter Beek, M.H., Ferrari, A., Legay, A. (2019). Modelling and analysing ERTMS L3 moving block railway signalling with simulink and Uppaal SMC. In International Workshop on Formal Methods for Industrial Critical Systems, 11687: 1-21. https://doi.org/10.1007/978-3-030-27008-7_1

[38] Haxthausen, A.E., Peleska, J., Kinder, S. (2009). A formal approach for the construction and verification of railway control systems. Formal Aspects of Computing. 23(2): 191-219. https://doi.org/10.1007/s00165-009-0143-6

[39] Vu, L.H., Haxthausen, A.E., Peleska, J. (2014). Formal Modeling and Verification of Interlocking Systems Featuring Sequential Release. Communications in Computer and Information Science, Springer, 476: 223-238. https://doi.org/10.1007/978-3-319-17581-2-15.

[40] Kanso, K., Moller, F., Setzer, A. (2009). Automated verification of signalling principles in railway interlocking systems. Electronic Notes in Theoretical Computer Science, 250(2): 19-31. https://doi.org/10.1016/j.entcs.2009.08.015

[41] Atkinson, W., Cunningham, J. (1991). Proving properties of a safety-critical system. Software Engineering Journal, 6(2): 41-50. https://doi.org/10.1049/sej.1991.0006

[42] Basten, T., Bol, R., Voorhoeve, M. (1995). Simulating and analyzing railway interlockings in ExSpect. IEEE Parallel & Distributed Technology: Systems & Applications. 3(3): 50-62. https://doi.org/10.1109/m-pdt.1995.414843

[43] Bernardeschi, C., Fantechi, A., Gnesi, S., Larosa, S., Mongardi, G., Romano, D. (1998). A formal verification environment for railway signaling system design. Formal Methods in System Design. 12(2): 139-161. https://doi.org/10.1023/a:1008645826258

[44] Bernardeschi, C., Fantechi, A., Gnesi, S., Mongardi, G. (1996). Proving safety properties for embedded control systems. Dependable Computing — EDCC-2, 321-332. https://doi.org/10.1007/3-540-61772-8_46

[45] Lukács, G., Bartha, T. (2019). Construction of formal models and verifying property specifications through an example of railway interlocking systems. Pollack Periodica, 14(2): 39-50. https://doi.org/10.1556/606.2019.14.2.4

[46] Laursen, P.L., Trinh, V.A.T., Haxthausen, A.E. (2020). Formal modelling and verification of a distributed railway interlocking system using UPPAAL. In International Symposium on Leveraging Applications of Formal Methods, 12478. https://doi.org/10.1007/978-3-030-61467-6_27

[47] Nazaruddin, Y.Y., Tamba, T.A., Pradityo, K., Aristyo, B., Widyotriatmo, A. (2019). Safety verification of a train interlocking timed automaton model. IFAC-PapersOnLine, 52(15): 331-335. https://doi.org/10.1016/j.ifacol.2019.11.696

[48] Peleska, J., Krafczyk, N., Haxthausen, A.E., Pinger, R. (2021). Efficient data validation for geographical interlocking systems. Formal Aspects of Computing, 1-31. https://doi.org/10.1007/978-3-030-18744-6_9

[49] Limbrée, C., Pecheur, C. (2019). A framework for the formal verification of networks of railway interlockings-application to the Belgian railway. Electronic Communications of the EASST, 76. https://doi.org/10.14279/tuj.eceasst.76.1077

[50] Macedo, H.D., Fantechi, A., Haxthausen, A.E. (2017). Compositional model checking of interlocking systems for lines with multiple stations. In NASA Formal Methods Symposium, 10227: 146-162. https://doi.org/10.1007/978-3-319-57288-8_11

[51] Haxthausen, A.E., Stergaard, P.H. (2016). On the use of static checking in the verification of interlocking systems. In International Symposium on Leveraging Applications of Formal Methods, 9953: 266-278. https://doi.org/10.1007/978-3-319-47169-3_19

[52] Stankaitis, P., Iliasov, A. (2017). Safety Verification of Modern Railway Signalling with the SafeCap Platform. In 2017 IEEE International Symposium on Software Reliability Engineering Workshops (ISSREW), pp. 153-156. https://doi.org/10.1109/issrew.2017.66

[53] Fantechi, A., Haxthausen, A.E., Nielsen, M.B.R. (2017). Model checking geographically distributed interlocking systems using UMC. In 2017 25th Euromicro International Conference on Parallel, Distributed And Network-Based Processing (PDP), pp. 278-286. https://doi.org/10.1109/pdp.2017.66

[54] Rakesh, L., Kadakolmath, L. (2018). Modeling and formal verification of SMT rail interlocking system using PyNuSMV. In 2018 4th International Conference on Recent Advances in Information Technology (RAIT), pp. 1-8. https://doi.org/10.1109/RAIT.2018.8388983

[55] Wang, X., Tang, T., Liu, S. (2013). Study on Modeling and Verification of CBTC Interlocking System. 5th IET International Conference on Wireless, Mobile and Multimedia Networks, pp. 350-354. https://doi.org/10.1049/cp.2013.2439

[56] James, P., Roggenbach, M. (2011). Automatically verifying railway interlockings using SAT-based model checking. Electronic Communications of the EASST, 35: 1-17. http://dx.doi.org/10.14279/tuj.eceasst.35.547

[57] Winter, K., Johnston, W., Robinson, P., Strooper, P., Van Den Berg, L. (2006). Tool support for checking railway interlocking designs. In Proceedings of the 10th Australian Workshop on Safety Critical Systems and Software, pp. 101-107.

[58] Bernardeschi, C., Fantechi, A., Gnesi, S. (1997). Formal verification of safety requirements on complex systems. In Safe Comp 96. Springer, London, 21-30. https://doi.org/10.1007/978-1-4471-0937-2_2

[59] Bechina, A., Hermle, J., Siormanolakis, M. (1996). Using Prolog for a railway control system. In Fourth International Conference on the Practical Application of Prolog, London, UK, pp. 19-30.

[60] Anselmi, A., Bernardeschi, C., Fantechi, A., Gnesi, S., Larosa, S., Mongardi, G., Torielli, F. (1995). An experience in formal verification of safety properties of a railway signalling control system. In Safe Comp 95 Springer, London, 474-488. https://doi.org/10.1007/978-1-4471-3054-3_33

[61] Banci, M., Fantechi, A., Ginesi, S. (2005). Some experiences on formal specification of railway

interlocking systems using statecharts. In Train International Workshop at SEFM2005. Koblenz Germany.
[62] Prover. Safety Verification Methods for Rail Control Software. https://www.prover.com/expertise/#white-papers, accessed on May 10, 2021.
[63] Prover. Automated Verification and Validation of Signaling Systems in PTC and CBTC Environments. https://www.prover.com/expertise/#white-papers, accessed on May 10, 2021.
[64] Borälv, A., Stålmarck, G. (1998). Prover technology in railways. In FMERail Workshop 1, Stockholm, Sweden, Prover Technology AB. http://www0.cs.ucl.ac.uk/research/renoir/funded_activities/fmerail.html.
[65] Borälv, A., Stålmarck, G. (1999). Formal verification in railways. In Industrial-Strength Formal Methods in Practice (Formal Approaches to Computing and Information Technology (FACIT)), Michael, G. H. and Bowen, J. P. (eds), Springer-Verlag London, pp. 329-350. https://doi.org/10.1007/978-1-4471-0523-7
[66] PROVER. References. https://www.prover.com/references/, accessed on May. 10, 2021.
[67] Fantechi, A., Fokkink, W., Morzenti, A. (2013). Some trends in formal methods applications to railway signaling. Formal Methods for Industrial critical Systems: A Survey of Applications, 61-84. https://doi.org/10.1002/9781118459898.ch4
[68] Fantechi, A., Flammini, F., Gnesi, S. (2014). Formal methods for railway control systems. International Journal on Software Tools for Technology Transfer, 16(6): 643-646. https://doi.org/10.1007/s10009-014-0342-1
[69] Ferrari, A., Spagnolo, G.O., Martelli, G., Menabeni, S. (2014). From commercial documents to system requirements: an approach for the engineering of novel CBTC solutions. International Journal on Software Tools for Technology Transfer, 16(6): 647-667. https://doi.org/10.1007/s10009-013-0298-6
[70] ter Beek, M.H., Gnesi, S., Knapp, A. (2018). Formal methods for transport systems. International Journal on Software Tools for Technology Transfer, 20(3): 237-241. https://doi.org/10.1007/s10009-018-0487-4
[71] Ferrari, A., ter Beek, M.H., Mazzanti, F., Basile, D., Fantechi, A., Gnesi, S., Trentini, D. (2019). Survey on formal methods and tools in railways: The ASTRail approach. In International Conference on Reliability, Safety, and Security of Railway Systems. Springer, Cham, pp. 226-241. https://doi.org/10.1007/978-3-030-18744-6_15
[72] Kondakova, L. Formal methods for industrial critical systems 2020. PROVER. https://www.prover.com/formal-methods/formal-methods-for-industrial-critical-systems-2020/, accessed on Jan. 20, 2021.
[73] Computer Science bibliography. https://dblp.uni-trier.de/, accessed on Nov. 20, 2020.
[74] Mazzanti, F., Ferrari, A., Spagnolo, G.O. (2018). Towards formal methods diversity in railways: An experience report with seven frameworks. International Journal on Software Tools for Technology Transfer, 20(3): 263-288. https://doi.org/10.1007/s10009-018-0488-3
[75] Mazzanti, F., Ferrari, A. (2018). Ten Diverse Formal Models for a CBTC Automatic Train Supervision System. Electronic Proceedings in Theoretical Computer Science, 268: 104-149. https://doi.org/10.4204/EPTCS.268.4