Survey on Formal Methods and Tools in Railways: The ASTRail Approach

Alessio Ferrari[0000−0002−0636−5663]1, Maurice H. ter Beek[0000−0002−2930−6367]1, Franco Mazzanti[0000−0003−4562−8777]1, Davide Basile[0000−0002−7196−6609]1,2, Alessandro Fantechi[0000−0002−4648−4667]1,2, Stefania Gnesi[0000−0002−0139−0421]1, Andrea Piattino[3], and Daniele Trentini[3]

[1] ISTI–CNR, Pisa, Italy
{a.ferrari,m.terbeek,s.gnesi,f.mazzanti}@isti.cnr.it
[2] Università di Firenze, Firenze, Italy
{davide.basile,alessandro.fantechi}@unifi.it
[3] SIRTI S.p.A., Genova, Italy
{a.piattino,d.trentini}@sirti.it

Abstract. Formal methods and tools have been widely applied to the development of railway systems during the last decades. However, no universally accepted formal framework has emerged, and railway companies wishing to introduce formal methods have little guidance for the selection of the most appropriate methods and tools to adopt. A work package (WP) of the European project ASTRail, funded under the Shift2Rail initiative, addresses this problem, by performing a survey that considers scientific literature, international projects, and practitioners' perspectives to identify a collection of formal methods and tools to be applied in railways. This paper summarises the current results of this WP. We surveyed 114 scientific publications, 44 practitioners, and 8 projects to come to a shortlist of 14 methods considered suitable for system modelling and verification in railways. The methods and tools were reviewed according to a set of functional, language-related, and quality features. The current paper extends the body of knowledge with a set of publicly available documents that can be leveraged by companies for guidance on formal methods selection in railway system development.

Keywords: formal methods · model-based development · railways

1 Introduction

The railway field is characterised by its rigorous development processes and its robust safety requirements. During the last decades, formal methods and tools have been widely applied to the development of railway systems (cf., e.g., [14, 20, 17, 11, 5, 21, 29, 6, 1, 15, 12, 25, 8, 18, 28, 23, 4, 22, 9, 24]). Formal methods are mentioned as highly recommended practices for SIL 3–4 platforms [10, 14] by the CENELEC EN 50128 standard for the development of software for railway control and protection systems. The extensive survey on applications of formal methods by Woodcock et al. [30], which includes a structured questionnaire submitted

to the participants of 56 projects, also identified the transport domain, including railways, as the one in which the largest number of projects including applications of formal methods has been performed. Relevant examples are the usage of the B method for developing railway signalling systems in France, like, e.g., Line 14 of the Paris Métro and the driverless Paris–Roissy Airport shuttle [1]. Another is the usage of Simulink / Stateflow for formal model-based development, code generation, model-based testing, and abstract interpretation in the development of the Metrô Rio ATP system [12]. Many projects have been also carried out, often in collaboration with national railway companies, for the verification of interlocking systems [29, 16, 18, 28, 27, 7].

Also the EU's Shift2Rail initiative[4] considers formal methods to be fundamental to the provision of safe and reliable technological advances to increase the competitiveness of the railway industry. In particular, a specific call was issued asking for an analysis of the suitability of formal methods in supporting the transition to the next generation of ERTMS/ETCS signalling systems, which will include satellite-based train positioning, moving block distancing, and automatic driving. The Horizon 2020 Shift2Rail-RIA-777561 project ASTRail[5] (SAtellite-based Signalling and Automation SysTems on Railways along with Formal Method and Moving Block Validation) responds to this call. As partners of this project, we are involved in a specific work package (WP) of the ASTRail project, focussing on the contribution of formal methods to address this challenging transition; this WP operates in the following two phases:

1. An *analysis phase* dedicated to a comparison and evaluation of the main formal methods and tools that are currently being used in the railway industry to guarantee that software bugs do not jeopardise safety;
2. An *application phase* in which selected formal methods are used to model and analyse two main goals addressed by the project, namely moving block distancing and automatic driving, in order to validate that the methods are not only able to guarantee safety issues, but also—more in general—the long term reliability and availability of the software.

This paper reports on the first phase. It illustrates the results from a survey based on 114 publications and 8 projects, and a questionnaire filled in by 44 practitioners. Based on the results of the survey, a set of 14 formal tools have been analysed according to a set of functional, language-related, and quality features. Given the extensive amount of work, this paper only summarises the results. The interested reader can refer to our public deliverable [13] for further insights.

The remainder of the paper is structured as follows. In Sect. 2, an overview of the approach is provided. In Sects. 3–5, the results of a literature review, projects review, and questionnaire are presented. In Sect. 6, the tools review is presented. Sect. 7 provides final remarks.

[4] shift2rail.org
[5] astrail.eu

2 Context: Formal Methods and Tools in ASTRail

In this section, we briefly describe the context of our paper, namely the ASTRail project and its specific concern for formal methods and tools.

2.1 ASTRail Objectives

ASTRail is one of the Shift2Rail initiatives to increase the competitiveness of the European railway industry, in particular concerning the transition to the next generation of ERTMS/ETCS signalling systems, which will include satellite-based train positioning, moving block distancing, and automatic driving. ASTRail aims to introduce recent scientific results and methodologies as well as cutting-edge technologies from other transport sectors, in particular avionics and automotive, in the railway sector, leveraging on formal methods and tools for careful analyses of the resulting novel applications and solutions in terms of safety and performance.

One of the main focuses of ASTRail concerns the usage of the global navigation satellite system (GNSS) [26] for onboard train localisation. While satellite-based positioning systems have been in use for quite some time now in the avionics and automotive sectors to provide accurate positioning and distancing, the current train separation system is still based on fixed blocks (a block is the section of the track between two fixed points), implemented by specific equipment along the lines. One of ASTRail's aims is to define a *moving block* signalling [2] (according to which a safe zone around the moving train can be computed, thus optimising the line's exploitation) and to perform its hazard analysis. For this solution to work, it requires the precise absolute location, speed, and direction of each train, to be determined by a combination of sensors: active and passive markers along the track, as well as train-borne speedometers. One of the current challenges in the railway sector is to make such moving block signalling systems as effective and precise as possible, leveraging on an integrated solution for signal outages (think, e.g., of tunnels) and the problem of multipaths [26]. A related aim of the project is to study the possibility of deploying the resulting precise and reliable train localisation to improve *automatic driving* technologies in the railway sector.

2.2 Formal Methods and Tools in ASTRail

WP4 of the ASTRail project—discussed in this paper—aims to identify, on the basis of an analysis of the state of the art, of the past experiences of the involved partners and on work done in previous projects, the candidate set of formal and semi-formal techniques that appear as the most adequate to be used in the different phases of the conception, design, and development of railway systems in general, and of the class of signalling systems that is the subject of the ASTRail project in particular. In the following, when we will use the general term formal method, we will implicitly include also semi-formal methods, i.e., those methods that use languages for which the semantics is not formally defined but depends

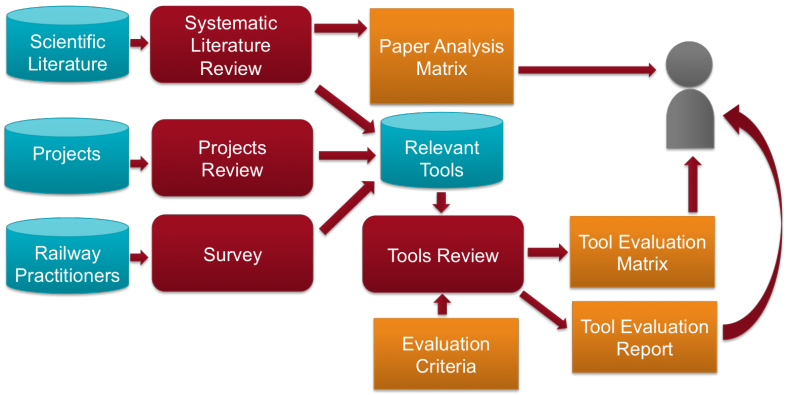

Fig. 1. Overview of the approach adopted in the analysis phase of WP4

on its execution engine. Furthermore, given that in practice a formal method always needs a support tool to be practically applicable, we will use the terms formal methods and formal tools interchangeably.

Figure 1 presents the overall approach applied in the context of this analysis phase. To address the goal of identifying the most mature formal / semi-formal languages and tools to be applied for the development of railway systems, we first performed a benchmarking task, by gathering information from three different sources: Scientific Literature, information from other Projects, and Railway Practitioners. Information from these sources were gathered through a Systematic Literature Review (SLR), a Projects Review and a Survey submitted to practitioners in the form of a questionnaire. The information was used to identify a set of main formal and semi-formal tools that appear to have been used in the railway domain (Relevant Tools in Fig. 1). Specifically, scientific literature was used as a primary source, since it provides more extensive information for guidance in the selection of relevant formal methods, while other projects and railway practitioners were used as sources to complement the information from the literature review. Furthermore, Evaluation Criteria for the different tools were defined based on collaboration between academic and industrial partners. These were applied to carefully evaluate the selected tools in a Tools Review.

The SLR produced a Paper Analysis Matrix (included as Annex 1 in our deliverable [13]), which may support the identification of the possible tools to be used depending on the specific railway system to be developed, and depending on the life-cycle phase to address. Furthermore, a Tool Evaluation Matrix (Annex 2) was defined for the different tools based on the tools review, and a Tool Evaluation Report (Annex 3), which provides details about the evaluated tools. The Tool Evaluation Matrix aims to support the selection of a formal or semi-formal tool for the railway problem at hand, based on specific preferences selected by the user of the matrix, concerning different evaluation criteria (e.g., functionalities supported by the tool, flexibility, usability) and guided by

the information from the Paper Analysis Matrix. The Tool Evaluation Report provides details to perform a more informed selection.

3 Literature Review on Formal Methods in Railways

The primary goal of the systematic literature review (SLR) was to identify the most mature formal and semi-formal methods to be applied in railway development. The SLR was conducted based on the guidelines of Kitchenham [19]. Performing a SLR requires to define a search string (e.g., "formal methods" and "railways") to automatically retrieve scientific papers from search engines, such as Scopus and SpringerLink, and to extract the data of interests from the relevant papers. The complete report of the SLR, including search string and data analysis procedures, can be found in the project's deliverable. Here, we present the most relevant results.

The search was conducted on the 7th of December, 2017, while the analysis and data extraction were performed during the following months. From the initial search, and a first analysis of the abstracts of the papers, we identified a set of 411 potentially relevant papers to use for data extraction. Given the large amount of literature, and given that the focus of ASTRail is not on interlocking systems, we decided to focus solely on studies that do not deal exclusively with interlocking (hence, 124 papers focussing mainly on interlocking were excluded from our analysis). We manually analysed 294 papers to check their quality and to identify shorter versions of other papers in the set. We excluded 180 studies of low quality, according to our quality checklist, or which turned out to be shorter versions of other papers from the set. In the end, a set of 114 papers was used for data extraction. Therefore, in the following, we report on the data extracted from 114 high-quality, and non-interlocking studies.

When appropriate, the statistics in the following sections will distinguish between the total number of papers considered in the review, and the papers that had either an industrial evaluation, or that led to actually developed products. These papers, identified as IND/DEV in the statistics, were considered more important, since they show evidence of industrial maturity of a certain method or tool.

3.1 Languages from the Literature Review

Figure 2 reports the results in terms of number of papers that use certain semi-formal and formal languages. The list of languages is extensive, and, in the statistics, we do not report on languages that appeared in only one non-industrial paper. The most used input language, according to the analysed papers, is UML. This is a semi-formal language, which is often used in the early phases of system design, and it is typically translated into a formal language, like, e.g., the B language, in the considered studies. State Machines or Statecharts, in their different dialects, such as Simulink/Stateflow, are also frequently used. Also more formal

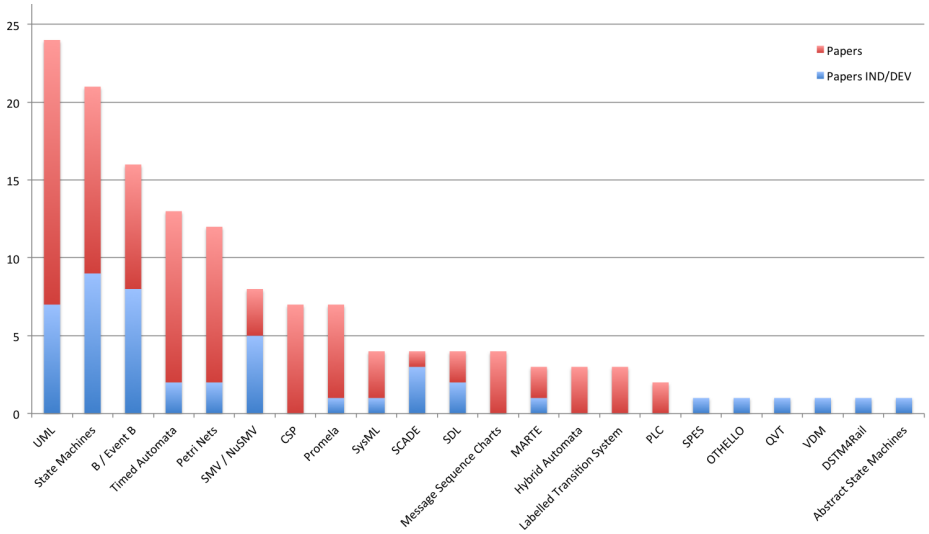

Fig. 2. Languages cited in the literature

languages, like Timed Automata, Petri Nets, CSP and Promela occur in a non-negligible amount of papers. However, these formal languages are mainly used in academic papers, while few industrial papers use them. Indeed, industrial papers tend to privilege State Machines, UML and B/Event B, or the SCADE language. It is also worth noticing that some industrial papers use several specific modelling languages, e.g. DSTM4Rail, that are used only in the context of the paper, but not in purely academic papers.

3.2 Tools from the Literature Review

Figure 3 reports the results in terms of number of papers that use a certain support tool. The list of tools mentioned in these papers is extremely extensive and each paper uses a different combination of methods and tools. Therefore, in the statistics we consider solely those tools for which there are at least two papers using the tool. The most used tools are those that belong to the B family: by summing up the contributions of Atelier B and ProB, we have 13 papers using these tools (Rodin is normally used in combination with Atelier B or ProB). By summing up the contribution of the two tools, they also dominate in industrial studies. These B method tools are followed by Simulink, UPPAAL, NuSMV, SPIN and other tools. We do not report the complete list of identified tools, since this is particularly long, and because here we are interested in identifying the most used tools for industrial studies in railways.

Interestingly, tools such as UPPAAL and SPIN, which appear frequently in the papers, are less frequent in industrial papers, in which, besides Atelier B, we see a greater usage of NuSMV, Simulink, Statemate and SCADE. We also see

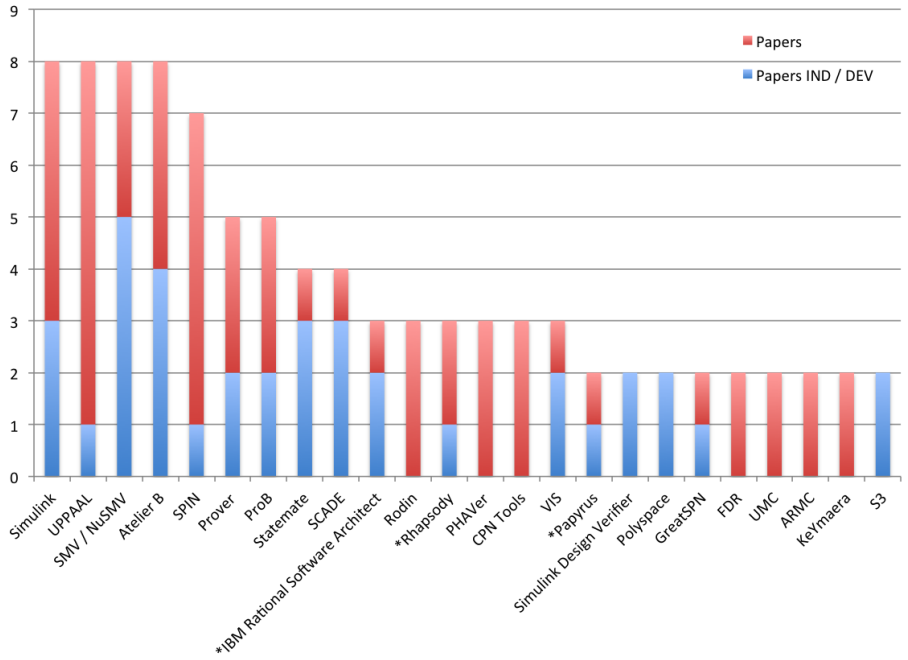

Fig. 3. Tools cited in the literature (tools marked with * support semi-formal modelling only, and do not have formal verification capabilities).

that, among the industrial papers, NuSMV appears to be more frequently used than other tools, such as, e.g., Simulink, which is inherently more industry oriented. We argue that this may be related to the particular capability of NuSMV to deal with the formal verification of large, realistic systems. Simulink is more oriented to modelling and simulation, and its formal verification tool, Simulink Design Verifier, although used in industrial works, has been rarely used for formal verification of large systems, but more of sub-components [12]. It should be noted, however, that in the inspected industrial papers, modelling and formal verification with NuSMV was not performed by railway practitioners, but by formal methods experts [9]. This suggests that the usage of state-of-the-art formal verification in industry still requires the support of formal methods experts to be actually effective in practice.

Overall, we notice that there is a large fragmentation of the papers in terms of used tools, and even the most used tools appear in no more than eight papers. This indicates that in the literature there is no clear, indisputable evidence or direction about which tools to employ in railway system development, and many tools may be adequate for the same purpose.

3.3 Maturity of Formal Methods for Railways

To identify the most mature tools, we consider the papers that are marked as IND/DEV, which indicate studies with industrial participation. We recall that the answer to this question is given for the railway context, and for non-interlocking systems. If we consider solely the tools and languages used in industrial papers, the most mature languages appear to be State Machines/Statecharts, UML and B/Event B. The literature shows an acceptable amount of evidence in this sense, with more than five industrial scientific publications for each language. Furthermore, non-industrial works also confirm the dominance of these languages. Less evidence is available for tools. If we arguably consider a tool to be industrially mature if it is used in at least two industrial studies, then the tools that can be considered mature are: Simulink, NuSMV, Atelier B, Prover, ProB, SCADE, IBM Rational Software Architect, Polyspace, and S3. Statemate also appears to be mature, but there is no recent work using the tool, and the tool appears not to be maintained anymore by IBM. A similar situation occurs for VIS, which does not appear to be used in recent publications, and does not appear to be currently maintained.

As mentioned, these considerations on tools are based on fragmented evidence from the literature, and no empirically grounded answer can be given on the most appropriate tools to employ for railway software development. However, in the context of ASTRail, this information was considered sufficient to be used as first guidance for selecting relevant tools to be evaluated during the tool review. It should also be noticed that these conclusions are applicable solely based on the published evidence, and do not take into account possible experience performed in industry with formal tools, if they do not have an associated scientific publication. To have an insight on tools that may be neglected by the literature, we complement the SLR with a Projects Review and a Survey with railway practitioners, which are presented in the following sections.

4 Projects on Formal Methods and Railways

The projects review has been based on the identification of projects from the last twenty years that have addressed the use of formal methods and tools in railway applications. The list of projects was identified based on pointers from the papers analysed in the SLR, and based on the knowledge of the authors. The available documentation for each project, like papers and web pages, has been examined in order to list the formal methods used. We found 14 projects which, starting from 1998 to this day, have addressed the use of formal methods and tools in railway applications. Among those projects, only 8 are not dealing solely with interlocking-related applications, namely: CRYSTAL, Deploy, DITTO, EuRailCheck, MBAT, OpenCOSS, OpenETCS-ITEA2, and PERFECT.

Figure 4 shows the adopted modelling languages, while Fig. 5 shows the tools used. The two figures substantially confirm the information extracted by the SLR, with a prominence of the “B eco-system”, but otherwise confirming the industrial preference to UML/SysML as modelling languages, followed by

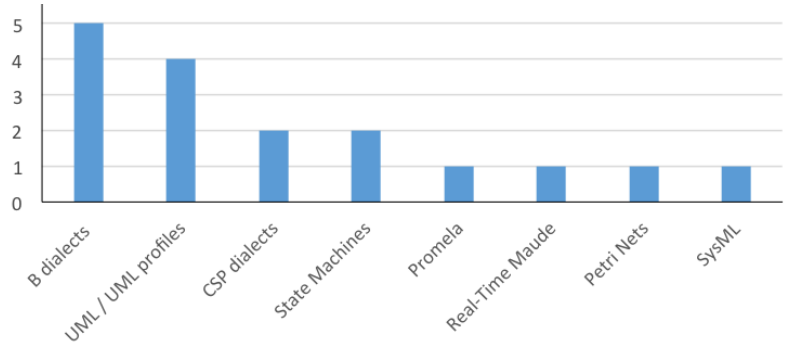

Fig. 4. Languages used in the projects

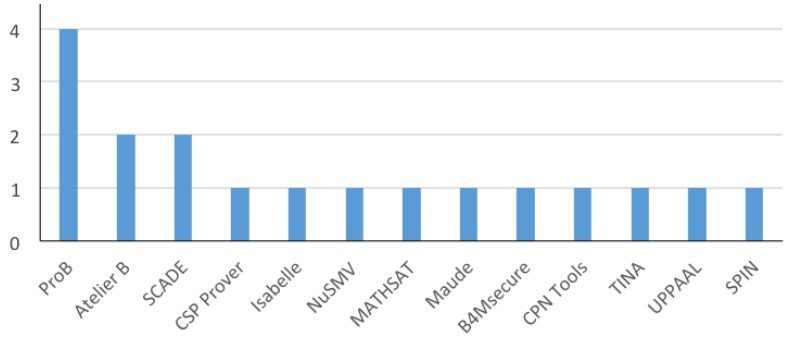

Fig. 5. Tools used in the projects

different state machine-based languages, and the importance of a commercial tool such as SCADE, emerging from a number of academic tools, mostly dedicated to formal verification.

5 Survey with Practitioners

For the non-trivial task of obtaining a significant amount of data from industrial stakeholders, a survey was carried out by means of a structured questionnaire, submitted to the participants of the RSSRail'17 conference[6], which is normally attended by academics and practitioners interested in applying formal methods in railways, and as such a promising source for a population sample that might be able to provide a well-informed judgment. We have reported and discussed the detailed results from the questionnaire in a recent paper [3]. Here, we report the ones that are more relevant in the context of this paper.

One of the goals of the questionnaire was to identify the current uptake of formal and semi-formal methods and tools in the railway sector according to

[6] http://conferences.ncl.ac.uk/rssrail/

the experience of practitioners. The first part of the questionnaire was dedicated to identify the respondents in terms of affiliation and experience in railways and in using formal/semi-formal methods and tools. The 44 respondents are balanced between academics (50%) and practitioners (50%, of which 47.7% from railway companies and 2.3% from aerospace and defense). A large percentage of respondents had several years of experience in railways (68% more than 3 years and 39% more than 10 years) and in formal methods (75% more than 3 years, 52% more than 10 years), which confirms that the sample provides informed opinions on the proposed questions.

Tools Among the various questions, the respondents were also asked to list the tools used in the context of their projects. We believe it is interesting to separate the results of industrial respondents from those of academics. In Fig. 6, we can see that the large majority of industrial and academic respondents mentioned tools belonging to the B method family (e.g. B, ProB, Atelier B, Event B, RODIN). Actually, there are only slightly more industrial users than academic users in our sample, but we recall that the academic users were asked to report on their collaborative projects with industry. Other methods and tools mentioned by both groups are the Matlab toolsuite, including Simulink and Stateflow, SCADE, Petri nets/CPN tools and Monte Carlo Simulation: the overlap between tools used in industry and in academia is actually limited to these five. Industrial users named a few other tools as well, whereas a large list of other tools has been named by academics, with popular model checkers like NuSMV and SPIN leading this list. An interpretation of this can be that a frequent pattern of collaboration between academia and industry includes the academic support in adopting advanced formal verification techniques inside a collaborative project.

Quality aspects Figure 7 reports the most relevant quality aspects that a tool should have to be applied in railways. The maturity of the tool (stability and industry readiness) is considered to be among the most relevant quality aspects by 75% of the respondents, followed by learnability by a railway software developer (45.5%), quality of documentation (43.2%), and ease of integration in the CENELEC process (36.4%). Overall, the most relevant quality aspects are associated with the usability of the tool. Less relevant are deployment aspects, such as platforms supported (9.1%) and flexible license management (11.4%). Interestingly, also the low cost of the tool (13.6%) appears to be a not so relevant feature. This is a reasonable finding. Indeed, the development and certification cost of railway products is high and, hence, if a company expects to reduce these costs through a formal tool, it can certainly tolerate the investment on the tool.

6 Tools Review

The main goals of the SLR was to identify the most mature formal and semi-formal methods to be applied in railways. From the analysis of the papers, we derived the following list: Simulink, NuSMV/nuXmv (latest version of NuSMV),

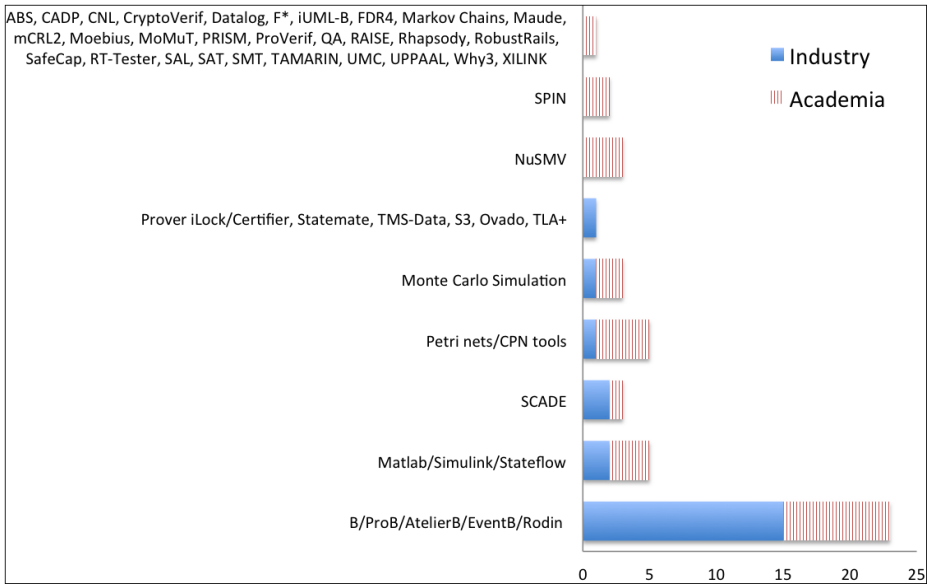

Fig. 6. Tools cited in the questionnaire (from [3])

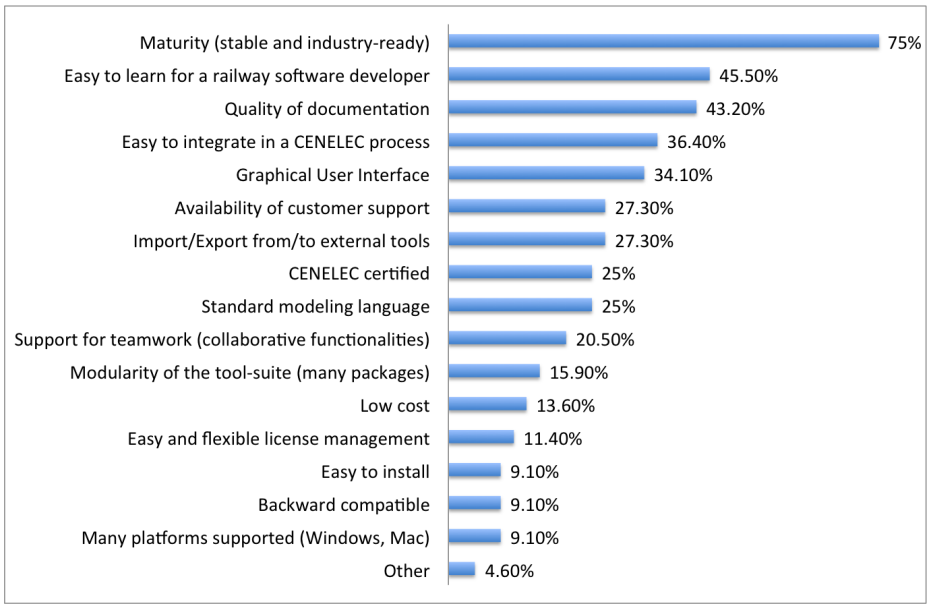

Fig. 7. The most relevant quality aspects a (semi-)formal tool should have (from [3])

Atelier B, Prover, ProB, SCADE, IBM Rational Software Architect, Polyspace, and S3. From this list, we discarded IBM Rational Software Architect because it

is just a design tool that does not allow any kind of formal verification, Polyspace because it is a static analysis tool that does not support any kind of behavioural verification, and Prover as well as S3 because of difficulties in finding sufficient documentation and inability to access a demo version within the time allocated to this project task. Thus, the subset of tools that have been selected for a further, more specific evaluation are Simulink, NuSMV/nuXmv, Atelier B, ProB, and SCADE. Furthermore, the results of the Survey with Practitioners indicate two additional tools sometimes used in railway-related industrial projects, namely SPIN and CPN Tools. Therefore, these tools have been also selected for further specific evaluation. Additionally, we are aware of other relevant tools and frameworks used in industrial projects, even if not widely used within the railway sector so far. Without the ambition to make an exhaustive coverage, and without any negative bias towards unselected tools, we wanted to experiment with a spectrum of tools and verification techniques (e.g. Logical approaches, Process Algebras, Statistical approaches) wider than that of the mainstream approaches. Therefore, we have decided to extend our specific evaluations adding to our list UPPAAL, FDR4, CADP, mCRL2, SAL, and TLA+. Finally, we have also taken into consideration one more tool, namely UMC, which—even if lacking a solid background in terms of industrial usage—has the uncommon feature of allowing a direct verification of UML-based models. We recall that, according to the SLR, UML is the most common semi-formal language used for the high-level specification of railway systems. Hence, the final list of 14 tools or frameworks selected for a deeper evaluation is as follows:

Simulink, nuXmv, Atelier B, ProB, SCADE, SPIN, CPN Tools, UPPAAL, FDR4, CADP, mCRL2, SAL, TLA+, and UMC.

Each of these tools, with the exception of SCADE[7], has been downloaded, installed, and experimented with the design and verification of simple railway-related cases studies. Part of the results were published in recent works [22, 2]. The corresponding available tool documentation has been analysed with the depth allowed by the project timeline. To evaluate the tools, a set of 34 evaluation features was considered, including functional features (e.g., formal verification, code generation), language-related aspects (e.g., support for concurrency, non-determinism), and the quality aspects also considered in the questionnaire (e.g., maturity, ease of use). The complete list of features is reported in the deliverable [13].

6.1 Results and Discussion

The tools review produced two main reference documents. A Tool Evaluation Report, in which for each feature a qualitative evaluation is given, together with

[7] In the case of SCADE, due to licensing issues, it was not possible to gain a hands-on experience within the limited timespan of the project. Hence, our evaluation is based on the analysis of the available official tool documentation and presentations, and on the experiences reported in students' assignments at the University of Florence, carried out under the ANSYS SCADE Academic Program.

Category	Name	SPIN	Simulink	nuXmv	ProB	AtelierB	UPPAAL	SCADE
Tool Flexibility	Backward Compatibility	LIKELY	LIKELY	LIKELY	LIKELY	MODERATE	LIKELY	LIKELY
	Standard Input Format	YES	PARTIAL	YES	YES	YES	PARTIAL	PARTIAL
	Import/Export	MEDIUM	LOW	MEDIUM	HIGH	MEDIUM	LOW	LOW
	Modularity of the Tool	LOW	HIGH	LOW	LOW	MEDIUM	LOW	HIGH
	Team Support	NO	NO	NO	NO	YES	NO	NO
Maturity	Industrial Diffusion	MEDIUM	HIGH	MEDIUM	RAILWAY	RAILWAY	MEDIUM	RAILWAY
	Stage of Development	YES	YES	YES	YES	YES	YES	YES
Usability	Customer Support	PARTIAL	YES	PARTIAL	YES	YES	YES	YES
	Graphical User Interface	LIMITED	YES	NO	PARTIAL	PARTIAL	PARTIAL	YES
	Easy to Use	MEDIUM	BASIC	MEDIUM	MEDIUM	ADVANCED	MEDIUM	BASIC
	Quality of Documentation	GOOD	EXCELLENT	GOOD	GOOD	EXCELLENT	GOOD	EXCELLENT
Company Constraints	Cost	FREE	PAY	MIX	FREE	FREE	MIX	PAY
	Supported Platforms	Windows, Linux, macOS	Windows, Linux, macOS	Windows, Linux, macOS	Windows, Linux, macOS	Windows, Linux, macOS	Windows, Linux, macOS	Windows
	Complexity of License Management	EASY	ADEQUATE	EASY	EASY	EASY	MODERATE	ADEQUATE
	Easy to Install	YES	YES	YES	YES	YES	YES	YES
Railway-specific Criteria	CENELEC Certification	NO	PARTIAL	NO	NO	NO	NO	YES
	Integration into the CENELEC Process	MEDIUM	YES	MEDIUM	YES	YES	MEDIUM	YES

Fig. 8. Tool Evaluation Matrix (Excerpt)

the motivation for the assigned evaluation, and a Tool Evaluation Matrix, which summarises the evaluation for the tools. An excerpt of the matrix focussing on quality aspects is presented in Fig. 8 (the matrix is reported in its entirety in our deliverable). Overall, the majority of tools offer formal modelling and verification through model checking, and they generally offer simulation in textual or graphical form. Less frequent are features such as code generation, model-based testing, and traceability. With few exceptions, such as SCADE and Simulink, graphical user interfaces (GUIs) for these different tools are rather limited. Furthermore, in terms of learnability, the tools mainly require medium to advanced competences in formal methods, and, in the majority of the cases, require the support of an expert to be successfully used. This is in contrast with the demands of practitioners (Fig. 7), who primarily require tools that are easy to learn. It is also worth noticing that only SCADE is fully certified according to CENELEC.

7 Conclusion

The current paper reports ongoing results from WP4 of the ASTRail project. We presented a number of activities aimed at supporting the identification of the most suitable formal and semi-formal methods to be used for railway system development. Specifically, a SLR was conducted to categorise 114 scientific publications on formal methods and railways according to features such as the type of system and the phase of the development process addressed by the experi-

ence considered in the publication. The SLR was complemented with a projects review and a survey with practitioners, to identify the most mature formal and semi-formal methods and tools to be used in a railway context. This analysis has shown a dominance of the UML modelling language for high-level representation of system models, and a large variety of formal tools being used, with a dominance of the tools from the B family (ProB and Atelier B), followed by several other tools, including Simulink, NuSMV/nuXmv, Prover, SCADE, IBM Rational Software Architect, Polyspace, S3, SPIN, CPN Tools, etc. The projects review and the survey with practitioners confirmed this scattered landscape. As part of a tools review, tools supporting both modelling and formal verification were considered for accurate experimentation and evaluation. A set of 14 tools, considered to be the most promising, was carefully reviewed by means of a systematic evaluation based on a set of 34 evaluation features. The final product of these activities is a set of informative documents to support the ranking and selection of formal and semi-formal methods for railways, based on (a) the information retrieved from the literature, summarised in a Paper Analysis Matrix, (b) the information available from the tools evaluation, and (c) the Tool Evaluation Matrix, which allows practitioners to perform a fine-grained selection of the most appropriate formal methods and tools, suitable to their specific needs.

Based on the results presented in this paper, we are currently conducting the application phase of the project. In this phase, we first model the moving block distancing principles by means of 8 formal tools, namely Simulink, SCADE, NuSMV/nuXmv, SPIN, Atelier B, ProB, UPPAAL and UMC, selected based on the previous results. We then perform a usability evaluation of the tools together with railway practitioners. Finally, we further assess the applicability of the tools, involving our industrial partners in the modelling of automated driving principles.

Acknowledgements This work has been partially funded by the ASTRail project. This project received funding from the Shift2Rail Joint Undertaking under the European Union's Horizon 2020 research and innovation programme under grant agreement No 777561. The content of this paper reflects only the authors' view and the Shift2Rail Joint Undertaking is not responsible for any use that may be made of the included information.

References

1. Abrial, J.R.: Formal Methods: Theory Becoming Practice. J. Univers. Comput. Sci. **13**(5), 619–628 (2007). https://doi.org/10.3217/jucs-013-05-0619
2. Basile, D., ter Beek, M.H., Ciancia, V.: Statistical Model Checking of a Moving Block Railway Signalling Scenario with Uppaal SMC. In: Margaria, T., Steffen, B. (eds.) Proceedings of the 8th International Symposium on Leveraging Applications of Formal Methods, Verification and Validation — Verification (ISoLA 2018). LNCS, vol. 11245, pp. 372–391. Springer, Germany (2018). https://doi.org/10.1007/978-3-030-03421-4_24

3. Basile, D., ter Beek, M.H., Fantechi, A., Gnesi, S., Mazzanti, F., Piattino, A., Trentini, D., Ferrari, A.: On the Industrial Uptake of Formal Methods in the Railway Domain – A Survey with Stakeholders. In: Furia, C.A., Winter, K. (eds.) Proceedings of the 14th International Conference on Integrated Formal Methods (iFM 2018). LNCS, vol. 11023, pp. 20–29. Springer, Germany (2018). https://doi.org/10.1007/978-3-319-98938-9_2
4. ter Beek, M.H., Gnesi, S., Knapp, A.: Formal methods for transport systems. Int. J. Softw. Tools Technol. Transf. **20**(3), 237–241 (2018). https://doi.org/10.1007/s10009-018-0487-4
5. Berger, U., James, P., Lawrence, A., Roggenbach, M., Seisenberger, M.: Verification of the European Rail Traffic Management System in Real-Time Maude. Sci. Comput. Program. **154**, 61–88 (2018). https://doi.org/10.1016/j.scico.2017.10.011
6. Bjørner, D.: New Results and Trends in Formal Techniques and Tools for the Development of Software for Transportation Systems — A Review. In: Tarnai, G., Schnieder, E. (eds.) Proceedings of the 4th Symposium on Formal Methods for Railway Operation and Control Systems (FORMS 2003). L'Harmattan, Hungary (2003)
7. Bosschaart, M., Quaglietta, E., Janssen, B., Goverde, R.M.P.: Efficient formalization of railway interlocking data in RailML. Inf. Syst. **49**, 126–141 (2015). https://doi.org/10.1016/j.is.2014.11.007
8. Boulanger, J.L. (ed.): Formal Methods Applied to Industrial Complex Systems — Implementation of the B Method. John Wiley & Sons, USA (2014). https://doi.org/10.1002/9781119002727
9. Chiappini, A., Cimatti, A., Macchi, L., Rebollo, O., Roveri, M., Susi, A., Tonetta, S., Vittorini, B.: Formalization and Validation of a subset of the European Train Control System. In: Proceedings of the 32nd ACM/IEEE International Conference on Software Engineering (ICSE 2010). vol. 2, pp. 109–118. ACM, USA (2010). https://doi.org/10.1145/1810295.1810312
10. European Committee for Electrotechnical Standardization: CENELEC EN 50128 — Railway applications – Communication, signalling and processing systems – Software for railway control and protection systems (1 June 2011), https://standards.globalspec.com/std/1678027/cenelec-en-50128
11. Fantechi, A.: Twenty-Five Years of Formal Methods and Railways: What Next? In: Counsell, S., Núñez, M. (eds.) Software Engineering and Formal Methods — Revised Selected Papers of the SEFM 2013 Collocated Workshops: BEAT2, WS-FMDS, FM-RAIL-Bok, MoKMaSD, and OpenCert. LNCS, vol. 8368, pp. 167–183. Springer, Germany (2013). https://doi.org/10.1007/978-3-319-05032-4_13
12. Ferrari, A., Fantechi, A., Magnani, G., Grasso, D., Tempestini, M.: The Metrô Rio case study. Sci. Comput. Program. **78**(7), 828–842 (2013). https://doi.org/10.1016/j.scico.2012.04.003
13. Ferrari, A., ter Beek, M.H., Mazzanti, F., Basile, D., Fantechi, A., Gnesi, S., Piattino, A., Sturani, B., Trentini, D.: Survey on Formal Methods and Tools in Railways Technical Report on the activities performed within ASTRail, Deliverable D4.1. Tech. Rep. 396822, ISTI–CNR (2018). https://doi.org/10.5281/zenodo.2573921
14. Ferrari, A., Fantechi, A., Gnesi, S., Magnani, G.: Model-based development and formal methods in the railway industry. IEEE software **30**(3), 28–34 (2013). https://doi.org/10.1109/MS.2013.44
15. Flammini, F. (ed.): Railway Safety, Reliability, and Security: Technologies and Systems Engineering. IGI Global, USA (2012). https://doi.org/10.4018/978-1-4666-1643-1

16. Haxthausen, A.E., Peleska, J., Kinder, S.: A formal approach for the construction and verification of railway control systems. Formal Asp. Comput. **23**(2), 191–219 (2011). https://doi.org/10.1007/s00165-009-0143-6
17. Iliasov, A., Taylor, D., Laibinis, L., Romanovsky, A.B.: Formal Verification of Signalling Programs with SafeCap. In: Gallina, B., Skavhaug, A., Bitsch, F. (eds.) Proceedings of the 37th International Conference on Computer Safety, Reliability, and Security (SAFECOMP 2018). LNCS, vol. 11093, pp. 91–106. Springer, Germany (2018). https://doi.org/10.1007/978-3-319-99130-6_7
18. James, P., Moller, F., Nguyen, H.N., Roggenbach, M., Schneider, S., Treharne, H.: Techniques for modelling and verifying railway interlockings. Int. J. Softw. Tools Technol. Transf. **16**, 685–711 (2014). https://doi.org/10.1007/s10009-014-0304-7
19. Kitchenham, B.: Procedures for performing systematic reviews. Tech. Rep. TR/SE-0401, University of Keele, UK (July 2004), https://goo.gl/vYU8Fu
20. Lecomte, T., Déharbe, D., Prun, É., Mottin, E.: Applying a Formal Method in Industry: A 25-Year Trajectory. In: da Costa Cavalheiro, S.A., Fiadeiro, J.L. (eds.) Proceedings of the 20th Brazilian Symposium on Formal Methods: Foundations and Applications (SBMF 2017). LNCS, vol. 10623, pp. 70–87. Springer, Germany (2017). https://doi.org/10.1007/978-3-319-70848-5_6
21. Leuschel, M., Falampin, J., Fritz, F., Plagge, D.: Automated property verification for large scale B models with ProB. Formal Asp. Comput. **23**(6), 683–709 (2011). https://doi.org/10.1007/s00165-010-0172-1
22. Mazzanti, F., Ferrari, A.: Ten Diverse Formal Models for a CBTC Automatic Train Supervision System. In: Gallagher, J.P., van Glabbeek, R., Serwe, W. (eds.) Proceedings of the 3rd Workshop on Models for Formal Analysis of Real Systems and the 6th International Workshop on Verification and Program Transformation (MARS/VPT 2018). EPTCS, vol. 268, pp. 104–149 (2018). https://doi.org/10.4204/EPTCS.268.4
23. Mazzanti, F., Ferrari, A., Spagnolo, G.O.: Towards formal methods diversity in railways: an experience report with seven frameworks. Int. J. Softw. Tools Technol. Transf. **20**(3), 263–288 (2018). https://doi.org/10.1007/s10009-018-0488-3
24. Mazzanti, F., Spagnolo, G.O., Longa, S.D., Ferrari, A.: Deadlock Avoidance in Train Scheduling: A Model Checking Approach. In: Lang, F., Flammini, F. (eds.) Proceedings of the 19th International Conference on Formal Methods for Industrial Critical Systems (FMICS 2014). LNCS, vol. 8718, pp. 109–123. Springer, Germany (2014). https://doi.org/10.1007/978-3-319-10702-8
25. Moller, F., Nguyen, H.N., Roggenbach, M., Schneider, S., Treharne, H.: Defining and Model Checking Abstractions of Complex Railway Models Using CSP‖B. In: Biere, A., Nahir, A., Vos, T. (eds.) Hardware and Software: Verification and Testing — Revised Selected Papers of the 8th International Haifa Verification Conference (HVC 2012). LNCS, vol. 7857, pp. 193–208. Springer, Germany (2013). https://doi.org/10.1007/978-3-642-39611-3_20
26. Rispoli, F., Castorina, M., Neri, A., Filip, A., Di Mambro, G., Senesi, F.: Recent progress in application of GNSS and advanced communications for railway signaling. In: Proceedings of the 23rd International Conference Radioelektronika (RADIOELEKTRONIKA 2013). pp. 13–22. IEEE (2013). https://doi.org/10.1109/RadioElek.2013.6530882
27. Vanit-Anunchai, S.: Modelling and simulating a Thai railway signalling system using Coloured Petri Nets. Int. J. Softw. Tools Technol. Transf. **20**(3), 243–262 (2018). https://doi.org/10.1007/s10009-018-0482-9

28. Vu, L.H., Haxthausen, A.E., Peleska, J.: Formal modelling and verification of interlocking systems featuring sequential release. Sci. Comput. Program. **133**, 91–115 (2017). https://doi.org/10.1016/j.scico.2016.05.010
29. Winter, K., Robinson, N.J.: Modelling large railway interlockings and model checking small ones. In: Oudshoorn, M.J. (ed.) Proceedings of the 26th Australasian Computer Science Conference (ACSC 2003). Conferences in Research and Practice in Information Technology, vol. 16, pp. 309–316. Australian Computer Society, Australia (2003), http://crpit.com/confpapers/CRPITV16Winter.pdf
30. Woodcock, J., Larsen, P.G., Bicarregui, J., Fitzgerald, J.S.: Formal methods: Practice and experience. ACM Comput. Surv. **41**(4), 19:1–19:36 (2009). https://doi.org/10.1145/1592434.1592436