Theories, Techniques and Tools for Engineering Heterogeneous Railway Networks

Paulius Stankaitis(✉) and Alexei Iliasov

School of Computing Science, Newcastle University, Newcastle upon Tyne, UK
{p.stankaitis,alexei.iliasov}@ncl.ac.uk

Abstract. Modernising outdated national railway systems will be done gradually due to practical constraints thus creating network areas with different signalling systems. Formal methods have been successfully applied in the railway domain for years. Yet the latest railway challenges such as heterogeneous railway signalling will require novel modelling techniques and adequate verification tools support. In this research we aim to develop new theories, techniques and tools for modelling and verification of complex networks comprising areas with a mixed signalling. This student paper discusses the research problem, related work and presents the ongoing work.

Keywords: Distributed railway interlocking · Hybrid systems · Event-B method

1 Introduction

In the last few decades railway domain has proved to be a fruitful area for applying various formal methods. Yet the latest railway challenges will require novel modelling techniques and adequate formal verification tools support. Integrating modern railway signalling systems within an outdated national railway networks is currently one of the major challenges. Indeed a gradual railway modernisation process means that heterogeneous railway signalling networks will be inevitable due to practical constraints. In some situations mainline services must be integrated with urban networks which simply require different signalling solutions as high service availability can only be achieved with a moving block signalling solution[1]. To give an example Crossrail is a major ongoing railway project where mainline services will be integrated with a high performance urban railway system. This particular network will operate with three different signalling systems. In western and eastern branches of the network fixed block signalling systems will operate whereas the central area will be operated with a moving block principle. Novel signalling interfaces will be developed to ensure a smooth and safe rolling stock signalling transition. In short this PhD study aims to address the challenge of modelling and verification of railway networks with different signalling systems.

[1] To this date a moving block signalling solution only operates in urban networks.

A. Fantechi et al. (Eds.): RSSRail 2017, LNCS 10598, pp. 241–250, 2017.
https://doi.org/10.1007/978-3-319-68499-4_16

The following section overviews key difficulties in formally modelling and verifying such systems which are in fact cyber-physical systems. Secondly we discuss more notable related work examples and present technical contributions this research aims to achieve. Last two sections discuss the current work on modelling and verification of a distributed railway network in the Event-B language and future research objectives.

2 Formal Methods in Railway Domain

Formal methods - a mathematical model driven methods provide a systematic approach for developing complex systems. They offer an approach to specify systems precisely via a mathematically defined syntax and semantics as well as formally validate them by using semi-automatic or automatic techniques. At the moment among the biggest challenges in the field is ensuring safety and correctness of cyber-physical systems.

For years formal methods have been successfully applied to the railway domain however yet a considerably little work has been done in including a cyber-physical nature of railway for a safety reasoning. Established railway operation principles did not require that so formal methods mainly focused on a static railway data verification - control table verification. However modern signalling systems were developed to reduce an overdesign and hence increase the capacity of railway networks. Railway operational principles have been rapidly moving towards a continuous agent communication and a more dynamic parameter (e.g. permitted speed profile) computation which are indeed two essential aspects of cyber-physical systems - communication and computation. Therefore to model and reason about safety of a modern signalling system we believe it is paramount to consider a cyber-physical nature of railway.

In general cyber-physical systems [30] have tight integration of communication, computation and control aspects and include discrete as well as continuous behaviours. Indeed the difficulty in modelling and verifying cyber-physical systems is a necessity to consider all these aspects together. To this date there exists no formal framework which could capture a tight integration of these systems aspects [18]. Furthermore for a lot of safety-critical system the dynamic nature of an environment has to be considered in the model as well. For instance a lossy communication aspect is particularly important when modelling modern-radio based railway signalling systems or railway systems with signalling transitions. Hybrid systems formal verification challenges arise mainly due to continuous variables with non-linear dynamics [3,31]. An algorithmic verification of hybrid systems with available model checking tools is limited even under severe restrictions whereas simulation tools coverage is not adequate for a safety reasoning. In spite of that system validation through simulation is still the most prevalent method used by railway industry today. Alternative methods such as a deductive verification method are not limited by the state space and combined with computer algebra systems can deal with non-linear dynamics though some problems for an automated deductive verification still have to be resolved [4].

Related work. Over the years formal methods were primarily applied only for a discrete safety reasoning of the railway systems. The literature review revealed that only a small fraction of all railway oriented research considered a cyber-physical nature of railway systems. The following paragraphs discuss a more notable related work on distributed dynamic railway systems which are a class of cyber-physical systems.

To authors knowledge the earliest attempt to formally analyse distributed railway solid-state interlocking systems was completed by Morley [23]. In this interesting work author developed a formal model of a protocol for a cross boundary route locking and releasing mechanism. By analysing temporal properties of the model he discovered that in certain scenarios safety properties can be violated. Few years later a paper by Cimatti et al. [10] presented an industrially driven formal methods study where authors formally modelled a communication protocol for safety-critical distributed systems including distributed railway interlocking systems. Their method used Statecharts diagrams to specify high level protocol properties and the OBJECTGEODE model checker for the protocol validation. In other work a different concept of distributed railway control system was introduced by Haxthausen and Peleska [14]. Their presented engineering concept of the control system relied on a radio based communication and switch boxes - systems which can only control a single railway point. Authors formally modelled the system with the RAISE [13] specification language which allowed to develop a formal model incrementally using a refinement process and prove refinement and safety properties with available justification tools. The timing properties of the design were considered in the extended work [22]. Similar ideas for distributed railway interlocking system were also presented in [8,15] where authors used Statecharts and Petri Nets to model and verify decentralised railway interlocking.

At the same time André Platzer introduced an alternative approach to exploring a state-space with model checkers in verifying systems safety. A developed formalism and logic for reasoning about hybrid systems uses a deductive verification and can be implemented in a KeyMaera X verification tool [24,26]. The later work presented a case study where differential dynamic logic was applied for a safety verification of the European Train Control System [27]. Differential Dynamic Logic was also used to model and verify a handover protocol between two trackside train control systems (radio-block centres) by Liu et al. [21]. In a work by Cimatti et al. [11] authors proposed a different logic based on the temporal logic with regular expressions. Their motivation was driven by a need of the automatic verification method for verifying hybrid requirements for hybrid railway system. A more recent work by Iliasov et al. [17] proposed a domain specific language - Unified Train Driving Policy. The formal notation allows to express both static and dynamic properties of railway in readable syntax which can be interpreted by railway engineers without prior knowledge of formal methods. A few recent formal methods projects on cyber-physical systems applied their novel techniques for modelling and verification of hybrid railway systems [16,28,29].

In the previous project on modelling and verification of railway interlocking systems we discussed possible future PhD study directions for addressing the safety of heterogeneous railway networks [32]. The two year project focused on developing an expressive railway oriented simulator which would enable modelling and analysing complex railway including railway systems with mixed signalling. In the future we plan to use the system-level simulator as a specification front-end for our modelling framework discussed in the following paragraph.

This PhD research aims to focus on theories and techniques for formal modelling and verification of classes of distributed hybrid railway systems which are in fact what we define as heterogeneous railway networks. In particular we are interested in developing a railway oriented formal modelling framework which could capture dynamic distributed hybrid systems. A similar work [12,20] has been completed for more general cooperating agent based systems by exploring design patterns or more focused on dynamic distributed hybrid systems in [25]. In our work we would like to continue in this direction but by restricting our methodology to the railway domain. First of all to develop such a formal framework to reason about distributed hybrid railway networks one needs to understand and formally define a general railway design structure. The formal framework should not only capture existing railway operation principles for which a number of domain-specific languages already exists but also allow modelling moving block signalling systems. In the previous paragraphs we also emphasised the necessity to consider a cyber-physical nature of railway for safety reasoning. Therefore an important requirement for the modelling formal framework is to allow capturing continuous evolution of agents and for that we can use existing approaches for instance hybrid automata. The modelling notation should not only have executional semantics which is exactly the simulation of railway operation but it also should offer proof semantics. The work by Damm et al. [12] proposed a generic proof-rules for reducing the complexity of the reasoning about collision avoidance systems. In this PhD research we will attempt to further improve this approach by specifically addressing the railway domain. To enable reasoning about safety of heterogeneous railway signalling we will need to include new safety rules for a system transition reasoning - a similar but more generic to presented in [21]. Lastly in order to ensure that results have potential to be useful in the industrial setting this research will be conducted in a close cooperation with Siemens Rail Automation.

In the following section we present an ongoing work which aims to develop a generic design pattern for distributed railway networks. For that we use the Event-B modelling language as a back-end formal notation which offers a refinement based modelling language. It allows to start with an abstract model for instance the skeleton of a dynamic distributed railway system and then include new details through a number of correctness preserving refinement steps for instance details of a specific signalling system. In this paper we will not discuss hybrid modelling part of the framework but we will base our work on existing methods developed for Event-B [5,7].

3 Distributed Formal Railway Model in Event-B

The Event-B mathematical language used in the system development and analysis is an evolution of the classical B method [1] and Action Systems [6]. Perhaps due to the success of the B method and a good tool support Event-B has also been a popular language choice for modelling railway systems [2,9,19]. The formal specification language offers a fairly high-level mathematical language based on a first-order logic and Zermelo-Fraenkel set theory as well as an economical yet expressive modelling notation. The formalism belongs to a family of state-based modelling languages where a state of a discrete system is simply a collection of variables and constants whereas the transition is a guarded variable transformation.

A cornerstone of the Event-B method is the step-wise development that facilitates a gradual design of a system implementation through a number of correctness preserving refinement steps. The model development starts with a creation of a very abstract specification and the model is completed when all requirements and specifications are covered. The Event-B model is made of two key components - machines and contexts which respectively describe dynamic and static parts of the system. The context contains modeller declared constants and associated axioms which can be made visible in machines. The dynamic part of the model contains variables which are constrained by invariants and initialised by an action. The state variables are then transformed by actions which are part of events and the modeller may use predicate guards to denote when event is triggered (see Fig. 1). Specifying a model is not sufficient one must provide evidence about the correctness of the model as well. The Event-B method is a proof driven specification language where model correctness is demonstrated by generating and discharging proof obligations - theorems in the first-order logic. The model is considered to be correct when all proof obligations are discharged.

The following subsections present an ongoing work on modelling a distributed railway interlocking. In particular we focus on modelling the distributed resource allocation problem where processes can capture and release available resources as it is paramount for a distributed railway interlocking. To develop a protocol for a safe distributed route locking mechanism in further refinements undischarged proof obligations will be used.

```
machine M
  sees Context
  variables v
  invariant I(c, s, v)
  initialisation R(c, s, v′)
  events
    E1 = any vl where g(c, s, vl, v) then S(c, s, vl, v, v′) end
    ...
end
```

Fig. 1. Event-B machine structure.

3.1 Abstract Distributed Railway Interlocking Model

First of all we describe the modelling and refinement plan of a distributed railway signalling with main requirements at each step. The initial abstract model specifies the general concept of a distributed resource allocation protocol - processes capturing and releasing available resources. Starting with such a mathematical abstraction allows to simplify the development of a protocol without considering complicated railway requirements at early modelling stages.

Initial model. An abstract model of processes capturing resources.

1. An abstract model context - processes and resources (finite sets).
2. An abstract model contains events for capturing and releasing resources.
3. Processes can only capture not already captured resources.
4. Processes can only release their captured resources.
5. Processes could capture more than a single resource at a time.
6. No two or more processes can have same resources captured.

Refinement 1. Extending the model with events for requesting and granting resources and solving a contention problem.

1. Introducing events for requesting and granting resources.
2. Introducing events for detecting and solving the contention problem.
3. Resources can only be captured if requested and granted by the process.
4. Same resources can be requested by multiple processes at the same time.
5. Resources request from a single process cannot be partially granted.
6. Processes can request any set of resources.
7. Resources can be granted to the process if they have not been requested, granted or captured by other processes or if the conflict has been solved with detect/solve events.

Other Refinements. Introducing graph based resource structures and railway related context (not discussed in this paper).

1. Distributing resources in to separate zones with associated controllers.
2. Introducing a graph based resource structure in to the model.
3. Introducing a railway related context and route locking principles.

Other properties such as a system progress can be addressed by assuming that processes release resources eventually and also by introducing a resource granting queue. In the initial model we only impose a single railway related safety rule which states that collision freedom is ensured if no two or more trains share the same route. This can be simply expressed by the invariant - no two or more processes can have same resources captured.

The modelling was started by creating the abstract context with two carrier sets for processes and resources with two associated axioms stating that these sets are finite. In the dynamic part of the model we defined a global variable *mrk* (marked) for mapping resources to processes. Furthermore we introduced

two events for capturing and releasing resources which are in fact abstract representations of a railway route locking and releasing operations. Both events have similar guards except one can only release resources if they have already been captured.

Event *capture* $\widehat{=}$
any $r,\ p,\ pr$
where
grd1 : $r \not\subseteq dom(mrk)$
grd2 : $p \in P$
grd3 : $pr \in r \rightarrow \{p\}$
grd4 : $\varnothing \subset pr$
then
act1 : $mrk := mrk \cup pr$
end

Event *release* $\widehat{=}$
any $r,\ p,\ pr$
where
grd1 : $r \subseteq dom(mrk)$
grd2 : $p \in ran(mrk)$
grd3 : $pr \in r \rightarrow \{p\}$
grd4 : $\varnothing \subset pr$
then
act1 : $mrk := mrk \setminus pr$
end

In the next model refinement a logical step then was to introduce two new events for requesting and granting resources and two buffers for storing resourced requests (req) and granted resources (ack). A process can request any subset of resources and grant event then checks whether those resources are not captured or granted for other processes. Because of new events we also needed to update the abstract capture event with stronger guards and additional action to update both buffers.

Event *send_request* $\widehat{=}$
any $r,\ p,\ pr$
where
grd1 : $p \in P$
grd2 : $r \subseteq R$
grd3 : $pr \in r \rightarrow \{p\}$
grd4 : $\varnothing \subset pr$
then
act1 : $req := req \cup pr$
end

Event *grant_request* $\widehat{=}$
any p
where
grd1 : $p \in ran(req)$
grd2 : $req^{-1}[\{p\}] \cap dom(mrk \rhd \{p\}) = \varnothing$
grd3 : $req^{-1}[\{p\}] \cap dom(req \rhd \{p\}) = \varnothing$
grd4 : $req^{-1}[\{p\}] \cap dom(ack \rhd \{p\}) = \varnothing$
then
act1 : $ack := ack \cup (req \rhd \{p\})$
act2 : $req := req \rhd \{p\}$
end

The request buffer may contain multiple requests for the same resources from different processes. So the resource grant event will only grant a set of resources to a single process if they have not been requested by other process. In case of multiple requests for the same resources from different processes we needed to introduce another two events for detecting and solving such a situation discussed in the following subsection.

3.2 Contention Problem for Distributed Railway Interlocking

A very common problem in developing distributed systems is the contention problem. In our model the problem can arise when a number of same resources have been requested by different processes. Since we do not allow partial resource

allocation because of the safety principle which comes from the railway domain the system deadlocks. To resolve this we simply introduced two new events for detecting and solving this problem. The contention detection event is enabled when there exists a set of processes which all requested common resources and if those resources have not been captured yet. This event action simply copies the set of interested requests to another buffer - cnt (contention).

Event *detect_contention* $\widehat{=}$
  any
    p
  where
    grd1 : $p = \{x | \exists y \cdot y \neq x \wedge req^{-1}[\{x\}] \cap req^{-1}[\{y\}] \neq \varnothing\}$
    grd2 : $p = \{x | req^{-1}[\{x\}] \cap dom(mrk \rhd \{x\}) = \varnothing\}$
    grd4 : $p \neq \varnothing$
  then
    act1 : $cnt := cnt \cup (req \rhd p)$
  end

After that the following event grant (not shown here) nondeterministically selects a process from that buffer and grants resources for that single process and also removes its requests from the request buffer. The detect/solve process then can be repeated for remaining processes. At this level one does not need to consider which process is given a priority this becomes more important when graph based resource structure is introduced.

4 Conclusions and Future Work

In this paper we presented the main motivation of this PhD research which is the need of new formal methods techniques for modelling distributed dynamic railway networks and reasoning about their safety. The research proposed to develop a new railway oriented modelling framework with proof rules which could capture a cyber-physical nature of the heterogeneous railway networks. Then we presented an ongoing work on modelling a distributed railway signalling system which is necessary in order to explore common distributed railway design patterns and also deduce invariants for heterogeneous railway networks. In the following months we aim to complete this model and focus on hybrid framework part for modelling and reasoning about heterogeneous railway networks.

Acknowledgements. This work is supported by an iCASE studentship (EPSRC and Siemens Rail Automation). We are grateful to our colleagues from Siemens Rail Automation for invaluable feedback. We would also like to thank Guillaume Babin and Yamine Aït-Ameur for useful conversations.

References

1. Abrial, J.-R.: The B-book: Assigning Programs to Meanings. Cambridge University Press, New York (1996)
2. Abrial, J.-R.: Modeling in Event-B: System and Software Engineering. Cambridge University Press, New York (2013)
3. Alur, R.: Formal verification of hybrid systems. In: Proceedings of the Ninth ACM International Conference on Embedded Software, EMSOFT 2011, pp. 273–278. ACM, New York (2011)
4. Alur, R., Courcoubetis, C., Halbwachs, N., Henzinger, T.A., Ho, P.-H., Nicollin, X., Olivero, A., Sifakis, J., Yovine, S.: The algorithmic analysis of hybrid systems. Theor. Comput. Sci. **138**(1), 3–34 (1995)
5. Babin, G., Aït-Ameur, Y., Nakajima, S., Pantel, M.: Refinement and proof based development of systems characterized by continuous functions. In: Li, X., Liu, Z., Yi, W. (eds.) SETTA 2015. LNCS, vol. 9409, pp. 55–70. Springer, Cham (2015). doi:10.1007/978-3-319-25942-0_4
6. Back, R.J.R.: Refinement calculus, part II: parallel and reactive programs. In: Bakker, J.W., Roever, W.-P., Rozenberg, G. (eds.) REX 1989. LNCS, vol. 430, pp. 67–93. Springer, Heidelberg (1990). doi:10.1007/3-540-52559-9_61
7. Banach, R., Butler, M., Qin, S., Verma, N., Zhu, H.: Core hybrid Event-B I: single hybrid event-B machines. Sci. Comput. Program. **105**, 92–123 (2015)
8. Banci, M., Fantechi, A., Gnesi, S.: The role of formal methods in developing a distributed railway interlocking system. In: Proceedings of the 5th Symposium on Formal Methods for Automation and Safety in Railway and Automotive Systems (FORMS/FORMAT 2004), pp. 220–230 (2004)
9. Butler, M.: A system-based approach to the formal development of embedded controllers for a railway. Des. Autom. Embed. Syst. **6**(4), 355–366 (2002)
10. Cimatti, A., Pieraccini, P.L., Sebastiani, R., Traverso, P., Villafiorita, A.: Formal specification and validation of a vital communication protocol. In: Wing, J.M., Woodcock, J., Davies, J. (eds.) FM 1999. LNCS, vol. 1709, pp. 1584–1604. Springer, Heidelberg (1999). doi:10.1007/3-540-48118-4_34
11. Cimatti, A., Roveri, M., Tonetta, S.: Requirements validation for hybrid systems. In: Bouajjani, A., Maler, O. (eds.) CAV 2009. LNCS, vol. 5643, pp. 188–203. Springer, Heidelberg (2009). doi:10.1007/978-3-642-02658-4_17
12. Damm, W., Hungar, H., Olderog, E.R.: Verification of cooperating traffic agents. Int. J. Control **79**(5), 395–421 (2006)
13. George, C., Haxthausen, A.E., Hughes, S., Milne, R., Prehn, S., Pedersen, J.S.: The RAISE Development Method. Prentice Hall International (1995)
14. Haxthausen, A.E., Peleska, J.: Formal development and verification of a distributed railway control system. IEEE Trans. Software Eng. **26**(8), 687–701 (2000)
15. Hei, X., Takahashi, S., Hideo, N.: Toward developing a decentralized railway signalling system using petri nets. In: Proceedings of the IEEE Conference on Robotics, Automation and Mechatronics, pp. 851–855 (2008)
16. Hermanns, H., Jansen, D.N., Usenko, Y.S.: A comparative reliability analysis of ETCS train radio communications. Reports of SFB/TR 14 AVACS 2, SFB/TR 14 AVACS, February 2005. ISSN: 1860-9821. http://www.avacs.org
17. Iliasov, A., Lopatkin, I., Romanovsky, A.: Unified Train Driving Policy, pp. 447–474. Wiley (2014)
18. Kim, K.D., Kumar, P.R.: Cyber-physical systems: a perspective at the centennial. Proc. IEEE **100**(Special Centennial Issue), 1287–1308 (2012)

19. Kiss, T., Jánosi-Rancz, K.T.: Developing railway interlocking systems with session types and Event-B. In: Proceedings of the IEEE 11th International Symposium on Applied Computational Intelligence and Informatics (SACI), pp. 93–98, May 2016
20. Knudsen, J., Ravn, A.P., Skou, A.: Design verification patterns. In: Jones, C.B., Liu, Z., Woodcock, J. (eds.) Formal Methods and Hybrid Real-Time Systems. LNCS, vol. 4700, pp. 399–413. Springer, Heidelberg (2007). doi:10.1007/978-3-540-75221-9_18
21. Liu, Y., Tang, T., Liu, J., Zhao, L., Xu, T.: Formal modeling and verification of RBC handover of ETCS using differential dynamic logic. In: Proceedings of the International Symposium on the Autonomous Decentralized Systems (ISADS), pp. 67–72. IEEE (2011)
22. Madsen, M.S., Bæk, M.M.: Modelling a distributed railway control system. Master's thesis, Technical University of Denmark, DTU, DK-2800 Kgs, Lyngby, Denmark (2005)
23. Morley, M.J.: Safety assurance in interlocking design. PhD thesis (1996)
24. Platzer, A.: Differential dynamic logic for hybrid systems. J. Autom. Reason. **41**(2), 143–189 (2008)
25. Platzer, A.: Quantified differential dynamic logic for distributed hybrid systems. In: Dawar, A., Veith, H. (eds.) CSL 2010. LNCS, vol. 6247, pp. 469–483. Springer, Heidelberg (2010). doi:10.1007/978-3-642-15205-4_36
26. Platzer, A., Quesel, J.-D.: KeYmaera: a hybrid theorem prover for hybrid systems (system description). In: Armando, A., Baumgartner, P., Dowek, G. (eds.) IJCAR 2008. LNCS, vol. 5195, pp. 171–178. Springer, Heidelberg (2008). doi:10.1007/978-3-540-71070-7_15
27. Platzer, A., Quesel, J.-D.: European train control system: a case study in formal verification. In: Breitman, K., Cavalcanti, A. (eds.) ICFEM 2009. LNCS, vol. 5885, pp. 246–265. Springer, Heidelberg (2009). doi:10.1007/978-3-642-10373-5_13
28. ADVANCE project: Final report on application on railway domai, deliverable d1.4 workpackage 1. Technical report, 30 November 2014
29. INTO-CPS project: Case studies 2, deliverable d1.2. Technical report, November 2016
30. Sha, L., Gopalakrishnan, S., Liu, X., Wang, Q.: Cyber-physical systems: a new frontier. In: Proceedings of the IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing, SUTC 2008, pp. 1–9, June 2008
31. Silva, B.I., Stursberg, O., Krogh, B.H., Engell, S.: An assessment of the current status of algorithmic approaches to the verification of hybrid systems. In: Proceedings of the 40th IEEE Conference on Decision and Control, vol. 3, pp. 2867–2874. IEEE (2001)
32. Stankaitis, P., Iliasov, A.: Safety verification of heterogeneous railway networks. In: Lecomte, T., Pinger, R., Romanovsky, A. (eds.) RSSRail 2016. LNCS, vol. 9707, pp. 150–159. Springer, Cham (2016). doi:10.1007/978-3-319-33951-1_11