




# Supporting Railway Innovations with Formal Modelling and Verification

Bas Luttik<sup>(✉)</sup> 

Eindhoven University of Technology, Eindhoven, The Netherlands  
s.p.luttik@tue.nl

It is a continuing challenge for European railway infrastructure managers to increase the capacity of the dense European railway network and to achieve cost reductions at the same time. Innovations developed to that effect rely heavily on digital technology. To cope with the ensued complexity, railway infrastructure managers are starting to appreciate more and more the use of formal modelling and verification techniques to support the development of these digital innovations. In my presentation I will discuss our contributions to two ongoing innovations in the railway domain: EULYNX and ERTMS/ETCS Hybrid Level 3.

## EULYNX

The goal of the EULYNX<sup>1</sup> undertaking is to develop digital standardised interfaces between interlockings and trackside equipment (signals, points, level crossings, etc.). It is crucial that the standard is unambiguous, that it ensures all relevant safety requirements, and that compliance to the standard can be tested thoroughly. To this end, the FormaSig project<sup>2</sup>—a collaboration between railway infrastructure managers DB Netz and ProRail, Eindhoven University of Technology and the University of Twente—supports EULYNX with formal verification and model-based test technology.

The EULYNX standardised interfaces are defined using SysML internal block diagrams and state machine diagrams. The approach of the FormaSig project is to derive from these SysML models a formal model in the process specification language mCRL2 [10]. The mCRL2 toolset<sup>3</sup> then offers model-checking facilities to formally analyse the correctness of the interface model with respect to high-level requirements [6]. Moreover, since the semantics of an mCRL2 model is a labelled transition system, it also facilitates automated model-based testing of compliance of implementations to the standard in accordance with formal testing theory [13].

In a first case study, we have manually derived an mCRL2 model from the SysML models specifying the EULYNX Point interface [4]. A formal analysis of

---

<sup>1</sup> <https://www.eulynx.eu>.

<sup>2</sup> <https://fsa.win.tue.nl/formasig/>.

<sup>3</sup> <https://www.mcr2.org>.

the model using the mCRL2 toolset revealed a deadlock caused by event buffers overflowing and a discrepancy in the interaction of the EULYNX standard with the underlying communication protocol. We also performed some preliminary model-based testing experiments using JTorX [2] to automatically generate tests from the mCRL2 model, running those tests on a simulator of the EULYNX interface. The case study showed the feasibility of our approach.

Our next step was to automate the translation of EULYNX SysML models to mCRL2. The precise semantic interpretation of the SysML models developed in EULYNX, however, is not fixed. To achieve maximal flexibility in our analysis, we have therefore set up the translation from SysML to mCRL2 such that it can be easily modified. At its core is a generic formalisation of the semantics of SysML state machines in the expressive mCRL2 language [3]. The automated translation interprets the SysML internal block diagrams, and renders the SysML model as a data object within the mCRL2 specification of SysML state-machine semantics. The translation framework, with an application to the EULYNX Point interface, is described in [5].

We are currently using the framework to analyse other EULYNX interfaces (level crossing, light signal, train detection). We observe that these other interfaces yield mCRL2 models with significantly larger state spaces. So we are investigating how we can use compositional state-space generation techniques [12] and symbolic model checking [11] recently developed for mCRL2. Also, we are experimenting with alternative semantic interpretations of the SysML models; the flexible set-up of the translation framework now pays off, because it allows us to experiment with variations of the state-machine semantics without changing the translation tool itself.

## ERTMS/ETCS HL3

Level 3 of ERTMS/ETCS<sup>4</sup>, the European standardised command and signalling system, introduces the concept of *virtual block*. Trains communicate their exact positions on the track to the trackside system through a radio connection, and the system computes movement authorities for trains ensuring that two trains never simultaneously occupy the same virtual block. This approach obviates the need for expensive train detection hardware. Moreover, since virtual blocks can be arbitrarily small, or even move along with the train, a capacity increase of the network is realised.

Transitioning to such a radically new train separation concept on the dense European railway network is an enormous challenge, because it requires the entire railway network and trains (passenger and freight) to be equipped with the enabling technology. To smoothen the transition, railway infrastructure managers are developing a hybrid version of ERTMS/ETCS Level 3 (HL3). It describes a train separation mechanism based on virtual blocks that is integrated with a traditional train detection system with train detection hardware.

---

<sup>4</sup> <https://ertms.be/workgroups/level3>.

The HL3 principles facilitate a partitioning of hardware protected track sections into so-called virtual subsections. Multiple suitably equipped trains can then be admitted on the same track section simultaneously, ensuring that they are never simultaneously occupying the same virtual subsection. For trains without the required equipment, the system still provides the traditional train separation mechanism. An added benefit of HL3 is that, by making use of the installed train detection hardware, it can recover from a failing radio communication between train and trackside.

There has been ample attention for the HL3 principles from the formal methods research community since version 1A of the principles [8] served as the ABZ 2018 case study (see [7] and references therein). At FMICS 2018 we reported on a formal analysis of the principles using mCRL2 [1]. That first version of our mCRL2 model formalised the core the principles; it ignored the influence of various timers that should prevent the system from qualifying a situation as hazardous too quickly. Since our presentation at FMICS 2018, we have updated the mCRL2 model to reflect version 1D of the principles [9], and also incorporated the behaviour of the timers. Our various analyses exposed potentially dangerous scenarios, especially also related to the behaviour of timers, and resulted in recommendations for improvement of the HL3 principles that were taken into account in subsequent versions. ProRail is using our mCRL2 model to simulate HL3 scenarios.

**Acknowledgements.** The contributions to EULYNX have been made in collaboration with Mark Bouwman from Eindhoven University of Technology and Arend Rensink, Mariëlle Stoelinga and Djurre van der Wal from the University of Twente; the research was funded by ProRail and DB Netz. The contributions to ERTMS/ETCS Hybrid Level 3 have been made in collaboration with Maarten Bartholomeus from ProRail and Rick Erkens and Tim Willemse from Eindhoven University of Technology; the research was partially funded by ProRail. The vision presented here does not necessarily reflect the strategy of DB Netz or ProRail.

## References

1. Bartholomeus, M., Luttik, B., Willemse, T.: Modelling and analysing ERTMS hybrid level 3 with the mCRL2 toolset. In: Howar, F., Barnat, J. (eds.) FMICS 2018. LNCS, vol. 11119, pp. 98–114. Springer, Cham (2018). [https://doi.org/10.1007/978-3-030-00244-2\\_7](https://doi.org/10.1007/978-3-030-00244-2_7)
2. Belinfante, A.: JTorX: a tool for on-line model-driven test derivation and execution. In: Esparza, J., Majumdar, R. (eds.) TACAS 2010. LNCS, vol. 6015, pp. 266–270. Springer, Heidelberg (2010). [https://doi.org/10.1007/978-3-642-12002-2\\_21](https://doi.org/10.1007/978-3-642-12002-2_21)
3. Bouwman, M., Luttik, B., van der Wal, D.: A formalisation of SysML state machines in mCRL2. In: Peters, K., Willemse, T.A.C. (eds.) FORTE 2021. LNCS, vol. 12719, pp. 42–59. Springer, Cham (2021). [https://doi.org/10.1007/978-3-030-78089-0\\_3](https://doi.org/10.1007/978-3-030-78089-0_3)

4. Bouwman, M., van der Wal, D., Luttik, B., Stoelinga, M., Rensink, A.: What is the point: formal analysis and test generation for a railway standard. In: Baraldi, P., di Maio, F., Zio, E. (eds.) *Proceedings of ESREL 2020 and PSAM 15*. Research Publishing, Singapore (2020). <http://www.rpsonline.com.sg/proceedings/esrel2020/html/4410.xml>
5. Bouwman, M., van der Wal, D., Luttik, B., Stoelinga, M., Rensink, A.: A case in point: verification and testing of a EULYNX interface. *Formal Aspects Comput.* (2022). <https://doi.org/10.1145/3528207>
6. Bunte, O., et al.: The mCRL2 toolset for analysing concurrent systems. In: Vojnar, T., Zhang, L. (eds.) *TACAS 2019, Part II*. LNCS, vol. 11428, pp. 21–39. Springer, Cham (2019). [https://doi.org/10.1007/978-3-030-17465-1\\_2](https://doi.org/10.1007/978-3-030-17465-1_2)
7. Butler, M.J., Hoang, T.S., Raschke, A., Reichl, K.: Introduction to special section on the ABZ 2018 case study: Hybrid ERTMS/ETCS level 3. *Int. J. Softw. Tools Technol. Transf.* **22**(3), 249–255 (2020)
8. EEIG ERTMS Users Group: Hybrid ERTMS/ETCS Level 3, ref: 16E045, Version: 1A, Date: 14/07/2017
9. EEIG ERTMS Users Group: Hybrid ERTMS/ETCS Level 3, ref: 16E042, Version: 1D, Date: 15/10/2020
10. Groote, J.F., Mousavi, M.R.: *Modeling and Analysis of Communicating Systems*. MIT Press, Cambridge (2014). <http://mitpress.mit.edu/books/modeling-and-analysis-communicating-systems>
11. Laveaux, M., Wesselink, W., Willemse, T.A.C.: On-the-fly solving for symbolic parity games. In: Fisman, D., Rosu, G. (eds.) *TACAS 2022*. LNCS, vol. 13244, pp. 137–155. Springer, Cham (2022). [https://doi.org/10.1007/978-3-030-99527-0\\_8](https://doi.org/10.1007/978-3-030-99527-0_8)
12. Laveaux, M., Willemse, T.A.C.: Decomposing monolithic processes in a process algebra with multi-actions. In: Lange, J., Mavridou, A., Safina, L., Scalas, A. (eds.) *Proceedings 14th Interaction and Concurrency Experience, ICE 2021, 18 June 2021*. EPTCS, vol. 347, pp. 57–76 (2021). <https://doi.org/10.4204/EPTCS.347.4>
13. Tretmans, J.: Model based testing with labelled transition systems. In: Hierons, R.M., Bowen, J.P., Harman, M. (eds.) *Formal Methods and Testing*. LNCS, vol. 4949, pp. 1–38. Springer, Heidelberg (2008). [https://doi.org/10.1007/978-3-540-78917-8\\_1](https://doi.org/10.1007/978-3-540-78917-8_1)