Science of Computer Programming ••• (••••) •••–•••

Contents lists available at ScienceDirect

Science of Computer Programming

www.elsevier.com/locate/scico

Formal modelling and verification of interlocking systems featuring sequential release

Linh Hong Vu [a,*,1], Anne E. Haxthausen [a,1], Jan Peleska [b,2]

[a] DTU Compute, Technical University of Denmark, Kongens Lyngby, Denmark
[b] Department of Mathematics and Computer Science, University of Bremen, Bremen, Germany

ARTICLE INFO

Article history:
Received 13 May 2015
Received in revised form 6 May 2016
Accepted 25 May 2016
Available online xxxx

Keywords:
Railway interlocking systems
Sequential release
Formal verification
Bounded model checking
k-Induction

ABSTRACT

In this article, we present a method and an associated toolchain for the formal verification of the new Danish railway interlocking systems that are compatible with the European Train Control System (ETCS) Level 2. We have made a generic and reconfigurable model of the system behaviour and generic safety properties. This model accommodates *sequential release* – a feature in the new Danish interlocking systems. To verify the safety of an interlocking system, first a domain-specific description of interlocking configuration data is constructed and validated. Then the generic model and safety properties are automatically instantiated with the well-formed description of interlocking configuration data. This instantiation produces a model instance in the form of a Kripke structure, and concrete safety properties expressed as invariants. Finally, using a combination of SMT based bounded model checking (BMC) and inductive reasoning, it is verified that the generated model instance satisfies the generated safety properties. Using this method, we are able to verify the safety properties for model instances corresponding to railway networks of industrial size. Experiments show that BMC is also efficient for finding bugs in the railway interlocking designs. Additionally, benchmarking results comparing the performance of our approach with alternative verification techniques on the interlocking models are presented.

1. Introduction

An interlocking system is responsible for guiding trains safely through a given railway network. It is a vital part of any railway signalling system and has the highest safety integrity level (SIL4) according to the CENELEC 50128 standard [31]. Conventionally, the development and verification process of interlocking systems is informal and mostly manual, and hence time-consuming, costly, and error-prone. Automated verification of interlocking systems is therefore an active research topic, investigated by several research groups, see e.g., [20,17,38,25,19,24]. As part of the RobustRailS research project,[3] our work aims at establishing a holistic method supporting the verification of such systems. The method should be formal and facilitate automation in order to provide a better verification process compared to the conventional one. In Denmark, in the period of 2009–2021, new interlocking systems that are compatible with the standardised European Train Control System

* Corresponding author.
E-mail addresses: lvho@dtu.dk (L.H. Vu), aeha@dtu.dk (A.E. Haxthausen), jp@cs.uni-bremen.de (J. Peleska).
[1] The authors' research has been funded by the RobustRailS project granted by Innovation Fund Denmark under grant agreement 0603-00483B.
[2] The author's research has been partially funded by ITEA2 project openETCS under grant agreement 11025.
[3] http://robustrails.man.dtu.dk.

http://dx.doi.org/10.1016/j.scico.2016.05.010
0167-6423/

(ETCS) Level 2 [13] will be deployed in the entire country within the context of the Danish Signalling Programme.[4] In the context of the RobustRailS project accompanying the signalling programme on a scientific level, the proposed method will be applied to these new systems.

The main contributions presented in this article are the following. (1) We present a formal model of the behaviour of ETCS Level 2 compatible interlocking systems. (2) The model accommodates sequential release: this is a method for incrementally releasing route portions that have been traversed by the associated train, with the objective to increase the level of concurrency in route allocation and, consequently, the train throughput in the railway network. On the other hand, this feature poses extra challenges to the verification tasks as it makes the models more complex. (3) The state space encodings allow for high-level safety properties and state transition relations to be processed in a highly efficient manner by Satisfiability Modulo Theories (SMT) solvers supporting bit vector and integer arithmetic. (4) A verification technique combining induction with bounded model checking (BMC) using novel SMT solvers enables the verification of safety properties for railway network instances of industrial size.

Compared to the previously published work by the same authors [37], the following main extensions have been added: (E-1) the state space encodings and state transition relation are elaborated in more detail, (E-2) new experimental results are presented for the early deployment line (EDL) of the Danish Signalling Programme, and (E-3) benchmarking experiments comparing our BMC approach to other techniques such as BDD-based techniques or CEGAR techniques – by translating our models to nuXmv [9] – are reported.

The remainder of the article is organised as follows: First, in Section 2, some mathematical preliminaries are explained and Section 3 gives a brief introduction to the new Danish route-based interlocking systems. Then the proposed method is presented in Section 4. Next, the five steps constituting the method are described in more detail: (1)–(2) the specification and validation of configuration data are presented in Section 5 and Section 6, respectively; (3)–(4) the generation of Kripke structure models of the behaviours of interlocking systems as well as their desired formal properties, in Section 7 and Section 8, respectively; and (5) the verification strategy in Section 9. Afterwards experimental results with the toolchain as well as benchmarking results are shown in Section 10. Finally, related work and concluding remarks are presented in Section 11 and Section 12, respectively.

2. Mathematical preliminaries

This section explains some mathematical preliminaries that are used in our method, in particular Kripke structures and the k-induction scheme for proving invariants in a Kripke structure.

2.1. Kripke structures

Kripke structures are used to specify *behavioural models* of interlocking systems in our method. A *Kripke structure* K is a five-tuple (S, s_0, R, L, AP) with state space S, initial state $s_0 \in S$, a total transition relation $R \subseteq S \times S$, and labelling function $L : S \to 2^{AP}$, where AP is a set of atomic propositions and 2^{AP} is the power set of AP. The labelling function L maps a state s to the set $L(s)$ of atomic propositions that hold in s.

Two states s and s' are said to be *consecutive* in K, if there is a transition from s to s', i.e., $(s, s') \in R$. A *path* in K is a finite or infinite sequence of consecutive states. A state s' is said to be *reachable* from another state s in K, if there exists a finite path $s \ldots s'$ starting in s and ending in s'. A state $s \in S$ is said to be *reachable* if it is reachable from the initial state s_0. The *reachable states* of K is the set of all reachable states.

In the context of this article, the states of a Kripke structure are represented by valuation functions $s : V \to D$ over finite sets $V = \{v_0, \ldots, v_n\}$ of variables, where each variable $v_i \in V$ has an associated finite domain D_{v_i}. The range of a state s is $D = \bigcup_{v \in V} D_v$. The whole state space S is the set of all valuation functions $s : V \to D$ for which $s(v) \in D_v$ for all $v \in V$. The *equality* relation $(=)$ between states is defined by the equality of mathematical functions as follows: two states s and s' are equal – denoted by $s = s'$ – iff every variable $v \in V$ is evaluated to the same value in s and s', i.e.,

$$(s = s') \equiv \left(\bigwedge_{v \in V} s(v) = s'(v) \right) \tag{1}$$

For a proposition ϕ over free variables in V, we use $\phi(s)$ to denote the proposition obtained by replacing every occurrence of $v \in V$ in ϕ by the value $s(v)$. A proposition ϕ over free variables in V is said to *hold* in a state $s \in S$, denoted as $s \models \phi$, iff $\phi(s)$ holds. An *invariant* in K is a proposition that holds in all reachable states of K. The initial state s_0 can be represented by the following proposition.

$$\mathcal{I}(s_0) \equiv \bigwedge_{v \in V} v = s_0(v) \tag{2}$$

[4] http://www.bane.dk/signalprogrammet.

Please cite this article in press as: L.H. Vu et al., Formal modelling and verification of interlocking systems featuring sequential release, Sci. Comput. Program. (2016), http://dx.doi.org/10.1016/j.scico.2016.05.010