An Integrated Perspective on the Evaluation of Complex Railway Systems

Davide Basile[1], Maurice H. ter Beek[1(✉)], Laura Carnevali[2], Silvano Chiaradonna[1], Felicita Di Giandomenico[1], Alessandro Fantechi[1,2], and Gloria Gori[2]

[1] CNR–ISTI, Pisa, Italy
{davide.basile,maurice.terbeek,silvano.chiaradonna, felicita.digiandomenico}@isti.cnr.it
[2] DINFO, University of Florence, Florence, Italy
{alessandro.fantechi,laura.carnevali,gloria.gori}@unifi.it

Abstract. The project ADVENTURE (ADVancEd iNtegraTed evalUation of Railway systEms) aims to provide novel solutions for the evaluation of RAMS requirements as well as to present trade-offs between dependability attributes and energy consumption in complex railway systems, leveraging both qualitative and quantitative evaluation methods. To this end, case studies concerning distributed interlocking systems, standard interfaces, and railroad switch heaters are considered, comprising different challenging scenarios, notably representative of the complexity of railway systems. In this paper, we illustrate the objectives of the project and the activities planned to address them, devising future steps to integrate the envisaged contributions within a unified framework.

1 Introduction

As part of the European Green Deal, the EU has adopted a Sustainable and Smart Mobility Strategy[1]. This paves the way to more sustainable, smart, and resilient mobility, promoting rail as one of the most sustainable, smart, and safe means of transport currently available. It also encourages individuals and businesses to switch to train travel. This effort resulted in a Master Plan[2], which lists seven challenges that the future rail system needs to address. This list includes the "Need for improved performance and capacity. In order to deliver an overall more sustainable transport system, rail must be able to accommodate increased demand. New infrastructure will be necessary in certain areas, but the vast bulk

[1] https://transport.ec.europa.eu/system/files/2021-04/2021-mobility-strategy-and-action-plan.pdf.
[2] https://rail-research.europa.eu/wp-content/uploads/2022/03/EURAIL_Master-Plan.pdf.

T. Margaria and B. Steffen (Eds.): ISoLA 2024, LNCS 15223, pp. 190–207, 2025.
https://doi.org/10.1007/978-3-031-75390-9_13

of future increased capacity must leverage existing infrastructure, through a systemic digitalization and automation of operations". Another challenge posed by this Master Plan is the need for interoperability across Europe, since "railways across Europe do not operate in the same manner and use a variety of technical systems, which are neither integrated nor interoperable". Formal methods have successfully been applied in railways to address such challenges, mixing both qualitative and quantitative evaluation techniques [13,14,33,34,57].

In the Sustainable and Smart Mobility objective, great emphasis is placed on promoting railways as a means of transport capable of reducing environmental impact and energy consumption, for which it is crucial to increase attractiveness of railway services in terms of frequency and capillarity. To this end, railway systems must guarantee a set of expected *Key Performance Indicators* (KPIs) such as safety of the train movement, capacity (e.g., number of trains or passengers per time unit), energy efficiency, operating cost, etc. Nowadays, and even more in the future, these KPIs are determined by the collective operation of a number of innovative subsystems that cooperate to the smooth working of railway systems, notably supporting monitoring, command, and control of physical railway equipment. The many specific and complex interactions among these subsystems raise new challenges that endanger accurate and efficient estimation and evaluation of KPIs, as well as safe interoperability. On the one hand, addressing these challenges requires overcoming the limitations of state-of-the-art hierarchical and compositional techniques for estimation of non-functional attributes of component-based systems, to properly fit railway needs. On the other hand, advancements on formal specification of behavioral interfaces among heterogeneous components are advocated to improve the reliability of the composition of railway sub-systems while reducing their cost.

In this paper, we present ADVENTURE (ADVancEd iNtegraTed evalUation of Railway systEms), a two-year project funded by the Italian Ministry of Universities and Research (MUR) as part of the PRIN 2022 PNRR Research Projects of National Relevance funded under the National Recovery and Resilience Plan program. ADVENTURE aims at developing innovative solutions for the evaluation of complex railway systems. Using Model-Driven Engineering (MDE) [51,56] methods and multi-paradigm or multi-formalism approaches to help create bridges between different abstraction levels, ADVENTURE focuses on the following objectives: i) qualitative evaluation of safety of complex distributed railway systems, by means of diverse techniques such as compositional model checking, synthesis of specifications given as behavioral interfaces, and tool support for relating specifications with implementations; ii) quantitative evaluation of dependability attributes in spite of failures, in particular considering communication failures, through quantitative modeling and evaluation of the timed failure logic of the system; and iii) quantitative evaluation of trade-offs between energy efficiency and availability/reliability, considering different smart policies of energy saving and considering failures, criticalities, and priorities of the system under analysis. The developed solutions will be experimented and validated by their application to different case studies, that are considered as representative

of the innovation trends in railways, namely decentralized interlocking systems, standard interfaces, and smart deicing systems.

A common trait of the systems of these case studies is that they can be considered as distributed Cyber-Physical Systems ensuring safe transit of trains along a station route. In all cases, the route is allocated if specific conditions are verified on a set of elements lying along the route, with also an eye to the energy consumption in case the involved equipment are characterized by energy-intensive operation. A failure of one of these elements generally means that the itinerary is unavailable: in this case, the availability and thus the overall transit capacity of the station decreases as well, with the possible occurrence of single points of failure blocking all operations. Modeling such a complex system for the purpose of quantitative assessment of availability suffers from the problem of state-space explosion. It is therefore desired to: i) identify a model-based compositional method to analyze such a complex network by combining results of the analysis of its elements, with the aim of performing network analysis in linear time with respect to the number of elements (the approach will be tried on some topological instances of the case studies); and ii) generalize the identified approach so that it can be automatically instantiated on different network topologies, both for ADVENTURE's case studies and for networks defining other distributed systems that have similar characteristics or similar dependability requirements

Outline Sect. 2 describes the selected case studies, while Sect. 3 discusses the qualitative and quantitative techniques that will be adopted to analyse them. The next steps to perform during the remainder of the project are described in Sect. 4, after which we conclude in Sect. 5.

2 Background and Context

The selected case studies are representative of the complexity of systems that support the safe transit of a train in a station, namely decentralized interlocking systems and switch heaters. Their complexity is related to the layout of the tracks in a station, which can be a quite complex graph for a large station.

2.1 Distributed Interlocking and Standard Interfaces

Railway interlocking systems (Input and eXit Locking, IXLs) are responsible for granting a train the exclusive access to a *route*, i.e. a sequence of track elements that are exclusively assigned for the movement of a train through a station or a network.

Current computer-based interlocking systems usually have a centralized design, with all logic residing in a single computer. Large stations are sometimes partitioned in (still quite large) areas, each governed by a subsystem of the whole interlocking. Centralized interlockings are complex to design.

A general trend, akin to *edge computing*, is to attack complexity by means of distributed intelligence, that is, multiple computational elements, each dedicated to a specific portion of the physical system to be monitored, controlled and commanded. This trend has produced several proposals of distributed interlocking systems: the work in [30] analyzes some of them and compares them with respect to the following aspects: distribution granularity, where the data defining the station topology reside, and how such data rules the communication between nodes.

The first case study of ADVENTURE considers IXL decentralization at different granularity levels, starting from the proposal in [31], which formalizes the function of each computational element using UML Statecharts and enables formal verification by means of the UMC model checker [12]: in any case, correctness of communication between computing elements must be ensured by correct interactions between properly defined interfaces. Such interfaces should be designed so that any failure (e.g., a communication failure between two computing elements) does not allow trains to be routed on conflicting tracks. They should support a robust distributed consensus on the availability of tracks in the controlled portion, also in case of failure of one of the computational elements, again to avoid potentially incorrect routing of trains. Given that the different computational elements could be produced by different manufacturers, *standard interfaces* should be rigorously defined to allow for their interoperability.

We now discuss standard interfaces. The adoption of formal methods to rigorously define standard interfaces has been advocated as a general issue in railways system technology due to the geographical distribution and number of different systems, subsystems and components [11,15] (cf. EULYNX[3] collective of railway Infrastructure Managers). Behavioral contracts [35] have been introduced to model behavioral interfaces, and can be applied to formalize the interfaces between components (cf. Sect. 3.1). This first case study will thus address the formal definition of suitable standard interfaces for the elements of a distributed interlocking system. Such definition will support the verification that safety is not jeopardized by communication failures, nor by other kinds of failure.

Moreover, availability and capacity concerns also comprise relevant issues in this case study. In fact, in the railway domain, halting trains in case of problems is the basic form of safety enforcement, which is achieved at the cost of reduced availability. As the number of controlling elements increases, safety mechanisms that simply halt trains at any single element failure could easily lower the availability until below acceptable limits, which in its turn lowers transport capacity due to missing service. For this reason, availability and capacity requirements have to be analyzed quantitatively for distributed interlocking systems, in order to produce minimal availability requirements for the single controllers.

[3] https://eulynx.eu/.

2.2 Railroad Switch Heaters

In this case study, the focus is on the high energy-consuming railway switch heaters, used to avoid the formation of snow and ice on top of railway switches during the cold season, in order to guarantee their correct functioning. Railway switches are mechanical installations enabling trains to be guided from one track to another. They are a critical part of safe, reliable railway operations, since correct routing of trains strongly depends on the correct operation of such switches. In fact, in presence of malfunctions, train derailments or train collisions could occur, with expected catastrophic consequences for passengers. During the winter months, snow and ice buildup on track can prevent switches from properly aligning and locking into place.

Therefore, nowadays switches are equipped with a deicing system, commonly constituted by electric heaters installed in their vicinity, automatically operated to keep the temperature around the switches above freezing. In addition to the physical composition of the heater and of its placement around the switch, it is the policy adopted to switch the heater on/off that impacts on the energy consumption. Two major directions can be explored to fulfil the goal of reducing energy consumption in the addressed context: enhancing the switch heating and protection technology, or improving the policy that controls the operation of the heater. Since computing and communication technologies are expected to play a significant role to enable novel integrated control and management strategies in which heterogeneous data is exploited to noticeably increase energy efficiency, the second direction is chosen in this project.

From this perspective, there are several useful contributions to progress towards more sustainable railway systems. These include the definition of advanced control policies of the deicing system as well as of analysis frameworks to assess KPIs that are representative of both energy consumption and dependability properties. In fact, the former pursue a cautious usage of energy, especially in critical conditions when malfunctions occur, which is relevant from the environmental and sustainability viewpoint. The latter are needed to evaluate the impact of the chosen energy supply policy on dependability properties of the railroad switch system, to guide the choice of the best heating policy among the available alternatives, and to find the optimal tuning of policy parameters that assure the desired level of trade-off between the conflicting aspects of energy consumption and safe/reliable operation of the switches. ADVENTURE provides useful contributions to progress towards more sustainable railway systems by developing advanced control policies of the deicing system on one side, as well as analysis frameworks addressing KPIs representative of both energy consumption and dependability properties on the other side.

In this paper, the focus is on the analysis framework, assuming a variety of alternative switch heater policies and an overall hierarchical organization of the control system under analysis as depicted in Fig. 1. At a rather abstract level (for details, cf. [24]), the distributed ICT control system consists of local (*LocalControllers*, at the specific switch heater) and global (*Coordinators*, at higher level encompassing all the switch heaters under its control) operations,

both in case of the "nominal operational mode" (where no faults occur) and in the "critical operational mode" (where the communication subsystem is affected by faults preventing a subsets of control components local to switch heaters to send/receive weather related information necessary to the energy management policy).

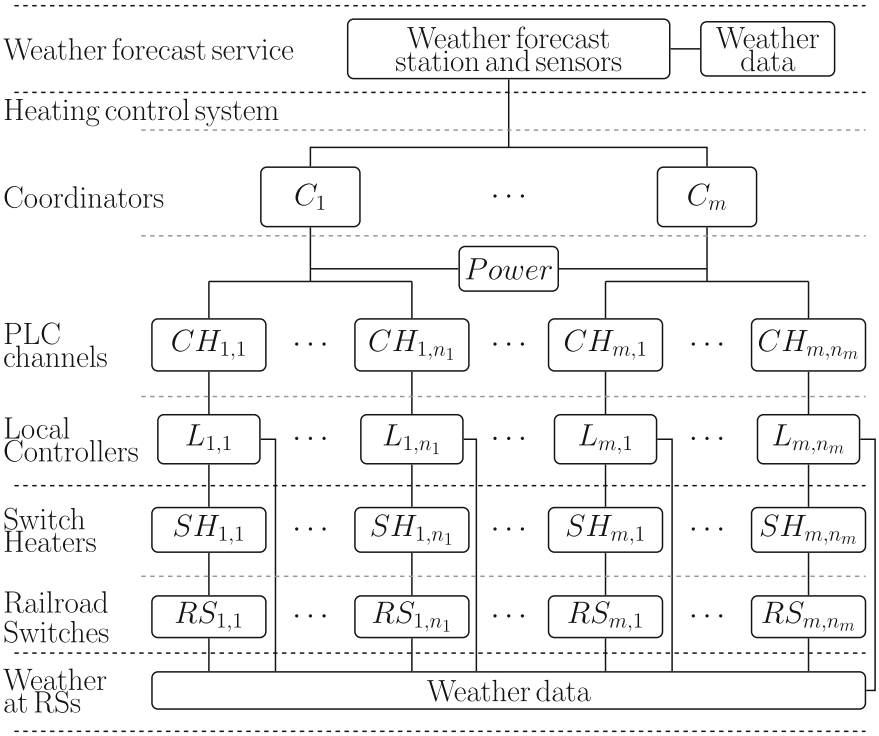

Fig. 1. Logical architecture of the Switch Heaters control system.

3 Qualitative and Quantitative Evaluation Techniques

As anticipated, the case studies will be used as workbenches to develop and assess innovative techniques for qualitative evaluation of safety and for quantitative evaluation of dependability attributes and their interactions. In the following, we present contract automata as an example of the former class of techniques, after which we discuss the possibilities offered by different quantitative evaluation techniques based on stochastic modeling. There are other contract-based formalisms and tools, some of which have also been applied to the railway and other transportation domains (cf., e.g., AGREE [46] and OCRA [44]).

3.1 Contract Automata

Contract automata (CA) are a dialect of finite state automata used to formally specify behavioral contracts [8]. A composition of contracts can be refined to a safe one using the orchestration synthesis algorithm [6,8]. Behavioral contracts can be used to formalize interfaces between components (cf. Sect. 2.1).

In this section, we briefly describe the application of CA to a simple example of a decentralized interlocking system which exploits autonomous (satellite-based) train positioning [10]. In this decentralized interlocking, each junction area is separately and independently commanded by a single interlocking. Virtual track circuits (VTCs) are used to detect the positions of trains. VTCs are not physical devices, but refer to virtual positions on a map: GPS signals received by trains are routed to the interlocking system to detect the presence of trains in VTCs.

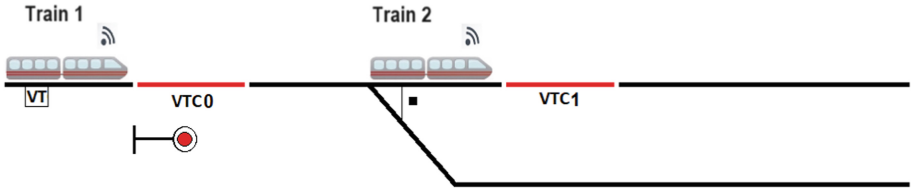

Fig. 2. The scenario taken from [10], where Train 1 is waiting to enter a junction area while Train 2 is traversing it.

In the scenario in Fig. 2, one junction area (commanded by one IXL) is composed of two VTCs, and there is one train outside the junction area and one train inside the junction area. Train 2 is traversing its assigned route, while Train 1 is waiting at a red signal for its route to be assigned. VTC 0 is used to detect the occupation of a route, whereas VTC 1 is used to detect the release of a route. Initially, both trains are located behind the semaphore. The first train arriving (Train 2) will communicate its route to the interlocking, which will proceed to set the route. This may cause the movement of the junction point. Once the route is set, the interlocking will open the semaphore to Train 2. Train 2 enters the junction point and the semaphore is closed again. While Train 2 is traversing its route, the second train arriving (Train 1) will stop at the (closed) semaphore to ask for its route. The route will be assigned, the junction point moved, and the semaphore opened only after Train 2 has exited the junction area. Otherwise, the movement of the junction point could cause the derailment of Train 2 inside the junction area [10].

Figure 3 shows three of the five CA used to model this scenario. The automata specify the behavioral interfaces of the components. The states of the **T**rain automaton represent coordinates in a bidimensional map (the railway track) of where the train is moving. Labelled transitions between states are used to model the train requests (prefixed by `?`) to move to an adjacent location. This

automaton models the railway track depicted in Fig. 2. The junction area is identified as the location with coordinate $x = 4$, where the train is allowed to perform the transition $[(4;1)] \xrightarrow{[?goup]}_{\square} [(4;0)]$ modeling the traversing of the point. Furthermore, in the experiments reported in detail in [10], the location of the semaphore is set to have coordinate $x = 2$.

A **D**river automaton is used to synchronize with the actions of the train to move. Similarly, the **S**emaphore control automaton controls the semaphore by opening and closing it. We do not display the automaton of the semaphore (identical to the semaphore control in Fig. 3, but with complemenary actions) and the automaton of the second train (identical to the automaton of the first train in Fig. 3, but with initial location $(1;1)$).

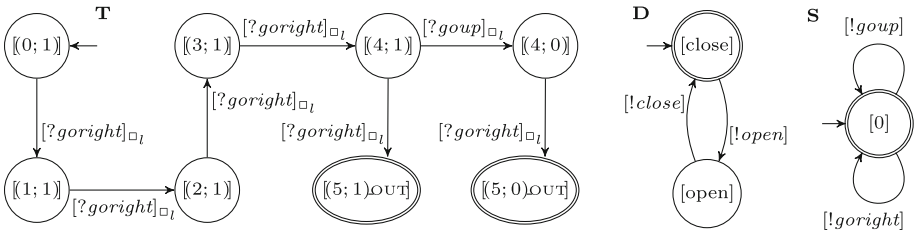

Fig. 3. The principal CA of the first **T**rain, the **D**river and the **S**emaphore control.

The composition of these five CA is defined as a synchronous product, in which forbidden states (declared through a predicate) are pruned from the composition. The forbidden states are states satisfying one or more of these requirements: i) both trains are in the same location (and are not in the final location modeling the exit from the area); ii) one train is at the semaphore location whilst the semaphore is closed; iii) both trains are inside the junction; iv) a train is inside the junction area and the semaphore is opened; and v) the semaphore is opened but no train is near it (in this case the semaphore must be closed). It is easy to see that all these requirements are invariants that can be checked on a state of the composition during the state space generation. This example has been computed using `CATLib` [3], a Java library implementing CA and their operations (e.g., composition, synthesis). Specifically, the above requirements have been expressed by exploiting the Java logic primitives. For example, as shown in Fig. 3, each state of the **T**rain automaton contains its coordinates as attributes. To check whether a train is inside the junction, it suffices to check whether its coordinates are within an interval that identifies the junction area.

The composed automaton is then refined to a strategy for the semaphore controller to command the opening and closing of the semaphore in such a way that both trains are allowed to reach the exit while satisfying the above requirements. The strategy is synthesized using the orchestration synthesis algorithm of CA, which is a variant of the standard most permissive controller synthesis [8]. In a nutshell, the synthesis works as follows. Starting from the composition,

paths leading to forbidden states are removed, whenever they involve removing controllable transitions. For what concerns uncontrollable transitions leading to forbidden states, these cannot be directly removed. In this case, the source states of these uncontrollable transitions are made unreachable in the strategy automaton.

3.2 Quantitative Modeling and Evaluation Techniques

Stochastic models support quantitative evaluation of dependability properties, providing a formal abstraction of the system behavior within its environment, and thus supporting the definition of formal solution techniques that quantify the measures of interest. According to the classification of [48], there are three major classes of model-based quantitative evaluation methods: i) *combinatorial methods*, which leverage models that explicitly encode relations among component behaviors (e.g., fault-to-failure chains), like Reliability Block Diagrams (RDBs) [20] and Fault Trees (FTs) as well as their extensions into dynamic RBDs (DRBDs) [29] and dynamic FTs (DFTs) [40] capable of handling statistical dependencies among component behaviors; ii) *stochastic model-checking methods*, which leverage reachability analysis of selected portions of the model state space to assess properties of interest expressed in some stochastic temporal logic [42], notably including Statistical Model Checking (SMC) [1,43] which uses a simulation-based approach for reachability analysis and property evaluation; and iii) *state-based stochastic methods*, which leverage different modeling formalisms and different (analytical or simulative) solution techniques depending on the class of the underlying stochastic process [25,59], with non-Markovian approaches improving the model expressivity while suffering the evaluation of large-scale models with respect to Markovian approaches.

Many of these approaches have been exploited to design quantitative models of the failure logic [41,45,52,58] of component-based systems [16] and to support the analysis of the chain of threats to system dependability [2], enabling early validation of design choices and development of predictive analytics [53,54]. To preserve consolidated industrial practices while guaranteeing affordable complexity of model analysis, various approaches [17,21,26,32,47] leverage MDE principles [51,56] to specify the system failure logic, mainly through UML extensions and the Architecture Analysis & Design Language (AADL), and to automatically derive quantitative formal models such as Stochastic Time Petri Nets (STPNs) to perform analyses [49]. In particular, FaultFlow [19,50] comprises a promising open-source Java library [18] for modeling and evaluation of dependability of component-based systems, combining ease and expressivity of modeling with efficiency of evaluation: a custom metamodel, encoding fault propagations with non-Markovian (expolynomial, also termed exponomial [60]) duration distribution, between directly and indirectly connected components; representation of repeated events in the failure logic; derivation of metamodel instances from SysML Block Definition Diagram (BDDs) and Stochastic Static Fault Trees (SSFTs), and their translation into STPNs, deriving the time-to-failure distribution through semi-symbolic analysis based on stochastic state

classes [22]; translation of metamodel instances into an extension of UML statecharts [37] if there are no repeated events, deriving fault importance measures by efficient numerical analysis [21]. Nevertheless, neither FaultFlow nor any of the mentioned model-based quantitative methods has been specifically applied and customized to the railway domain. For instance, FaultFlow could be extended to support SysML BDDs annotated with stereotypes modeling specific components of railway systems or to facilitate the representation of exponomial distributions that fit statistical duration data available in the railway setting (e.g., exponomial distributions fitting the 5th percentile, the 95th percentile, and the mean value).

State-based stochastic methods are recognized also as an effective method to perform quantitative assessment targeting trade-offs among system properties showing contrasting effects on each other (e.g., [36,38] involve power consumption), resorting to different formalisms such as automata [39], hybrid Petri nets [28], Stochastic Activity Networks (SANs) [55]. For deicing systems, evaluation frameworks to assess the energy consumption in conjunction with dependability-related properties have been investigated under specific assumptions on the system behavior, fault model and environment conditions. Among others, in [9] energy consumption and reliability indicators of energy management policies have been evaluated through SMC using the Uppaal SMC toolset, while in [23] a similar study (extended also to availability) has been conducted using SANs and the Möbius tool [27] (for a comparison, cf. [7]). More recently, the approach of [24] advances previous frameworks by pursuing fusion of available heterogeneous data (e.g., weather information made available from meteorological stations) with data collected from local sensors to achieve better energy saving. Still, advancements would be desirable, mainly in terms of improving the ability of the model to represent realistic scenarios.

4 Project Outlook

The ADVENTURE project investigates innovative qualitative and quantitative evaluation methods, according to the research lines described in Sects. 4.1 and 4.2. Also, a main objective of the project is the integration of such evaluation methods on the proposed case studies involved, as explained in Sect. 4.3.

4.1 Qualitative Methods

In this section, we discuss the definition of qualitative methods. One of the main issues is the investigation of novel synthesis algorithms and scalability techniques for CA [5]. Indeed, extensions to the current state-of-the-art are necessary to support richer specifications, needed to formalize railway interfaces. Other formalisms and techniques will also be investigated for their potential application to model and verify the proposed case studies.

The project also focuses on the problem of distinguishing correct too coarse abstractions from precise ones, in relation to the actual feasibility of a given formal analysis. The level of abstraction can only be estimated by an a-posteriori

analysis, i.e., when the implementation has been produced and it becomes possible to measure how many details were abstracted away. A proof-of-concept distributed system related to CA is currently being modelled and implemented to test the adequacy of the adopted level of abstraction using model-based testing. In particular, the chosen distributed system is the Contract Automata Runtime Environment (CARE) [4], which is a middleware used for implementing applications specified with CA.

4.2 Quantitative Methods

In this section, we discuss the definition of quantitative methods for the evaluation of the system failure logic and dependability/energy consumption trade-offs.

MDE methods can be used to effectively support modeling and analysis of complex component-based systems studied by the ADVENTURE project, addressing different challenges concerned with the expressivity of models and the computational complexity of their analysis. On the one hand, a custom extensible observation metamodel needs to be developed to effectively represent time-stamped observations of the system behavior, capturing not only fault and failures of system components, but also active measurements taken on specific system variables. On the other hand, metamodel instances need to be automatically derived from semi-formal specifications, to make the modeling step easier for users, and translated into stochastic formal models such as STPNs, not only to perform analysis of the failure logic model and derive dependability measures of interest, but also to support generation of synthetic data sets of observations via model simulation. In particular, these synthetic data could be used to train and test online failure prediction methods, overcoming the difficulties arising in acquiring real data sets for safety-critical systems like the ones considered in the ADVENTURE case studies.

Focusing on the highly energy-consuming switch heaters, research in this area has been active since several years, producing a stochastic modeling framework suitable to assess and compare the ability of a group of switch heating policies in trading between energy consumption and dependability (e.g., [24]). In the context of the ADVENTURE project, research will build upon this existing framework, providing advancements mainly in terms of i) development of innovative and integrated control and management strategies; and ii) improvement of the model ability to represent realistic scenarios.

Regarding point i), research concentrates on extending the set of energy management policies, with the aim of offering to railway operators a larger variety of solutions that can better fit their needs in terms of accuracy and/or flexibility of the mechanism, depending on the available input data and failure assumptions to be considered.

Regarding point ii), the envisioned modeling framework will be able to account for: a) specific heterogeneous aspects characterizing the switch heaters (including different reliability values, environmental conditions and specific failure events), in accordance with their position within the railway station; b) real usage of the railway switch each heater is attached to, in accordance with the

schedule of trains on the tracks where they are located; and c) new KPIs integrating aspects of efficiency and dependability attributes, to make better-informed decision on the energy management policy that leads to the best trade-off, e.g., considering also cost factors to quantify the impact of violation of availability of the railway switch (possibly considering different categories of trains, e.g., freight trains, high-speed passenger trains, local passenger trains).

The above listed new investigations are mainly based on: i) adoption of advanced stochastic modeling features, promoting efficient replication and composition of submodels, to deal with the increased heterogeneous aspects to be managed; ii) definition of new KPIs in terms of model elements; and iii) development of new software functions and/or template models to manage the additional information on the structure and operation of the railway switch heating system, in order to allow the more sophisticated analysis of the scenarios of interest.

To demonstrate the applicability and usefulness of the developed framework, an evaluation campaign involving several representative scenarios will be performed. To assure sufficient variety of the analyses results, the selected scenarios will primarily differ in terms of: i) number of switches, ii) criticality levels of the switches, as determined by intensity of rail traffic involving them; and iii) number of different weather profiles in relation to switch partitions.

4.3 Path to Integration

The proposed case studies will also serve as a workbench to assess the integration of qualitative and quantitative evaluation techniques.

In the case of distributed interlocking systems, we start from the definition given in [31] as a collection of UML state machines: each state machine represents the behavior either of a *train*, or of a *track circuit* (linear track section acting also as a sensor of occupancy by a train) or of a *switch* (also called *point*). An actual station layout will consist of multiple objects, instances of these classes, that are connected together by cross-references.

We will define the behavior of these objects at their interfaces, using formal models (e.g., CA, UMC models): in this way, the interoperability of the distributed computational elements, together with matching contracts at the interfaces, will be demonstrated to guarantee the safe operation of the interlocking system, also in presence of failures of the objects themselves (i.e., of the computational elements that host them) or of communication between them. Note that both UMC and contract automata are developed and maintained by the Formal Methods and Tools lab of CNR–ISTI, one of this project's partners.

Safety is guaranteed by the fact that any failure to achieve consensus of the distributed protocol does not allow trains to move. Repeated failures could easily lower the availability under acceptable limits, with important effects on transport capacity. On the one hand, the distributed nature of the system enables minimization of service disruptions by re-routing through available nodes. On the other hand, the precise definition of interfaces enables location of the effects of node and communication failures. The estimation of probability distributions

for these failures then enables to feed appropriate models aimed to quantitatively evaluate availability figures. A challenge in this area is the definition of compositional methods to combine results obtained on the single elements or on groups of elements, so to achieve overall availability/capacity figures for the whole station, without incurring in state space explosion problems, by exploiting typical connection patterns between elements.

In the case of railroad switch heaters, we will work in a reverse fashion, i.e., addressing first the quantitative analysis of dependability/energy consumption trade-off on the basis of a distributed logical architecture of the system under analysis, while leaving the study of the possibility of a distributed implementation of the optimal heating policies as a later goal. This goal will enforce a precise definition of the interfaces between the distributed controllers of the switch heaters. In principle, this will allow a switch controller to participate both in the interlocking distributed algorithm and in the distributed heating algorithm. Figure 4 represents the ultimate goal: for a very simple station, the deployment of local controllers is shown, which communicate over the air to implement appropriate distributed algorithms; standard and formally defined interfaces guarantee proper collaboration even if the controllers are provided by different manufacturers. In particular, the boxes are local controllers, presenting the mentioned interfaces, and connected to physical devices, either sensing *track occupancy*, or commanding the *aspect* of signals, or commanding and controlling *points positions*, or commanding and controlling *heaters*: the two latter functions are allocated to the same controller. Onboard train equipment is not shown, but may be part of the distributed algorithms.

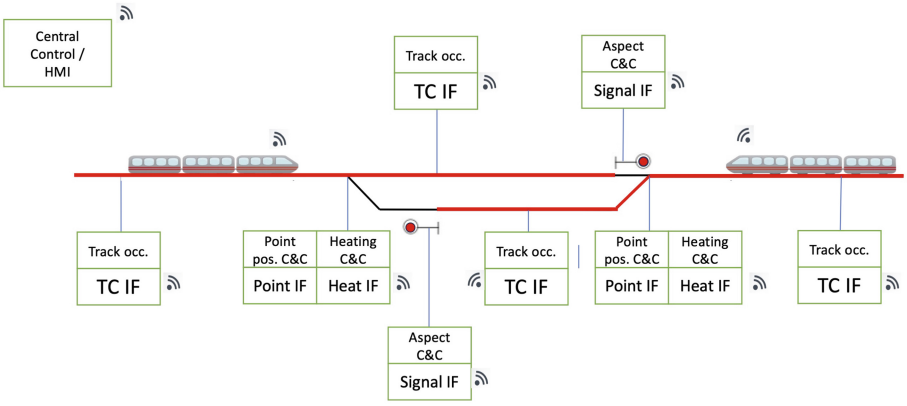

Fig. 4. The ultimate distributed system.

5 Conclusion

In this paper, we have presented the project ADVENTURE (ADVancEd iNtegraTed evalUation of Railway systEms). This project proposes innovative solutions for evaluating complex railway systems. Leveraging MDE and formal approaches, the project addresses safety, dependability, and energy efficiency of complex railway systems. In particular, the focus is on systems that are representative of the innovation trends in railways, such as decentralized interlocking systems and smart deicing systems for railway switches.

Novel techniques like compositional model checking and synthesis of behavioral interfaces are discussed, aimed at addressing qualitative safety in distributed railway systems. An example of formalisation of a distributed railway interlocking, with automatic synthesis of a safe controller, has been presented. The adopted formalism is contract automata.

Quantitative modeling and timed failure logic evaluation assess dependability in the face of communication failures. ADVENTURE also addresses cautious usage of energy, as requested in case of energy-intensive equipment, by developing methods to quantify trade-offs between energy efficiency, availability and reliability, as support to the design and incorporation of smart energy-saving policies.

The project aims to apply the proposed techniques in railway applications, including decentralized interlocking systems, standard interfaces, and smart deicing systems, paving the way for more reliable, safe and energy-efficient railway systems.

Acknowledgements. This study was carried out within the MUR PRIN 2022 PNRR P2022A492B project ADVENTURE (ADVancEd iNtegraTed evalUation of Railway systEms) and the MOST – Sustainable Mobility National Research Center and received funding from the European Union NextGenerationEU (PIANO NAZIONALE DI RIPRESA E RESILIENZA (PNRR) – MISSIONE 4, COMPONENTE 2, INVESTIMENTO 1.4 – D.D. 1033 17/06/2022, CN00000023. This manuscript reflects only the authors views and opinions, neither the European Union nor the European Commission can be considered responsible for them.

References

1. Agha, G., Palmskog, K.: A survey of statistical model checking. ACM Trans. Model. Comput. Simul. **28**(1), 6:1–6:39 (2018). https://doi.org/10.1145/3158668
2. Avizienis, A., Laprie, J., Randell, B., Landwehr, C.E.: Basic concepts and taxonomy of dependable and secure computing. IEEE Trans. Dependable Secur. Comput. **1**(1), 11–33 (2004). https://doi.org/10.1109/TDSC.2004.2
3. Basile, D., ter Beek, M.H.: Contract automata library. Sci. Comput. Program. **221** (2022). https://doi.org/10.1016/j.scico.2022.102841, https://github.com/contractautomataproject/ContractAutomataLib
4. Basile, D., ter Beek, M.H.: A runtime environment for contract automata. In: Chechik, M., Katoen, J., Leucker, M. (eds.) Formal Methods. FM 2023. LNCS, vol. 14000, pp. 550–567. Springer (2023). https://doi.org/10.1007/978-3-031-27481-7_31

5. Basile, D., ter Beek, M.H.: Advancing orchestration synthesis for contract automata. J. Log. Algebr. Methods Program. **141** (2024). https://doi.org/10.1016/j.jlamp.2024.100998
6. Basile, D., et al.: Controller synthesis of service contracts with variability. Sci. Comput. Program. **187** (2020). https://doi.org/10.1016/j.scico.2019.102344
7. Basile, D., ter Beek, M.H., Di Giandomenico, F., Fantechi, A., Gnesi, S., Spagnolo, G.O.: 30 years of simulation-based quantitative analysis tools: a comparison experiment between Möbius and Uppaal SMC. In: Margaria, T., Steffen, B. (eds.) Leveraging Applications of Formal Methods, Verification and Validation: Verification Principles. ISoLA 2020. LNCS, vol. 12476, pp. 368–384. Springer (2020). https://doi.org/10.1007/978-3-030-61362-4_21
8. Basile, D., ter Beek, M.H., Pugliese, R.: Synthesis of orchestrations and choreographies: bridging the gap between supervisory control and coordination of services. Log. Methods Comput. Sci. **16**(2), 9:1–9:29 (2020). https://doi.org/10.23638/LMCS-16(2:9)2020
9. Basile, D., Di Giandomenico, F., Gnesi, S.: Statistical model checking of an energy-saving cyber-physical system in the railway domain. In: Proceedings of the 32nd Symposium on Applied Computing (SAC'17), pp. 1356–1363. ACM (2017). https://doi.org/10.1145/3019612.3019824
10. Basile, D., Fantechi, A., Rucher, L., Mandò, G.: Analysing an autonomous tramway positioning system with the UPPAAL statistical model checker. Form. Asp. Comput. **33**(6), 957–987 (2021). https://doi.org/10.1007/s00165-021-00556-1
11. Basile, D., Mazzanti, F., Ferrari, A.: Experimenting with formal verification and model-based development in railways: the case of UMC and Sparx enterprise architect. In: Cimatti, A., Titolo, L. (eds.) Formal Methods for Industrial Critical Systems. FMICS 2023. LNCS, vol. 14290, pp. 1–21. Springer (2023). https://doi.org/10.1007/978-3-031-43681-9_1
12. ter Beek, M.H., Fantechi, A., Gnesi, S., Mazzanti, F.: A state/event-based model-checking approach for the analysis of abstract system properties. Sci. Comput. Program. **76**(2), 119–135 (2011). https://doi.org/10.1016/j.scico.2010.07.002
13. ter Beek, M.H.: Formal methods and tools applied in the railway domain. In: Bonfanti, S., Gargantini, A., Leuschel, M., Riccobene, E., Scandurra, P. (eds.) Rigorous State-Based Methods. ABZ 2024. LNCS, vol. 14759, pp. 3–21. Springer, Cham (2024). https://doi.org/10.1007/978-3-031-63790-2_1
14. ter Beek, M.H., et al.: Adopting formal methods in an industrial setting: the railways case. In: ter Beek, M.H., McIver, A., Oliveira, J.N. (eds.) Formal Methods – The Next 30 Years. FM 2019. LNCS, vol. 11800, pp. 762–772. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-30942-8_46
15. Belli, D., et al.: The 4SECURail case study on rigorous standard interface specifications. In: Cimatti, A., Titolo, L. (eds.) Formal Methods for Industrial Critical Systems. FMICS 2023. LNCS, vol. 14290, pp. 22–39. Springer, Cham (2023). https://doi.org/10.1007/978-3-031-43681-9_2
16. Boardman, J.T., Sauser, B.J.: System of Systems – the meaning of *of*. In: Proceedings of the 1st International Conference on System of Systems Engineering (SoSE'06), pp. 1–6. IEEE (2006). https://doi.org/10.1109/SYSOSE.2006.1652284
17. Bressan, L., de Oliveira, A.L., Montecchi, L., Gallina, B.: A systematic process for applying the CHESS methodology in the creation of certifiable evidence. In: Proceedings of the 14th European Dependable Computing Conference (EDCC'18), pp. 49–56. IEEE (2018). https://doi.org/10.1109/EDCC.2018.00019
18. Carnevali, L., Cerboni, S.: FaultFlow library (2023). https://github.com/oris-tool/faultflow

19. Carnevali, L., Cerboni, S., Picano, B., Scommegna, L., Vicario, E.: An observation metamodel for dependability tools. In: Proceedings of the 19th European Dependable Computing Conference (EDCC'24), pp. 169–172. IEEE (2024). https://doi.org/10.1109/EDCC61798.2024.00041
20. Carnevali, L., Ciani, L., Fantechi, A., Gori, G., Papini, M.: An efficient library for reliability block diagram evaluation. Appl. Sci. **11**(9), 4026:2–4026:24 (2021). https://doi.org/10.3390/app11094026
21. Carnevali, L., German, R., Santoni, F., Vicario, E.: Compositional analysis of hierarchical UML Statecharts. IEEE Trans. Softw. Eng. **48**(12), 4762–4788 (2022). https://doi.org/10.1109/TSE.2021.3125720
22. Carnevali, L., Ridi, L., Vicario, E.: A framework for simulation and symbolic state space analysis of non-markovian models. In: Flammini, F., Bologna, S., Vittorini, V. (eds.) SAFECOMP 2011. LNCS, vol. 6894, pp. 409–422. Springer (2011). https://doi.org/10.1007/978-3-642-24270-0_30
23. Chiaradonna, S., Di Giandomenico, F., Masetti, G.: Trading dependability and energy consumption in critical infrastructures: focus on the rail switch heating system. In: Proceedings of the 25th Pacific Rim International Symposium on Dependable Computing (PRDC'20), pp. 150–159. IEEE (2020). https://doi.org/10.1109/PRDC50213.2020.00026
24. Chiaradonna, S., Masetti, G., Di Giandomenico, F., Righetti, F., Vallati, C.: Enhancing sustainability of the railway infrastructure: trading energy saving and unavailability through efficient switch heating policies. Sustain. Comput. Inform. Syst. **30**, 100519 (2021). https://doi.org/10.1016/J.SUSCOM.2021.100519
25. Ciardo, G., German, R., Lindemann, C.: A characterization of the Stochastic process underlying a stochastic Petri net. IEEE Trans. Softw. Eng. **20**(7), 506–515 (1994). https://doi.org/10.1109/32.297939
26. Cicchetti, A., et al.: CHESS: a model-driven engineering tool environment for aiding the development of complex industrial systems. In: Proceedings of the 27th International Conference on Automated Software Engineering (ASE'12), pp. 362–365. ACM (2012). https://doi.org/10.1145/2351676.2351748
27. Courtney, T., Gaonkar, S., Keefe, K., Rozier, E., Sanders, W.H.: Möbius 2.3: an extensible tool for dependability, security, and performance evaluation of large and complex system models. In: Proceedings of the 39th International Conference on Dependable Systems and Networks (DSN'09), pp. 353–358. IEEE (2009). https://doi.org/10.1109/DSN.2009.5270318
28. David, R., Alla, H.: On hybrid petri nets. Discret. Event Dyn. Syst. **11**(1–2), 9–40 (2001). https://doi.org/10.1023/A:1008330914786
29. Distefano, S., Puliafito, A.: Dynamic reliability block diagrams: overview of a methodology. In: Aven, T., Vinnem, J.E. (eds.) Risk, Reliability and Societal Safety: Proceedings of the 18th European Safety and Reliability Conference (ESREL'07), pp. 1059–1068. Taylor and Francis (2007)
30. Fantechi, A., Haxthausen, A.E.: Safety interlocking as a distributed mutual exclusion problem. In: Howar, F., Barnat, J. (eds.) FMICS 2018. LNCS, vol. 11119, pp. 52–66. Springer (2018). https://doi.org/10.1007/978-3-030-00244-2_4
31. Fantechi, A., Haxthausen, A.E., Nielsen, M.B.R.: Model checking geographically distributed interlocking systems using UMC. In: Proceedings of the 25th Euromicro International Conference on Parallel, Distributed and Network-based Processing (PDP'17), pp. 278–286. IEEE (2017). https://doi.org/10.1109/PDP.2017.66
32. Feiler, P.H., Lewis, B.A., Vestal, S.: The SAE architecture analysis and design language (AADL): a standard for engineering performance critical systems. In: Pro-

ceedings of the Joint Conference on Computer Aided Control System Design, International Conference on Control Applications, and International Symposium on Intelligent Control (CACSD-CCA-ISIC'06), pp. 1206–1211. IEEE (2006). https://doi.org/10.1109/CACSD-CCA-ISIC.2006.4776814
33. Ferrari, A., ter Beek, M.H.: Formal methods in railways: a systematic mapping study. ACM Comput. Surv. **55**(4), 69:1–69:37 (2023). https://doi.org/10.1145/3520480
34. Ferrari, A., Mazzanti, F., Basile, D., ter Beek, M.H.: Systematic evaluation and usability analysis of formal methods tools for railway signaling system design. IEEE Trans. Softw. Eng. **48**(11), 4675–4691 (2022). https://doi.org/10.1109/TSE.2021.3124677
35. Gay, S., Ravara, A. (eds.): Behavioural Types: from Theory to Tools. River (2017). https://doi.org/10.13052/rp-9788793519817
36. Ghosh, R., Naik, V.K., Trivedi, K.S.: Power-performance trade-offs in IaaS cloud: a scalable analytic approach. In: Proceedings of the 41st International Conference on Dependable Systems and Networks Workshops (DSN-W'11), pp. 152–157. IEEE (2011). https://doi.org/10.1109/DSNW.2011.5958802
37. Harel, D.: Statecharts: a visual formalism for complex systems. Sci. Comput. Program. **8**(3), 231–274 (1987). https://doi.org/10.1016/0167-6423(87)90035-9
38. Haverkort, B.R., Postema, B.: Towards simple models for energy-performance trade-offs in data centers. In: Proceedings of the International Workshops SOCNET and FGENET (MMB/DFT'14), pp. 113–122. University of Bamberg (2014)
39. Henzinger, T.A.: The theory of hybrid automata. In: Proceedings of the 11th Symposium on Logic in Computer Science (LICS'96), pp. 278–292. IEEE (1996). https://doi.org/10.1109/LICS.1996.561342
40. Junges, S., Guck, D., Katoen, J., Stoelinga, M.: Uncovering dynamic fault trees. In: Proceedings of the 46th International Conference on Dependable Systems and Networks (DSN'16), pp. 299–310. IEEE (2016). https://doi.org/10.1109/DSN.2016.35
41. Kabir, S.: An overview of fault tree analysis and its application in model based dependability analysis. Expert Syst. Appl. **77**, 114–135 (2017). https://doi.org/10.1016/J.ESWA.2017.01.058
42. Kwiatkowska, M.Z., Norman, G., Parker, D.: Stochastic model checking. In: Bernardo, M., Hillston, J. (eds.) SFM 2007. LNCS, vol. 4486, pp. 220–270. Springer (2007). https://doi.org/10.1007/978-3-540-72522-0_6
43. Legay, A., Lukina, A., Traonouez, L., Yang, J., Smolka, S.A., Grosu, R.: Statistical model checking. In: Steffen, B., Woeginger, G.J. (eds.) Computing and Software Science: State of the Art and Perspectives. LNCS, vol. 10000, pp. 478–504. Springer (2019). https://doi.org/10.1007/978-3-319-91908-9_23
44. Limbrée, C., Cappart, Q., Pecheur, C., Tonetta, S.: Verification of railway interlocking: compositional approach with OCRA. In: Lecomte, T., Pinger, R., Romanovsky, A.B. (eds.) RSSRail 2016. LNCS, vol. 9707, pp. 134–149. Springer (2016). https://doi.org/10.1007/978-3-319-33951-1_10
45. Lisagor, O.: Failure logic modelling: a pragmatic approach, Ph.D. thesis, University of York (2010). https://etheses.whiterose.ac.uk/1044/
46. Liu, J., Backes, J.D., Cofer, D.D., Gacek, A.: From Design Contracts to Component Requirements Verification. In: Rayadurgam, S., Tkachuk, O. (eds.) NFM 2016. LNCS, vol. 9690, pp. 373–387. Springer (2016). https://doi.org/10.1007/978-3-319-40648-0_28
47. Montecchi, L., Lollini, P., Bondavalli, A.: Towards a MDE transformation workflow for dependability analysis. In: Proceedings of the 16th International Conference

on Engineering of Complex Computer Systems (ICECCS'11), pp. 157–166. IEEE (2011). https://doi.org/10.1109/ICECCS.2011.23
48. Nicol, D.M., Sanders, W.H., Trivedi, K.S.: Model-based evaluation: from dependability to security. IEEE Trans. Dependable Secur. Comput. **1**(1), 48–65 (2004). https://doi.org/10.1109/TDSC.2004.11
49. Paolieri, M., Biagi, M., Carnevali, L., Vicario, E.: The ORIS tool: quantitative evaluation of non-Markovian systems. IEEE Trans. Softw. Eng. **47**(6), 1211–1225 (2021). https://doi.org/10.1109/TSE.2019.2917202
50. Parri, J., Sampietro, S., Vicario, E.: FaultFlow: a tool supporting an MDE approach for timed failure logic analysis. In: Proceedings of the 17th European Dependable Computing Conference (EDCC'21), pp. 25–32. IEEE (2021). https://doi.org/10.1109/EDCC53658.2021.00011
51. Rodrigues da Silva, A.: Model-driven engineering: a survey supported by the unified conceptual model. Comput. Lang. Syst. Struct. **43**, 139–155 (2015). https://doi.org/10.1016/J.CL.2015.06.001
52. Ruijters, E., Stoelinga, M.: Fault tree analysis: a survey of the state-of-the-art in modeling, analysis and tools. Comput. Sci. Rev. **15**, 29–62 (2015). https://doi.org/10.1016/J.COSREV.2015.03.001
53. Salfner, F., Lenk, M., Malek, M.: A survey of online failure prediction methods. ACM Comput. Surv. **42**(3), 10:1–10:42 (2010). https://doi.org/10.1145/1670679.1670680
54. Salfner, F., Malek, M.: Using hidden semi-Markov models for effective online failure prediction. In: Proceedings of the 26th Symposium on Reliable Distributed Systems (SRDS'07), pp. 161–174. IEEE (2007). https://doi.org/10.1109/SRDS.2007.35
55. Sanders, W.H., Meyer, J.F.: Stochastic activity networks: formal definitions and concepts. In: Brinksma, E., Hermanns, H., Katoen, J. (eds.) Lectures on Formal Methods and Performance Analysis. LNCS, vol. 2090, pp. 315–343. Springer (2000). https://doi.org/10.1007/3-540-44667-2_9
56. Schmidt, D.C.: Model-driven engineering. IEEE Comp. **39**(2), 25–31 (2006). https://doi.org/10.1109/MC.2006.58
57. Seisenberger, M., et al.: Safe and secure future AI-driven railway technologies: challenges for formal methods in railway. In: Margaria, T., Steffen, B. (eds.) "Leveraging Applications of Formal Methods, Verification and Validation: Practice. ISoLA 2022. LNCS, vol. 13704, pp. 246–268. Springer (2022). https://doi.org/10.1007/978-3-031-19762-8_20
58. Stamatis, D.H.: Failure mode and effect analysis: FMEA from theory to execution. ASQ (2003). https://asq.org/quality-press/display-item?item=H1188#
59. Trivedi, K.S., Bobbio, A.: Reliability and availability engineering: modeling, analysis, and applications. Cambridge University Press (2017). https://www.cambridge.org/de/academic/subjects/engineering/engineering-general-interest/reliability-and-availability-engineering-modeling-analysis-and-applications
60. Trivedi, K.S., Sahner, R.A.: SHARPE at the age of twenty two. ACM SIGMETRICS Perform. Eval. Rev. **36**(4), 52–57 (2009). https://doi.org/10.1145/1530873.1530884