Statistical Model Checking of an Energy-Saving Cyber-Physical System in the Railway Domain

Davide Basile
Istituto di Scienza e
Tecnologia dell'Informazione
"A. Faedo"
Consiglio Nazionale delle
Ricerche, ISTI-CNR, Pisa,
Italy
d.basile@isti.cnr.it

Felicita Di Giandomenico
Istituto di Scienza e
Tecnologia dell'Informazione
"A. Faedo"
Consiglio Nazionale delle
Ricerche, ISTI-CNR, Pisa,
Italy
f.digiandomenico@isti.cnr.it

Stefania Gnesi
Istituto di Scienza e
Tecnologia dell'Informazione
"A. Faedo"
Consiglio Nazionale delle
Ricerche, ISTI-CNR, Pisa,
Italy
s.gnesi@isti.cnr.it

ABSTRACT

Studies devoted to reduce the energy consumption while guaranteeing acceptable reliability levels are nowadays gaining importance in a variety of application sectors. Analyses through formal models and tools help developers of energy supply strategies in properly trading between energy consumption and reliability. Generally, probabilistic phenomena are involved in those systems, and they can be modelled through stochastic formalisms. Validating these models is paramount, so to guarantee reliance on the analysis results they provide. In this paper, we uniformly address both evaluation and validation of energy consumption policies on a case study from the railway domain using formal techniques. In particular, we analyse a system of rail road switch heaters, which are used to keep the temperature of rail road switches above certain levels to assure their correct functioning. Strategies based on thresholds to control the energy supply are modelled through hybrid automata, a formalism which allows to analyse both the discrete and the continuous nature of cyber-physical systems. We verify the correctness of the proposed model, and we evaluate energy consumption and reliability indicators through Statistical Model Checking using the Uppaal SMC toolbox.

CCS Concepts

•Software and its engineering → Model checking; •Computer systems organization → Embedded and cyber-physical systems; Reliability; •Hardware → Power and energy;

Keywords

Statistical Model Checking, Green IT, Cyber-physical Systems

1. INTRODUCTION

Recently, studies dedicated to reduce the energy consumption in cyber-physical systems (CPS) [21] are gaining increasing attention, for saving both in economic terms and environmental impact. In CPS, digital control units interact with physical components and continuous phenomena affecting the surrounding environment. Examples of CPS can be found in disparate application domains, including the transportation sector.

Generally, when critical applications are considered, measures as dependability, performance, reliability, energy consumption are assessed through rigorous and formal approaches. To cope with potential defects introduced in the modelling phase, especially when error prone communication-based applications are involved, validation of the developed models is paramount. In fact, the introduced errors may compromise the accuracy of the results obtained through the analysis, which may lead to the delivery of flawed components, with both potential serious consequences for the components users and loss of time and money for industries, to recover from the late revealed deficiencies.

Approaches based on stochastic model-based analysis are useful for expressing the stochastic nature of physical phenomena involved in CPS, as well as dependability and efficiency aspects, as performance, energy consumed, probability of failures.

Concerning the verification of finite state systems, model checking [9] is a widely used and powerful approach, where a property usually specified in a temporal logic is automatically checked against a model of a system, by performing an exhaustive exploration of its state-space, obtaining a counter-example in case the property is not satisfied.

Generally, it is difficult to capture the continuous dynamic nature of CPS through discrete approaches; therefore, models as Timed automata [2], Hybrid Petri Nets [13] and Stochastic Activity Networks [23] have been applied for modelling CPS, where the evolution of the continuous variables can be described uniformly or by ordinary differential equations; and tools such as Uppaal [20], SimHPN [18], Möbius [8] have been proposed for their modelling, evaluation and verification.

When the continuous time behaviours of CPS are subject to complex and stochastic dynamics, the model checking problem is undecidable [17], and generally an approximation to more tractable models is performed. In this case, a valid alternative to model checking and testing is represented by *Statistical Model Checking* (SMC) [22]. SMC uses results from statistics on top of simulations of a system to decide whether a given property is satisfied with a specified degree of confidence. Uppaal SMC [12] has been proposed as a tool that implements SMC techniques.

In this paper, we conduct an evaluation and validation study of energy consumption policies on a specific system from the railway domain, the rail road switch heating system, by adopting a statistical model checking approach. Rail road switch heaters are es-

SAC 2017, April 03-07, 2017, Marrakech, Morocco
© 2017 ACM. ISBN 978-1-4503-4486-9/17/04…$15.00
DOI: http://dx.doi.org/10.1145/3019612.3019824

sential components for the correct functioning of railway stations, in absence of which possible disasters can be verified (i.e. derailments, trains collision). In particularly colder regions, ice and snow can prevent the switches to work properly, and heaters are used for guaranteeing the correct functioning of the rail road switch system. A central control unit is in charge of managing policies of energy consumption while satisfying reliability constraints, by communicating with the network of switches to manage the energy supply.

The contribution of this paper is twofold: we address (i) the quantitative assessment through SMC of stochastic measures of interest, i.e. reliability and energy consumption, for an energy-saving cyber physical system in the railway domain; and (ii) the qualitative verification through model checking of the correctness of important critical and error prone aspects of the developed stochastic model. Specifically, to accomplish this second task, the absence of deadlocks in the modelled interactions is verified. Note that safety of railway stations is generally guaranteed by additional specific mechanisms, as for example interlocking mechanisms.

The system is modelled as a network of stochastic hybrid automata, and the measures of interest are defined as properties in the Metric Interval Temporal Logic (MITL) [12]. The Uppaal SMC toolkit is used for evaluating such properties and for validating the network of hybrid automata. The proposed approach is useful for comparing different policies of energy consumption, so to select a suitable trade-off between energy consumption and reliability of the analysed system. The verification of model correctness allows relying trustworthiness upon the obtained results.

Structure of the paper.

To set the ground for the proposed approach, hybrid automata and Uppaal SMC are introduced in Section 2 and Section 3, respectively. The analysed case study is described in Section 4, together with its formalization in terms of the stochastic hybrid automata provided by Uppaal SMC. The evaluation and validation of the model are discussed in Section 5. Related work is presented in Section 6 while final conclusions are drawn in Section 7.

2. HYBRID AUTOMATA

Generally, in cyber-physical systems both discrete and continuous dynamics are involved. Timed automata [2] combine discrete systems with real-valued variables that evolve during the time a system spends in a state. These variables, called *clocks*, evolve with a uniform rate and they can be used for guarding transitions. Reachability and other key problems are decidable for timed automata, that have been implemented in tools such as Uppaal [20].

Hybrid automata [16, 19] generalises timed automata by including arbitrary dynamics for the real-valued variables (i.e. clocks), expressed through ordinary differential equations (ODEs), and are used in tools such as Uppaal SMC [12], where their stochastic extension is available (see Section 3). This model has been proved appropriate for specifying and verifying cyber-physical systems, and a number of case-studies demonstrate their applications [10] [12].

For simplifying the presentation, we slightly elaborate the formal definition of hybrid automata in [16, 19], starting by introducing some useful notation. In a hybrid automaton the states progress according to both continuous and discrete clocks, in the first case this behaviour is called *continuous flow*, while in the second *jump*. A flow function $\mathbb{R}^{|X|} \to \mathbb{R}^{|X|}$ characterises the flow (i.e. the dynamic) of the continuous variables in the set X through a system of ODEs $\dot{X} = F(X)$, where $\dot{X}$ is the first order derivatives of the variables in X, and as usual $\mathbb{R}$ is the set of real numbers. Moreover, let $\nu : X \to \mathbb{R}$ be a valuation of the variables in X, $\pi \in \text{pred}(X)$ be a predicate over X and $[\![\pi]\!] \in \mathbb{R}^{|X|}$ be the set of valuations of X that satisfies the predicate π. Predicates are used to (i) guard transitions, (ii) specify the jumps of a system (i.e. how variables evolve in a discrete-time step) and (iii) define the invariants for each state of the automaton. Hybrid automata are defined below (Definition 1). We will discuss their stochastic extension in Section 3.

Definition 1. A hybrid automaton $\mathcal{H}$ is defined as a tuple $\mathcal{H} = \langle Q,\ Q_0,\ \Sigma,\ X,\ \triangle,\ I,\ F,\ V_0\rangle$ where:

- Q is a finite set of *states* including a distinguished initial singleton set $Q_0 \subseteq Q$,
- Σ is a finite set of *actions*,
- X is a finite set of real-valued *variables*, called *clocks*,
- $\triangle \subseteq Q \times \text{pred}(X) \times \Sigma \times \text{pred}(X \cup X') \times Q$ is the *transition relation*,
- $I : Q \to \text{pred}(X)$ that assigns an invariant function to each state,
- $F : Q \to (\mathbb{R}^{|X|} \to \mathbb{R}^{|X|})$ that assigns a flow function to each state $q \in Q$ as the set of ODEs $X = F(q)(X)$, and
- $V_0 \in \text{pred}(X)$ is the set of initial valuations.

It is assumed that for each state $q \in Q$ the flow function $F(q)$ has unique solution. We now briefly describe the semantics of hybrid automata. A configuration of a hybrid automaton is a tuple (q, ν) where $q \in Q$ is a state and $\nu \in \mathbb{R}^{|X|}$ is a variable valuation.

The initial configuration of a hybrid automaton is (q_0, ν_0), where $q_0 \in Q_0$, $\nu_0 = [\![\pi]\!]$ such that $\pi \in V_0$ and $\nu_0 \in [\![I(q_0)]\!]$ (the invariant constraints are satisfied). During the time t a system spends in a state q, the clocks in X are updated according to the flow function of q, and at each step the new valuation must respect the invariant constraints in q. A transition $\delta = (q, g, a, j, q_1)$ is enabled after t time when the guard $g \in \text{pred}(X)$ is satisfied. When δ is executed, the automaton jumps to a new configuration (q_1, ν_1) such that q_1 is the target state of δ, ν_1 is the valuation of the jump constraints $j \in \text{pred}(X \cup X')$, and $\nu_1 \in [\![I(q_1)]\!]$.

Composing Hybrid Automata. For modelling complex hybrid systems it is convenient to adopt a modular approach where systems are described by interacting entities. This allows to separately verify different smaller components, that is more efficient then verifying a bigger monolithic model. Hybrid automata can be composed through a synchronous product operator, and they interact through actions and shared variables. Let $I = \{1, \ldots, n\}$ be a set of indexes, the product of hybrid automata $\bigotimes_{\mathcal{H}_i \in C} \mathcal{H}_i$, where $C = \{\mathcal{H}_i \mid i \in I\}$ is defined below.

Definition 2. Let C be a set of hybrid automata where $C = \{\mathcal{H}^i = \langle Q^i,\ Q^i_0,\ \Sigma^i,\ X^i,\ \triangle^i,\ I^i,\ F^i,\ V^i_0\rangle \mid i \in I, \forall i, j \in I, i \neq j.X_i \cap X_j = \emptyset\}$. Their product is $\bigotimes_{\mathcal{H}_i \in C} \mathcal{H}_i = \langle Q,\ Q_0,\ \Sigma,\ X,\ \triangle,\ I,\ F,\ V_0\rangle$ where:

$$Q = Q^1 \times \cdots \times Q^n \quad Q_0 = Q^1_0 \times \cdots \times Q^n_0 \quad \Sigma = \Sigma^1 \cup \ldots \cup \Sigma^n \quad X = X^1 \cup \ldots \cup X^n$$

$$I(q_1,\ \ldots,\ q_n) = \bigwedge_{i=1}^{n} I^i(q_i) \quad F(q_1,\ \ldots,\ q_n)(x) = F^i(q)(x) \text{ if } x \in X^i \quad V_0 = \bigwedge_{i=1}^{n} V^i_0$$

$$\triangle \subseteq (Q \times \text{pred}(X) \times \Sigma \times \text{pred}(X \cup X') \times Q) \text{ where given } Z \subseteq I, a \in \Sigma :$$

$$((q_1,\ \ldots,\ q_n),\ \textstyle\bigwedge_{z \in Z} g_z,\ a,\ \bigwedge_{z \in Z} j_z,\ (q'_1,\ \ldots,\ q'_n)) \in \triangle \text{ iff } \forall z \in Z.(q_z,\ g_z,\ a,\ j_z,\ q'_z) \in \triangle^i \text{ and } \forall j \in I \setminus Z.q_j = q'_j$$

The states of the product are composed by the product of the states of its components. Similarly, the alphabet and the variables are the union of those of its components. The invariants, flow function and initial valuations are defined homomorphically on their elements. Finally, the transitions are synchronous, i.e. all the components (satisfying the constraints on the corresponding transition) synchronise on an action a while the others stay idle (in the following we will also distinguish between input and output actions through broadcast channels).

3. UPPAAL SMC

Uppaal is a toolbox that has been adopted for verifying real-time systems, represented by (extended) timed automata, that interact through broadcast channels and shared variables. Uppaal SMC is an extension of Uppaal that allows to express both stochastic and non-linear dynamic features, by adopting a stochastic extension of hybrid automata. The stochastic interpretation replaces the non-deterministic choices for multiple enabled transitions and time delays with, respectively, probabilistic choices and probability distributions (uniform for bounded time and exponential for unbounded time). By composing different automata through the product in Definition 2, arbitrary complex behaviours can be obtained, where it is possible to statically or dynamically generate new instances of automata, that are uniquely identified.

Uppaal SMC uses *Statistical Model Checking* (SMC) [22] to evaluate probabilistic properties of interest. SMC uses results from Statistics to decide, based on a given number of monitored simulations, whether the system under analysis satisfies the property of interest within a given degree of confidence. An advantage of SMC is that it avoids the exploration of the whole state-space of a model, which is a main drawback of standard model checking techniques.

Temporal Logic formulae.

In addition to standard model checking techniques of properties as reachability, deadlock-freedom, in Uppaal SMC it is possible to evaluate the probability that a random run of a network M satisfies a property φ, that is defined using the Metric Interval Temporal Logic (MITL) [12] in a given amount of time t.

Definition 3. A MITL formula φ is inductively defined by the following grammar:

$$\varphi ::= \mathtt{ap} \mid \neg\varphi \mid \varphi_1 \wedge \varphi_2 \mid \bigcirc\varphi \mid \varphi_1 \cup^{x}_{\leq t} \varphi_2$$

In the definition above, $\mathtt{ap}$ are atomic predicates over states of an automaton, and the logical operators are standard, except for $\varphi_1 \cup^{x}_{\leq t} \varphi_2$ that checks whether a formula φ_1 is satisfied in a run *until* a formula φ_2 is satisfied, and this must happen before the clock x exceeds the value t. As usual, it is possible to derive the operators *exists* and *forall* as $\Diamond_{x\leq t}\varphi = \mathtt{true} \bigcup^{x}_{\leq t} \varphi$ and $\Box_{x\leq t}\varphi = \neg\Diamond_{x\leq t}\neg\varphi$, where both quantifiers are bounded by the time t for the clock x.

Checking if a model M satisfies a property $\mathbb{P}_M(\Diamond_{x\leq t}\varphi) \geq p$, $p \in [0,1]$ is undecidable in general [17]. Statistical algorithms are developed in Uppaal SMC for estimating the probability of cost-bounded reachability problems in a given interval of confidence. There are three types of queries: $\mathbb{P}_M(\Diamond_{x\leq t}\mathtt{ap})$ (probability estimation), $\mathbb{P}_M(\Diamond_{x\leq t}\mathtt{ap}) \geq p, p \in [0,1]$ (hypothesis testing), $\mathbb{P}_M(\Diamond_{x_1\leq t_1}\mathtt{ap}_1) \geq \mathbb{P}_M(\Diamond_{x_2\leq t_2}\mathtt{ap}_2)$ (probability comparison).

In the following, we introduce the case study that will be modelled as a network of stochastic hybrid automaton, where reliability and energy consumption indicators will be evaluated through MITL formulae using Uppaal SMC. We will also validate the correctness of the overall model.

4. MODELLING THE CASE STUDY

We propose a cyber-physical system from the railway domain as case study: a rail road switch heating system. A rail road switch is a mechanism enabling trains to be guided from one track to another. It works with a pair of linked tapering rails, known as points. These points can be moved laterally into different positions, in order to direct a train into the straight path or the diverging path. Such switches are therefore critical components in the railway domain, since reliability of the railway transportation system highly depends on their correct operation.

During winter, snow and ice can prevent the switches to work properly, hence heaters are used so that the temperature of the rail road switches can be kept above freezing. The devices considered in this paper are composed by a series of tubular flat heaters along the rail road tracks, which warm up the rail road by induction heating. To accomplish its task, a rail road switch heater system reads through sensors the temperatures of the air and of the rail road [4].

The management of the heaters is automatic, and is remotely controlled by a central computational unit. The central unit manages the policy of energy consumption of the system, so to ensure the overall reliability.

Rail road switch heaters may be affected by different weather conditions according to their displacement. Different policies may be adopted to power the heaters (by electricity), as for example to heat a selection of switches for a given amount of time or to heat all the switches together.

4.1 Stochastic Hybrid Model

The selected case study has been modelled through stochastic hybrid automata. Indeed, this formalism allows to capture both discrete, continuous and stochastic aspects in a single framework. We have been able to verify the correctness of the interactions, as well as energy and reliability indicators, by using the Uppaal SMC toolkit.

We briefly outline the formalization of the system of (remotely controlled) rail road switch heaters as a product of hybrid automata. We will adopt a dynamic power management policy (DPV), where an on-off mechanism will be used for heating the switches according to parametric thresholds representing the temperatures triggering the activation/deactivation of the heating. The rail road switch heaters whose temperature cannot be kept above the freezing threshold will experience a failure.

Continuous aspects.

The continuous physical behaviour concerning the increment and decrement of the temperature of the rail road track, respectively when the heater is turned on or off, is modelled by an ODE representing the balance of energy [4]. Assuming that the values of the temperature of the surrounding area T_e and the previous internal temperature T are known, the flow function $F(T)$ (see Definition 1) that updates the internal temperature T after time t is (we adopt the Newton's notation for differentiation):

$$\mathtt{T}' = \frac{-\mathtt{uA}(\mathtt{T} - \mathtt{T_e}) + \mathtt{Q}}{\mathtt{mc}}$$

where u is the coefficient of convective exchange; c, the heat capacity of iron; A, the surface area exposed to the external temperature; m, the mass of the iron bar; $\dot{Q}$, the power used when the heater is turned on, if the heater is turned off this value will be zero. This ODE is expressed in the stochastic hybrid model $\mathtt{H}$ in Figure 1, where the temperature T is a *continuous clock* and the flow function F is similar in different states. Indeed, when $\mathtt{H}$ is in state $\mathtt{on}$, F is adding the term $\mathtt{Q}$ (i.e. the power); which is not the case in states `off` and `ready`.

Discrete aspects.

The two main logical components describing the discrete cyber part of the analysed system are the *heater* $\mathtt{H}$ (depicted in Figure 1) and the *central coordinator* $\mathtt{K}$ (depicted in Figure 2). The central coordinator manages the activation/deactivation of each heater. The network of n heaters is realised by replicating the stochastic hybrid automaton $\mathtt{H_{id}}$, $id \in 1,\dots,n$, where each heater is uniquely identified by its *id*. The overall network $\mathtt{N}$ is obtained through the product of Definition 2:

$$\mathtt{N} = \Big(\bigotimes_{id\in 1,\dots,n} \mathtt{H_{id}}\Big) \otimes \mathtt{K}. \tag{1}$$

Interactions. We now discuss the interactions among the components. The coordinator sends messages to the network of heaters through an array of channels $\mathtt{NI}[\mathtt{id}]$ indexed by the identifiers of the heaters, to activate them. Note that Uppaal SMC only allows broadcast channels, hence an array of channels has been adopted in order to implement one-to-one communications. Following the standard notation, sending through a channel `a` is denoted as `a`!, while reading as `a`?. Upon reception of the notification $\mathtt{NI}[\mathtt{id}]?$, the heater with identifier `id` switches from state `ready` to state `on`.

The heaters communicate to the coordinator their transitions from `off` to `ready` through the channel `ins` (see transition 2 below), so asking for being activated, and their transition from `on` to `off` through the channel `rem` (see transition 3 below); both channels are many-to-one. All channels are *urgent*, which means that no delays will occur in case a synchronization is available.

While the coordinator is in a `busy` state, a shared variable `lock` is used as a semaphore to prevent a heater from sending messages that cannot be elaborated, and it is used by the heaters for communicating their identifiers to the coordinator.

We now discuss in detail the two main components of the analysed system.

Heater. The heater model is depicted in Figure 1 and it implements the policy for activating and deactivating the heating. The heater model has four main states: `on`, `off`, `ready` and `fail`. The dynamic power management policy is based on two threshold temperatures:

Warning threshold (T_{wa}): this threshold represents the lower temperature that the track should not exceed. If the temperature is lower than T_{wa}, then the risk of ice or snow can lead to a failure of the rail road switch and therefore the heating system needs to be activated. Indeed, let $\triangle_{\mathtt{H}}$ be the set of transitions of $\mathtt{H}$, then we have

$$(\mathtt{off}, ((\mathtt{T_{wa}} > \mathtt{T})\&\&(\mathtt{lock} < 0)), \mathtt{ins}!, \mathtt{lock} := \mathtt{id}, \mathtt{ready}) \in \triangle_{\mathtt{H}} \tag{2}$$

i.e. the guard $g = T_{wa} > T$ checks if the warning threshold has been exceeded;

Working threshold (T_{wo}): this is the working temperature of the heating system. Once this temperature is reached, the heating system can be safely turned off in order to avoid an excessive waste of energy. Indeed, we have

$$(\mathtt{on}, ((\mathtt{T} > \mathtt{T_{wo}})\&\&(\mathtt{lock} < 0)), \mathtt{rem}!, \mathtt{lock} := \mathtt{id}, \mathtt{off}) \in \triangle_{\mathtt{H}} \tag{3}$$

the guard $\mathtt{T} > \mathtt{T_{wo}}$ checks if the working threshold has been exceeded.

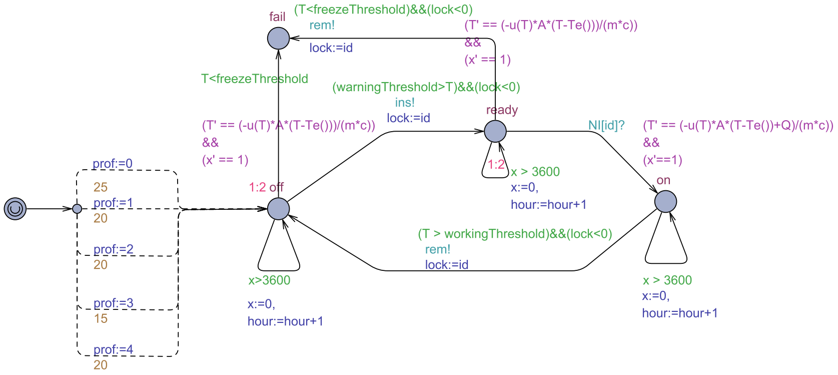


Figure 1: The stochastic hybrid automaton $\mathtt{H}$, modelling an instance of a rail road switch heater

Coordinator. The coordinator is modelled as the hybrid automaton $\mathtt{K}$ in Figure 2. It collects the requests of activation from the pending heaters, and it manages the energy supply according to a FIFO order, i.e. the first heater that asks to be turned on will be the first to be activated. The maximum number of heaters that can be turned on at the same time is called NH_{max}. This value represents the maximum amount of energy deliverable by the system, and cannot be exceeded. If there is no energy available, each request will be enqueued in the queue of pending heaters.

The states $Q_{\mathtt{K}}$ of $\mathtt{K}$ are: $Q_{\mathtt{K}} = \{\mathtt{available}, \mathtt{busy1}, \mathtt{busy2}, \mathtt{q_0}\}$. A variable `active` is used for counting the number of activated heaters. We have

$$\forall q \in Q_{\mathtt{K}}.I(q) = (\mathtt{active} \leq \mathtt{NHmax}\ \&\&\ \mathtt{active} \geq 0)$$

i.e. the invariant must hold in all states. Indeed, the number of active heaters must be non-negative and less or equal to the number of maximum heaters that can be activated.

The queue of pending heaters is modelled with the array $\mathtt{queue}[\,]$, and the functions $\mathtt{enqueue}(\mathtt{int\ id})$ and $\mathtt{dequeue}()$ are used for inserting and removing elements, while $\mathtt{empty}()$ returns true if the queue of pending heaters is empty. In state `available` the coordinator is waiting for a message from one of the heaters in the network. There are two inner cycles:

$$(\mathtt{available},\ \mathtt{active} == \mathtt{NHmax},\ \mathtt{ins}?,$$

$$(\mathtt{enqueue}(\mathtt{lock}); \mathtt{lock} = -1),\ \mathtt{available}) \in \triangle_{\mathtt{K}}$$

where a heater asks to be activated but there is no energy available, which is rendered by the guard $\mathtt{active} == \mathtt{NHmax}$. This request is inserted in the queue of pending requests. The second cycle is:

$$(\mathtt{available},\ \mathtt{empty}(),\ \mathtt{rem}?,$$

$$(\mathtt{active}\text{- -}, \mathtt{lock} = -1),\ \mathtt{available}) \in \triangle_{\mathtt{K}}$$

In the second cycle a heater asks to be deactivated and there are no other heaters that are waiting for being activated (guard $\mathtt{empty}()$). In the state `busy1` (see Figure 2) the coordinator activates the heater that is asking. In the state `busy2` the coordinator deactivates the heater whose asking and activates one of the pending heaters, according to a FIFO order.

Stochastic aspects.

There are two stochastic aspects involved in our model: the failure of a switch and the weather forecast. Concerning the first one, in Figure 1 the transitions from state `off` and `ready` to `fail` represent the failure of a component. For firing these transitions, the temperature must be below the freezing temperature (guard $\mathtt{T} < \mathtt{freezeThreshold}$). When the guard is satisfied, as usual, the transition fires within time that follows an exponential distribution, that this the more the temperature is below zero the more probable

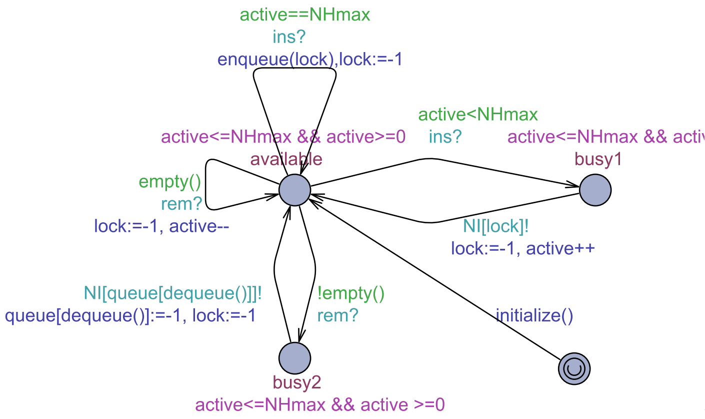


Figure 2: The stochastic hybrid automaton K, modelling the coordinator

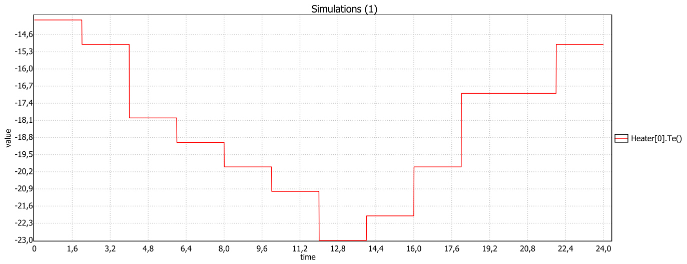
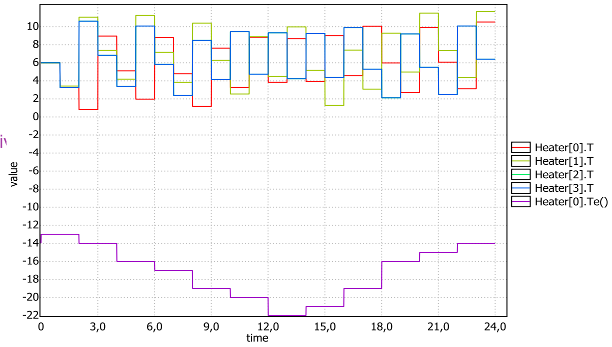


Figure 3: A weather profile corresponding to the temperatures for coldest winter nights in the city Montreal, retrieved from [25]. The simulation starts at 6:pm and ends after 24 hours.

will be that the corresponding heater fails. Accordingly, the model can spend an unbounded amount of time before firing these transitions. When the transition is fired from state ready, the failing heater is removed from the queue of pending heaters (for example through a failure detector).

The second stochastic aspect concerns the weather forecast. In this case, we have based the external temperatures on those days for which the analysis is relevant (i.e. winter days in northern cities), as depicted in Figure 3. For our experiments, we have five different daily weather profiles retrieved from the internet [25]. The time window under analysis is divided into intervals to which an average reference temperature is assigned. The current instance of the model concentrates on a whole day, divided into intervals of two hours. A data structure has been used to store the weather data. As shown in Figure 1, from the initial state of H we have different urgent (i.e. instantaneous) probabilistic transitions to the state off. Indeed, a probability is assigned to each weather profile (based on weather statistics) which is the probability to fire the corresponding transition. For example, the weather profile with index 0 is selected with a probability of 0.25. During a simulation, the current time is stored in the clock x, and a variable hour stores the current hour. Note that x is updated according to $\mathtt{x}' == 1$. The function Te() used in the flow function of T, selects the actual external temperature (corresponding to the selected weather profile) based on the current hour.

5. EVALUATION AND VALIDATION OF THE MODEL



(a) discretization step=1hour

Figure 4: A simulation showing the track temperatures trend of the heaters and the environment temperature Te of $\mathtt{H}_0$, where $NH_{max} = 75\%$, T_{wa}=4°C, T_{wo}=6°C and $n = 4$.

In this section we describe the results of the evaluations we have performed to analyse the reliability and energy consumption according to different policies. We will also certify communication safety in the proposed model, by verifying the absence of deadlocks. The network under analysis will be composed of four heaters and the coordinator, i.e. $n = 4$ (see equation 1). We assumed as freezing temperature -2°C. In our experiments, we consider a level of available power NH_{max} sufficient for heating 75% of the overall heaters in the network. We analyse different values of warning threshold (T_{wa}) and working threshold (T_{wo}). In all the considered evaluations, we assume that at starting time the system is in a safe condition, that is the internal temperature of all switches is equal to its working temperature. This assumption is useful for avoiding instantaneous failure.

Measures of Interest.

We consider two measures of interest, that will be expressed as properties in MITL. The first concerns the energy consumption while the second addresses the reliability of the system under analysis, defined as the probability that no failure occurs in the interval of time under analysis [24], that is $1 - PFAIL(t,l)$.

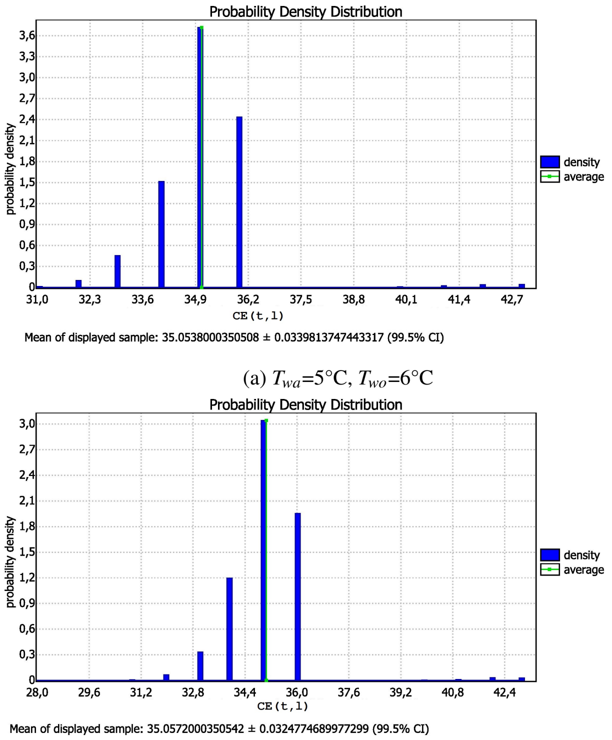
1 $CE(t,l)$: the average time (in hours) the heaters are activated in the time interval $[t, t+l]$. By multiplying $CE(t,l)$ for the power consumed (kilowatt per hour) it is possible to derive the energy consumed by the system;

2 $PFAIL(t,l)$: the probability that at least a switch fails (becomes frozen) at time $t+l$, given that at time t is not failed.

In the following, the considered interval of time $[t, t+l]$ will be $t = 6pm$ and $l = 24h$. The temperature values during a simulation are depicted in Figure 4. We assume that the coordinator performs a check on the temperatures each hour to eventually synchronize with the network of heaters, i.e. will consider as discretization step 1h (see Figure 4).

Evaluation.

We now evaluate $CE(t,l)$ and $PFAIL(t,l)$ through formulae of MITL (see Definition 3) defined in Uppaal SMC, enriched with quantification operators on the replicated models and expected values. We will consider a discrete clock energy that counts the hours H spends in state on. For enhancing readability, we have omitted this clock in Figure 1. For the energy consumption we estimate the



(a) T_{wa}=5°C, T_{wo}=6°C

(b) T_{wa}=4°C, T_{wo}=7°C

Figure 5: The probability density distribution of $CE(t,l)$ for two different policies of energy consumption

number of hours in which the heaters are active as:

$$CE(t,l) \overset{\text{def}}{=} \mathtt{E}[<=24;10000]\,(\mathtt{max}:\sum_{\mathtt{i:id_t}} \mathtt{H_i.energy})$$

where as usual E stands for the expected value, 24 is the considered interval of time (24h) and 10000 are the simulations executed by the tool, and the overall energy consumption is the sum for all $\mathtt{H_i}$ of all the clocks energy.

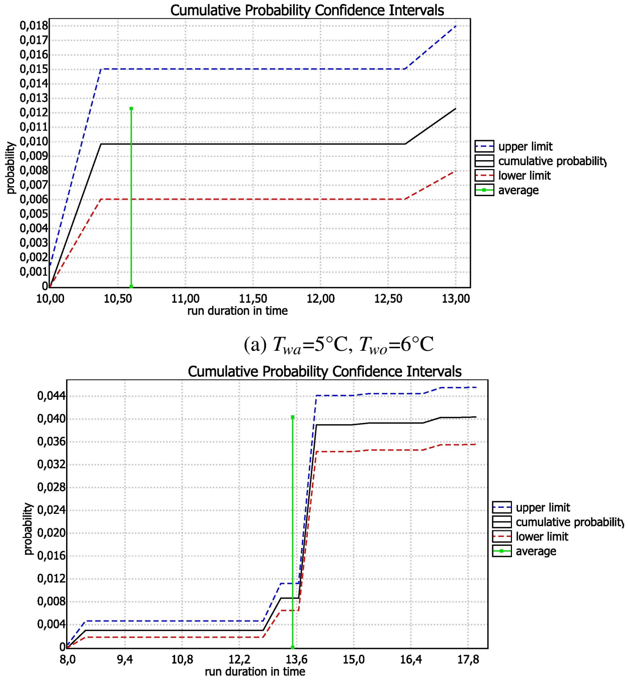
The probability of failure is estimated by Uppaal SMC as:

$$PFAIL(t,l) \overset{\text{def}}{=} \mathbb{P}(\Diamond_{\mathtt{h}\leq 24}\exists(\mathtt{i:id_t})(\mathtt{H_i.fail}))$$

the above formula evaluates the probability that in the time interval $[t, t+l]$ (24h) there exists at least a switch $\mathtt{H_i}$ in the network which has failed, i.e. $\mathtt{H_i}$ is in state fail. In this case the number of simulations to be performed depend on the given degree of confidence and are not fixed.

We now show how the proposed model can be used for evaluating different policies of energy consumption in order to improve the energy consumption and reliability level of the analysed system. We have evaluated the properties of interest for two policies of energy consumption that have shown optimal trade-off between energy consumption and reliability, that are: (1) (T_{wa},T_{wo})=(4°C, 7°C) and (2) (T_{wa},T_{wo})=(5°C, 6°C). In particular, we compare a "wide" gap between the two thresholds and a "tight" one. The results of the experiments are depicted in Figure 5 and Figure 6. The probability density distributions of $CE(t,l)$ for the two simulations are in Figure 5a ((T_{wa},T_{wo})=(5°C, 6°C)) and Figure 5b ((T_{wa},T_{wo})=(4°C, 7°C)), where on the x and y axis we have respectively the values for $CE(t,l)$ and their probability density.

The mean values computed for $CE(t,l)$ are $CE(t,l) \sim 35.0572 \pm 0.0324775$ for the pair of thresholds (1) and $CE(t,l) \sim 35.0538 \pm 0.0339814$ for the pair of thresholds (2). We note that the values of $CE(t,l)$ are affected by the corresponding values of $PFAIL(t,l)$, indeed if an heater fails it will no longer consume energy. For the measure of interest $PFAIL(t,l)$ the cumulative probability for the simulations with the two pair of thresholds are in Figure 6a and Figure 6b, together with its upper and lower bound given by the confidence interval (99.5% in these simulations). On the x axis the duration time of the experiment is showed while on the y axis the corresponding values of $PFAIL(t,l)$ are showed (note that the cumulative probability is displayed).



(a) T_{wa}=5°C, T_{wo}=6°C

(b) T_{wa}=4°C, T_{wo}=7°C

Figure 6: The cumulative probability confidence intervals of $PFAIL(t,l)$ for two different policies of energy consumption

From the results of the experiments we observe that with the "tight" policy we are able to reduce $PFAIL(t,l)$, i.e. $PFAIL(t,l) \in [0.035567, 0.045666]$ for (1), while for the "wide" policy (i.e. the pair of thresholds (2)) the values of $PFAIL(t,l)$are in [0.00799584, 0.0179953].

Intuitively, when the gap is "tight" there is a better distribution of energy supply among all the heaters, that are active for lesser time than when using a "wide" one, which in turns imply an improved reliability.

Validation.

After a model of the analysed system has been designed, it is paramount to validate it for ensuring that the obtained results are trustworthy. Indeed, unwanted communication mismatches could lead the overall simulation into an erroneous state (for example a deadlock). Basing on the outcome of the experiments, it is not always easy to detect a possible misbehaviour of the model. For example, if the message notifying the activation of a heater H is not retrieved, then H is prevented from being activated and it will experience a failure. A degradation in the overall reliability of the system would be then experienced, and the adopted policy of energy consumption could be wrongly detected as responsible. In order to prevent this unpleasant situation, we will verify the absence of deadlocks in the adopted model through the standard branching-time temporal logic provided by Uppaal. Temporal logic formulae cannot be model-checked for hybrid systems with continuous clocks used in guards, as it is in our case. We overcome this problem by assuming that the guards are satisfied non-deterministically, obtaining an over approximation of the real behaviour of the system. Note that if this approximation is deadlock-free, then it will be for the underlying hybrid system, because all its traces are model-checked.

Moreover, we removed the state `fail` from the model to exclude the trivial deadlock due to the failure of all components. The absence of deadlocks in `N` is certified through the formula

```
A□ not deadlock
```

which states that in all possible paths (`A`), for all states (□) the system progresses (`not deadlock`, which is a special formula in Uppaal). The formula holds in the model.

Experiments Performance.

We now report on the memory usage and performances of the tool for computing the measures of interest and for certifying the absence of deadlocks. It has been used a machine with CPU Intel Core i5-4570 at 3.20 GHZ with 8 GB of RAM, running 64-bit Windows 10, and Uppaal SMC academic version 4.1.19.

The elapsed times for computing $CE(t,l)$ and $PFAIL(t,l)$ are respectively 12.188s and 4.61s, and the resident/virtual memory usages peaks are 8.428MB / 28.732MB and 8.068MB / 28.136MB. We remark that Uppaal SMC computed 10000 and $\approx$ 4000 simulations for evaluating respectively $CE(t,l)$ and $PFAIL(t,l)$.

The absence of deadlocks has been certified with elapsed time of 0.672s and resident/virtual memory usage peaks 10.788MB / 32.536MB.

6. RELATED WORK

A vast literature concerns analysing and optimising the energy consumption in disparate application domains using formal methods. Concerning the rail road switch heating system, reliability and energy consumption indicators have been analysed in [4, 3], by designing and evaluating Stochastic Activity Networks models [23]. The measures of interest have been defined using Markov Reward Models [15] and evaluated through the Möbius tool [8]. Compared to the above works, here we uniformly deal with both continuous, stochastic and discrete aspects of the analysed system by using Statistical Model Checking of hybrid automata. Moreover, we have proved the correctness of the proposed model through standard model checking techniques. It would be interesting to extend our analysis to introduce a prioritised approach and adaptive thresholds, as it has been done respectively in [4, 5], so to better distribute the energy supply among the most critical devices (the switches in the main tracks) and less relevant switches (i.e. those located in side tracks). A flexible approach could be introduced to adapt the policies of energy consumption to the different temperature profiles during a day [5].

Optimizing energy consumption for energy aware buildings, represented as stochastic hybrid automata, is the selected case study in [10]. The model is parametrized by several cost values (e.g. rooms, building, heaters, weather) that need to be optimized in order to minimize energy consumption. Statistical Model Checking and analysis of variance has been used to identify Pareto-optimal configurations in terms of both discomfort (i.e. distance between the desired range of temperature and the current temperature) and energy consumption. We are planning to identify the optimal values of our parameters (i.e. temperature thresholds) by resorting to similar techniques.

Dynamic Power Management (DPV) and Dynamic Voltage and Frequency Scaling (DVFS) are adopted in [1] to reduce the energy consumption in multiprocessor dataflow applications by using Statistical Model Checking. DPM reduces the energy consumption of processors while they are idle, and DVFS reduces the energy consumption by lowering the voltage and clock frequency. The system is modelled as Stochastic Hybrid Games, where uncontrollable actions represent the exact finishing time of a task. The tool Uppaal Stratego [11] is used to synthesise the safe and near optimal strategy, that is the scheduling of tasks to processors that maximise the throughput and minimise the energy consumption. Similarly, we adopted Statistical Model Checking and DPM, to turn off the energy consumption of heaters when a given temperature is reached.

Hybrid automata have been used in [14] to study the dynamic power management control problem, where the discrete state represents the power mode of the system, and the continuous one represents the consumed power. Two strategies are compared: on demand wake-up of a component (that was previously turned off) and pre-emptive wake-up. The former provides better results for the conservation of energy and prevention of latency. It would be interesting to implement in our work a power adjustment mechanism.

7. CONCLUSION

We have addressed the quantitative assessment of reliability and energy consumption indicators of a rail road switch heating system, which is a critical system in the railway domain. Through the proposed analysis, it is possible to select among several policies of energy consumption the one that guarantees the best trade-off between energy consumption and reliability.

The policies of energy consumption are based on threshold temperatures triggering the activation/deactivation of the heating system. A central coordinator interacts with the network of heaters and manages a first-come, first-served energy supply. The system reads in real time both the environment temperature and the track temperature, and according to these thresholds decides when to turn on and off the heaters.

We have adopted recently introduced techniques, proving their effectiveness and usability in the context of a representative cyber-physical system. Specifically, Statistical Model Checking was selected, to accomplish the twofold objective of quantitatively assessing satisfaction of reliability and consumption properties and of validating the developed model itself against the relevant deadlock-free property.

In particular, to deal with both the continuous characteristics of the analysed system (i.e. convective heat exchange) and the stochastic nature of the involved phenomena (i.e. weather forecast, failures), the stochastic hybrid automata formalism and the Uppaal SMC toolkit have been used.

The reliability and energy consumption of the overall system have been defined using Metric Interval Temporal Logic formulae and evaluated through Statistical Model Checking. The obtained results have been certified by formally verifying the correctness of crucial aspects of the proposed model. In particular, the absence of deadlocks has been verified through a standard branching-time temporal logic formula. A main outcome of the proposed model is the possibility of comparing different policies of energy consumption in order to select the one with a suitable trade-off between reliability and energy consumption.

Extensions of this work are planned in several directions. General guidelines to model and evaluate energy-saving cyber-physical systems have been discussed by the authors in [6] and used in [4, 5]. We are planning to explicitly relate the methodology proposed in this paper to those guidelines. In particular, for comparing the scalability and other aspects of the proposed approach with the models in [4, 5], we will consider priorities for the heaters and a system composed of a larger number of components. Preliminary results have shown the scalability of the proposed approach. This methodology will also be applied to other energy-saving cyber-physical systems belonging to the railway domain, (e.g. self-powered trains, lighting of stations, regenerative braking). Finally, concerning the energy reduction of transducers (i.e. interactions among sensors, actuators), a novel formalism introduced by the authors for modelling adaptable communication-based applications will be extended to analyse non-functional aspects [7].

8. REFERENCES

[1] Ahmad, W., v. d. Pol, J.: Synthesizing energy-optimal controllers for multiprocessor dataflow applications with uppaal stratego. In: 7TH International Symposium on Leveraging Applications of Formal Methods, Verification and Validation, ISOLA 2016 (2016)

[2] Alur, R., Dill, D.L.: A theory of timed automata. Theoretical Computer Science (TCS) 126 1994 126(2), 183 – 235

[3] Basile, D., Chiaradonna, S., Di Giandomenico, F., Gnesi, S., Mazzanti, F.: Stochastic model-based analysis of energy consumption in a rail road switch heating system. In: Fantechi, A., Pelliccione, P. (eds.) Software Engineering for Resilient Systems: 7th International Workshop, SERENE 2015, Paris, France, September 7-8, 2015. Proceedings. pp. 82–98. Springer International Publishing, Cham (2015)

[4] Basile, D., Chiaradonna, S., Giandomenico, F.D., Gnesi, S.: A stochastic model-based approach to analyse reliable energy-saving rail road switch heating systems. Journal of Rail Transport Planning & Management 6(2), 163 – 181 (2016), http://www.sciencedirect.com/science/article/pii/S2210970616300051

[5] Basile, D., Di Giandomenico, F., Gnesi, S.: Tuning energy consumption strategies in the railway domain: a model-based approach. In: 7TH International Symposium on Leveraging Applications of Formal Methods, Verification and Validation, ISOLA 2016 (2016)

[6] Basile, D., Di Giandomenico, F., Gnesi, S.: Model-based evaluation of energy saving systems. In: Kharchenko, V., Kondratenko, Y., Kacprzyk, J. (eds.) Green IT Engineering: Concepts, Models, Complex Systems Architectures. pp. 187–208. Springer International Publishing (2017)

[7] Basile, D., Di Giandomenico, F., Gnesi, S., Degano, P., Ferrari, G.L.: Specifying variability in service contracts. In: Proceedings of the 11th International Workshop on Variability Modelling of Software-intensive Systems (Vamos), February 1 - 3 (2017), (to appear)

[8] Clark, G., Courtney, T., Daly, D., Deavours, D., Derisavi, S., Doyle, J.M., Sanders, W.H., Webster, P.: The möbius modeling tool. In: 9th Int. Workshop on Petri Nets and Performance Models (PNPM). pp. 241–250 (2001)

[9] Clarke, Jr., E.M., Grumberg, O., Peled, D.A.: Model Checking. MIT Press, Cambridge, MA, USA (1999)

[10] David, A., Du, D., Guldstrand Larsen, K., Legay, A., Mikučionis, M.: In Proceedings of the th NASA Formal Methods Symposium (NFM) 2013., chap. Optimizing Control Strategy Using Statistical Model Checking, pp. 352–367. Springer Berlin Heidelberg (2013)

[11] David, A., Jensen, P.G., Larsen, K.G., Mikučionis, M., Taankvist, J.H.: Uppaal stratego. In: Baier, C., Tinelli, C. (eds.) Tools and Algorithms for the Construction and Analysis of Systems: 21st International Conference, TACAS 2015, Held as Part of the European Joint Conferences on Theory and Practice of Software, ETAPS 2015, London, UK, April 11-18, 2015, Proceedings. pp. 206–211. Springer Berlin Heidelberg, Berlin, Heidelberg (2015)

[12] David, A., Larsen, K.G., Legay, A., Mikuăionis, M., Poulsen, D.B.: Uppaal smc tutorial. International Journal on Software Tools for Technology Transfer 17 (2015)

[13] David, R., Alla, H.: On hybrid petri nets. Discrete Event Dynamic Systems 11(1-2), 9–40 (2001)

[14] Erbes, T., Shukla, S.K., Kachroo, P.: Stochastic learning feedback hybrid automata for dynamic power management in embedded systems. In: IEEE Mid-Summer Workshop on Soft Computing in Industrial Applications (2005)

[15] Haverkort, B.R.: Lectures on formal methods and performance analysis. chap. Markovian Models for Performance and Dependability Evaluation, pp. 38–83. Springer-Verlag (2002)

[16] Henzinger, T.A.: The theory of hybrid automata. pp. 278–. Eleventh Annual IEEE Symposium on. Logic in Computer Science, IEEE C.S. (1996)

[17] Henzinger, T.A., Ho, P.H., Wong-Toi, H.: Algorithmic analysis of nonlinear hybrid systems. IEEE Transactions on Automatic Control 43(4), 540–554 (1998)

[18] Júlvez, J., Mahulea, C., Vázquez, C.R.: Simhpn: A matlab toolbox for simulation, analysis and design with hybrid petri nets. In: Nonlinear Analysis: Hybrid Systems. vol. 6, pp. 806–817 (2012)

[19] Krishna, S.N., Trivedi, A.: Hybrid automata for formal modeling and verification of cyber-physical systems. CoRR abs/1503.04928 (2015), http://arxiv.org/abs/1503.04928

[20] Larsen, K.G., Pettersson, P., Yi, W.: Uppaal in a nutshell. In Springer International Journal of Software Tools for Technology Transfer 1(1+2), 134–152 (May 1997)

[21] Lee, E.A.: Cyber physical systems: Design challenges. In: Proceedings of the 2008 11th IEEE Symposium on Object Oriented Real-Time Distributed Computing. pp. 363–369. ISORC '08, IEEE Computer Society, Washington, DC, USA (2008), http://dx.doi.org/10.1109/ISORC.2008.25

[22] Legay, A., Delahaye, B., Bensalem, S.: Statistical Model Checking: An Overview, pp. 122–135. Springer Berlin Heidelberg, Berlin, Heidelberg (2010)

[23] Sanders, W.H., Meyer, J.F.: Stochastic activity networks: Formal definitions and concepts. In: Lectures on Formal Methods and Performance Analysis (2000)

[24] Trivedi, K.S.: Probability and Statistics with Reliability, Queuing and Computer Science Applications. John Wiley and Sons Ltd., Chichester, UK, 2nd edition edn. (2002)

[25] https://weatherspark.com/#!graphs;ws=27985, (accessed on March 2016)