Approximate model checking of stochastic hybrid systems

A. Abate[*], J.-P. Katoen[†], J. Lygeros[‡], and M. Prandini[§]

Abstract

A method for approximate model checking of stochastic hybrid systems with provable approximation guarantees is proposed. We focus on the probabilistic invariance problem for discrete time stochastic hybrid systems and propose a two-step scheme. The stochastic hybrid system is first approximated by a finite state Markov chain. The approximating chain is then model checked for probabilistic invariance. Under certain regularity conditions on the transition and reset kernels governing the dynamics of the stochastic hybrid system, the invariance probability computed using the approximating Markov chain is shown to converge to the invariance probability of the original stochastic hybrid system, as the grid used in the approximation gets finer. A bound on the convergence rate is also provided. The performance of the two-step approximate model checking procedure is assessed on a case study of a multi-room heating system.

1 Introduction

Stochastic hybrid systems are a broad and widely applicable class of dynamical systems that involve the interaction of discrete, continuous, and probabilistic dynamics. Because of their generality, stochastic hybrid systems have found applications in many areas, including telecommunication networks, manufacturing systems, transportation, and biological systems (see, for example, [10, 13] for an overview). The importance of stochastic hybrid systems in applications has motivated a significant research effort into the foundations, analysis and control methods for this class of systems. Among the different problems addressed in this effort, of particular interest for the present paper are the problems of reachability and invariance, i.e., the characterization of the probability that the state of a stochastic hybrid system will reach (or, respectively, remain) in a specific region of the state space (see, for example, [2, 11, 12, 23, 31, 32]).

The development of computational tools is a crucial step in the application to practical problems of the new theoretical results that have emerged in the study of stochastic hybrid systems. Ideally such tools should be based on solid theoretical foundations, to quantify for instance the level of approximation introduced during the computation process. The tools should, however, also be versatile and efficient enough to be used on realistic applications. Many of the methods proposed in the area of stochastic hybrid systems for achieving this objective are based on numerical computations. These involve either imposing a grid on the state space, thus turning an infinite state problem into an approximate finite state one (see, for example, [1, 32] for reachability problems of the type considered here), or carrying out Monte-Carlo simulations to obtain empirical estimates of quantities such as expected values of reach probabilities (see [27] for an application of such methods). An alternative approach to the problem of verification of stochastic hybrid systems is based on satisfiability modulo theory, [19]. In [31], certain functions of the state are used to determine upper bounds on the invariance probability.

Even though computational tools based on numerical methods typically come with explicit approximation guarantees, their versatility and their computational requirements often limit their applicability to practical problems. To address a wider range of problems one would ideally like to combine numerical approximation

[*]Department of Aeronautics and Astronautics, Stanford University, U.S.A., and Delft Center for Systems and Control, TU Delft, The Netherlands, a.abate@tudelft.nl

[†]Department of Computer Science, RWTH Aachen University, Germany, katoen@cs.rwth-aachen.de

[‡]Automatic Control Laboratory, ETH Zurich, Switzerland, lygeros@control.ee.ethz.ch

[§]Dipartimento di Elettronica e Informazione, Politecnico di Milano, Italy, prandini@elet.polimi.it

with symbolic computation techniques that can be used to test a wider range of properties and that have been optimized for computational efficiency. Model checking is an interesting class of methods in this context. Model checking methods [6, 14] provide the means to algorithmically check whether a system satisfies a wide range of properties related to its evolution in time. In the context of reachability, model checking typically involves constructing forward/backward reachable sets based on a model of the system. More generally, model checkers can be employed to verify whether a model of the system satisfies various properties expressed in an appropriate temporal logic [29, 30].

A key difficulty in deploying model checking methods to hybrid systems is our ability to "compute" with sets, i.e., to represent sets of states and propagate them through the system dynamics. For finite state systems this is not an issue, at least conceptually. Storing and manipulating sets of states can be done either naively by enumeration, or in a more sophisticated way by using efficient representations such as binary decision diagrams; as a consequence, model checking tools for deterministic, discrete time, finite state systems have been available for many years [24] and have been successfully used in numerous applications. For systems whose state involves infinite or uncountable components it is sometimes possible to obtain an equivalent finite state representation on which finite state model checking methods can be applied. Even though several such classes of systems are known to exist in the deterministic setting [3, 4, 5, 21], very little is known in the stochastic context. Here model checking is typically limited to finite state Markov chains or decision processes, either in continuous or in discrete time [7, 15, 17, 20, 26].

The aim of the present paper is to take a first step toward combining numerical methods for approximate computation in stochastic hybrid systems with model checking methods developed to test temporal logic properties for finite state Markov chains. For the time being we concentrate on discrete time stochastic hybrid systems and finite time invariance specifications; current work focuses on extending the results to a wider range of properties of interest coded in the Probabilistic Computational Tree Logic (PCTL) [20]. The main idea is simple: given a stochastic hybrid system, we use numerical tools to generate a finite state Markov chain, together with guarantees on the level of approximation introduced in the process. The properties of the Markov chain (in our case the probability of remaining in a certain region of the state space) are then analyzed using a model checker. The result is combined with the approximation guarantees to provide an overall guarantee about the probability of satisfying the original property of interest for the stochastic hybrid system.

The material is organized in six sections. After this brief introduction, in Section 2 we describe the class of stochastic hybrid systems that we will consider, formally define the invariance problem of interest, and present some basic results on the characterization of its solution using a multiplicative cost function. Section 3 concentrates on numerical methods that can be used to approximately solve the invariance problem, with special emphasis on quantifying the error introduced in the process. Section 4 illustrates the model checking method and shows how it can be applied to (approximately) solve the original invariance problem. The approximate model checking approach is then applied to a benchmark problem in Section 5, and its computational complexity is quantitatively assessed. Finally, Section 6 outlines some further directions of investigation.

2 Probabilistic invariance for stochastic hybrid systems

2.1 Discrete time stochastic hybrid system modelling framework

We consider a discrete time stochastic hybrid system (DTSHS) model, inspired by [2]. The main difference with the model introduced in [2] is that here we consider autonomous systems, without control inputs. The state of a DTSHS comprises a discrete and a continuous component. The discrete component takes values in a finite set $\mathcal{Q}$; the elements of $\mathcal{Q}$ will be referred to as "modes." The continuous component of the state in each mode $q \in \mathcal{Q}$ lies in an Euclidean space $\mathbb{R}^{n(q)}$, whose dimension $n(q)$ is determined by a bounded map $n : \mathcal{Q} \to \mathbb{N}$. The hybrid state space is then given by the disjoint union of the Euclidean spaces associated to each mode, that is

$$\mathcal{S} := \bigcup_{q \in \mathcal{Q}} \{q\} \times \mathbb{R}^{n(q)}.$$

Let $\mathcal{B}(\mathcal{S})$ denote the σ-algebra generated by the subsets of $\mathcal{S}$ of the form $\cup_{q\in\mathcal{Q}}\{q\}\times A_q$, where $A_q \in \mathcal{B}(\mathbb{R}^{n(q)})$ is a Borel set in $\mathbb{R}^{n(q)}$. One can show [16] that $\mathcal{S}$ can be endowed with a metric that is equivalent to the usual Euclidean metric when restricted to each component $\mathbb{R}^{n(q)}$, making $(\mathcal{S}, \mathcal{B}(\mathcal{S}))$ a Borel space (i.e. homeomorphic to a Borel subset of a complete separable metric space).

The continuous state of a DTSHS evolves according to a probabilistic law specific to the current mode. A discrete transition from one mode to a different one may occur according to another probabilistic law; such a transition will cause a modification of the probabilistic law governing the continuous state dynamics. Furthermore, such a mode transition induces a probabilistic reset of the continuous state to a value in the Euclidean space associated with the new mode. The following definition formalizes this description.

Definition 1 (DTSHS) *A discrete time stochastic hybrid system is a collection $\mathcal{H} = (\mathcal{Q}, n, \mathrm{Init}, T_x, T_q, R)$, where*

- $\mathcal{Q} := \{q_1, q_2, \ldots, q_m\}$ *with $m \in \mathbb{N}$, represents the discrete state space;*
- $n : \mathcal{Q} \to \mathbb{N}$ *assigns to each discrete state $q \in \mathcal{Q}$ the dimension of the continuous state space $\mathbb{R}^{n(q)}$;*
- $\mathrm{Init} : \mathcal{B}(\mathcal{S}) \to [0,1]$ *is a probability measure on $\mathcal{S}$ for the initialization of the solution process;*
- $T_x : \mathcal{B}(\mathbb{R}^{n(\cdot)}) \times \mathcal{S} \to [0,1]$ *is a conditional stochastic kernel on $\mathbb{R}^{n(\cdot)}$ given $\mathcal{S}$. It assigns to each $s = (q,x) \in \mathcal{S}$ a probability measure, $T_x(\cdot|s)$, on the Borel space $(\mathbb{R}^{n(q)}, \mathcal{B}(\mathbb{R}^{n(q)}))$. The function $T_x(A|(q,\cdot))$ is assumed to be Borel measurable, for all $q \in \mathcal{Q}$ and all $A \in \mathcal{B}(\mathbb{R}^{n(q)})$;*
- $T_q : \mathcal{Q} \times \mathcal{S} \to [0,1]$ *is a conditional discrete stochastic kernel on $\mathcal{Q}$ given $\mathcal{S}$, which assigns to each $s \in \mathcal{S}$ a probability distribution, $T_q(\cdot|s)$, over $\mathcal{Q}$;*
- $R : \mathcal{B}(\mathbb{R}^{n(\cdot)}) \times \mathcal{S} \times \mathcal{Q} \to [0,1]$ *is a conditional stochastic kernel on $\mathbb{R}^{n(\cdot)}$ given $\mathcal{S} \times \mathcal{Q}$, that assigns to each $s \in \mathcal{S}$ and $q' \in \mathcal{Q}$, a probability measure, $R(\cdot|s,q')$, on the Borel space $(\mathbb{R}^{n(q')}, \mathcal{B}(\mathbb{R}^{n(q')}))$. The function $R(A|(q,\cdot),q')$ is assumed to be Borel measurable for all $q, q' \in \mathcal{Q}$ and all $A \in \mathcal{B}(\mathbb{R}^{n(q')})$.*

We consider the evolution of this model over a finite time horizon $[0, N]$ and define the semantics of the DTSHS algorithmically. In the sequel, we shall use boldface to denote random variables and normal typeset to denote sample values.

Definition 2 (Execution of a DTSHS) *Consider a DTSHS $\mathcal{H} = (\mathcal{Q}, n, \mathrm{Init}, T_x, T_q, R)$ and a time horizon $[0, N]$. A stochastic process $\{\mathbf{s}(k) = (\mathbf{q}(k), \mathbf{x}(k)),\ k \in [0,N]\}$ with values in $\mathcal{S} = \cup_{q\in\mathcal{Q}}\{q\} \times \mathbb{R}^{n(q)}$ is an execution of $\mathcal{H}$ if its sample paths are obtained according to the following algorithm:*

set $k = 0$;
extract a value $(q_k, x_k) \in \mathcal{S}$ for $(\mathbf{q}(k), \mathbf{x}(k))$ according to $\mathrm{Init}(\cdot)$;
while $k < N$ do
    extract a value $q_{k+1} \in \mathcal{Q}$ for $\mathbf{q}(k+1)$ according to $T_q(\cdot\,|(q_k, x_k))$;
    if $q_{k+1} = q_k$, then
        extract a value $x_{k+1} \in \mathbb{R}^{n(q_{k+1})}$ for $\mathbf{x}(k+1)$ according to $T_x(\cdot\,|(q_k, x_k))$;
    else
        extract a value $x_{k+1} \in \mathbb{R}^{n(q_{k+1})}$ for $\mathbf{x}(k+1)$ according to $R(\cdot\,|(q_k, x_k), q_{k+1})$;
    $k \to k+1$;
end.

By appropriately defining the discrete transition kernel T_q, it is possible to model situations such as "spontaneous jumps" (which are events that may occur during the continuous state evolution), as well as "forced jumps" (which are events that must occur, for instance when the continuous state exits some prescribed domain). For the spontaneous transitions, the fact that a discrete transition from q to $q' \neq q$ is enabled at $(q,x) \in \mathcal{S}$ can be encoded by the condition $T_q(q'|(q,x)) > 0$. For the forced transitions, one typically associates an "invariant set" $\mathrm{Inv}(q) \subseteq \mathbb{R}^{n(q)}$ with mode $q \in \mathcal{Q}$. The interpretation is that the discrete state can keep the

value q as long as the continuous state lies in the set $\mathrm{Inv}(q)$; when this condition is violated the discrete state will have to switch to a different value. This requirement can be expressed by setting $T_q(q|(q,x)) = 0$ for all $x \notin \mathrm{Inv}(q)$.

To simplify the notation, let us introduce a conditional stochastic kernel $T_s : \mathcal{B}(\mathcal{S}) \times \mathcal{S} \to [0,1]$ on $\mathcal{S}$, given $\mathcal{S}$, defined by

$$T_s(\{q'\} \times A_{q'} \,|(q,x)) = \begin{cases} T_x(A_{q'}|(q,x))T_q(q'|(q,x)), & \text{if } q' = q \\ R(A_{q'}|(q,x),q')T_q(q'|(q,x)), & \text{if } q' \neq q, \end{cases} \tag{1}$$

for all $A_{q'} \in \mathcal{B}(\mathbb{R}^{n(q')})$, $q' \in \mathcal{Q}$, and $(q,x) \in \mathcal{S}$. One can verify that the kernel T_s assigns to each $s = (q,x) \in \mathcal{S}$ a probability measure on the Borel space $(\mathcal{S}, \mathcal{B}(\mathcal{S}))$; moreover, under the conditions of Definition 1 the function $T_s(A|(q,\cdot))$ is Borel measurable for all $A \in \mathcal{B}(\mathcal{S})$ and all $q \in \mathcal{Q}$. The algorithm in Definition 2 now simplifies to:

set $k = 0$*;*
extract a value $s_k \in \mathcal{S}$ *for* $\mathbf{s}(k)$ *according to* $\mathrm{Init}(\cdot)$*;*
while $k < N$ *do*
    extract a value $s_{k+1} \in \mathcal{S}$ *for* $\mathbf{s}(k+1)$ *according to* $T_s(\cdot\,|s_k)$*;*
    $k \to k+1$*;*
end.

It follows that a DTSHS $\mathcal{H} = (\mathcal{Q}, n, \mathrm{Init}, T_x, T_q, R)$ defines a Markov process with state space $\mathcal{S} = \cup_{q\in\mathcal{Q}}\{q\} \times \mathbb{R}^{n(q)}$ and transition probability kernel $T_s : \mathcal{B}(\mathcal{S}) \times \mathcal{S} \to [0,1]$. The execution $\{\mathbf{s}(k),\, k \in [0,N]\}$ is a stochastic process defined on the canonical sample space $\Omega = \mathcal{S}^{N+1}$, endowed with the $\sigma-$algebra, $\mathcal{B}(\Omega)$, generated by the product topology, and with a probability measure P uniquely defined by the transition kernel T_s and the initial measure Init [9, Proposition 7.45]. In particular, we will use the notation P_{s_0} for the probability measure associated with the deterministic initial condition $s_0 \in \mathcal{S}$, i.e., $\mathrm{Init}(\cdot) = \delta_{s_0}(\cdot)$.

2.2 Formulation of the probabilistic invariance problem

We consider the following invariance problem: determine the probability that the state of a DTSHS $\mathcal{H}$ will remain within a certain "safe" set during the time horizon $[0,N]$, starting from an arbitrarily selected initial condition $s_0 \in \mathcal{S}$. The probabilistic invariance problem of interest can actually be referred to as a probabilistic safety problem. An invariance problem can be regarded as the dual of a reachability problem; this duality is formally discussed in [2].

Consider a compact Borel set $A \in \mathcal{B}(S)$, representing a safe set. Our goal is setting up an invariance computation procedure to determine the probability that the execution associated with the initial condition $s_0 \in \mathcal{S}$ will remain within A during the time horizon $[0,N]$:

$$\mathrm{p}_{s_0}(A) := P_{s_0}\{\mathbf{s}(k) \in A \text{ for all } k \in [0,N]\}. \tag{2}$$

If $\mathrm{p}_{s_0}(A) \geq \epsilon$, $\epsilon \in (0,1]$, we say that the system initialized at s_0 is safe with an ϵ probabilistic guarantee. Notice that the value of the probability $\mathrm{p}_{s_0}(A)$ depends on the initial condition. For a given $\epsilon \in (0,1]$, we can define as the *probabilistic safe set* with safety level ϵ the set

$$S(\epsilon) = \{s_0 \in \mathcal{S} : \mathrm{p}_{s_0}(A) \geq \epsilon\} \tag{3}$$

of those initial conditions s_0 that are safe with an ϵ probabilistic guarantee. We show that the problem of computing $\mathrm{p}_{s_0}(A)$ can be solved through a backward iterative procedure by representing $\mathrm{p}_{s_0}(A)$ as a multiplicative function.

Let $\mathbb{1}_C : \mathcal{S} \to \{0,1\}$ denote the indicator function of set $C \subseteq \mathcal{S}$: $\mathbb{1}_C(s) = 1$, if $s \in C$, and $\mathbb{1}_C(s) = 0$, if $s \notin C$. Observe that

$$\prod_{k=0}^{N} \mathbb{1}_A(s_k) = \begin{cases} 1, & \text{if } s_k \in A \text{ for all } k \in [0,N] \\ 0, & \text{otherwise,} \end{cases}$$

where $s_k \in \mathcal{S}$, $k \in [0, N]$. Then, the quantity $\mathrm{p}_{s_0}(A)$ in (2) can be expressed as the expectation with respect to the probability measure P_{s_0} of the Bernoulli random variable $\prod_{k=0}^{N} \mathbb{1}_A(\mathbf{s}(k))$:

$$\mathrm{p}_{s_0}(A) = E_{s_0}\left[\prod_{k=0}^{N} \mathbb{1}_A(\mathbf{s}(k))\right]. \tag{4}$$

Consider the sequence of functions $V_k : \mathcal{S} \to [0, 1]$, $k = 0, 1, \ldots, N$, defined by:

$$\begin{aligned} &V_k(s) = \mathbb{1}_A(s) \int_{\mathcal{S}^{N-k}} \prod_{l=k+1}^{N} \mathbb{1}_A(s_l) \prod_{l=k+1}^{N-1} T_s(ds_{l+1}|s_l) T_s(ds_{k+1}|s),\ s \in \mathcal{S},\ k = 0, 1, \ldots, N-1, \\ &V_N(s) = \mathbb{1}_A(s), s \in \mathcal{S}. \end{aligned} \tag{5}$$

It is easily seen that for any $k \in \{0, 1, \ldots, N\}$, $V_k(s)$ represents the probability that an execution of the DTSHS remains within the safe set A over the residual time horizon $[k, N]$, starting from s at time k. Following the dynamic programming terminology, we call $V_k(s)$ the value function at time k. In particular, $V_0(s) = E_s[\prod_{l=0}^{N} \mathbb{1}_A(\mathbf{s}(l))]$, $s \in \mathcal{S}$, evaluated at $s = s_0$ returns the quantity of interest $\mathrm{p}_{s_0}(A)$, and the probabilistic safe set with safety level ϵ defined in (3) can be expressed as $S(\epsilon) = \{s_0 \in \mathcal{S} : V_0(s_0) \geq \epsilon\}$.

The following result states that the value functions can be determined through a backward recursive procedure.

Proposition 1 ([2], Lemma 1) *The value functions $V_k : \mathcal{S} \to [0, 1]$, $k = 0, 1 \ldots, N-1$, defined in* (5) *can be computed through the following backward recursion:*

$$V_k(s) = \mathbb{1}_A(s) \int_{\mathcal{S}} V_{k+1}(s_{k+1}) T_s(ds_{k+1}|s),\ s \in \mathcal{S}, \tag{6}$$

initialized with $V_N(s) = \mathbb{1}_A(s), s \in \mathcal{S}$.

Since an explicit analytic solution to the recursion in equation (6) is generally impossible to find, the computational aspects to the problem are of key importance to its implementation. In Section 3 we design an approximation scheme for the numerical solution of the stochastic invariance problem. To this purpose, it is important to note that the value function $V_k : \mathcal{S} \to [0, 1]$ satisfies $V_k(s) = 0$, if $s \in \mathcal{S} \setminus A$, for any $k \in [0, N]$. As a consequence, the recursive equation (6) in Proposition 1 can be restricted to the compact set A:

$$\begin{aligned} &V_k(s) = \int_A V_{k+1}(s_{k+1}) T_s(ds_{k+1}|s),\ s \in A,\ k = 0, 1, \ldots, N-1, \\ &V_N(s) = 1, s \in A. \end{aligned} \tag{7}$$

The advantage of confining computations to the compact set A is that we can adopt a finite discretization of the continuous state component in the numerical scheme that approximates the quantity of interest. Moreover, under suitable regularity conditions on the transition kernels, the V_k functions can be shown to be Lipschitz continuous over A. This property (valid only within A, given the discontinuity when passing from a safe state within A to an unsafe state outside A) is used for determining bounds on the approximated numerical solution.

3 Estimation of invariance probability by finite state Markov chain approximation

We introduce a numerical scheme for estimating the invariance probability $\mathrm{p}_{s_0}(A)$ defined in (2). The scheme is based on a Discrete Time Markov Chain (DTMC) approximation of the original DTSHS. The proof that the estimated invariance probability converges to $\mathrm{p}_{s_0}(A)$ is inspired by [1, 8].

3.1 Approximating Markov chain

Let the safe set $A \in \mathcal{B}(\mathcal{S})$ be given by $A = \cup_{q\in\mathcal{Q}}\{q\} \times A_q$ with $A_q \in \mathcal{B}(\mathbb{R}^{n(q)})$. Recall that A is assumed to be compact. The size of the continuous state space within A is measured by $\lambda := \max_{q\in\mathcal{Q}} \mathcal{L}(A_q)$, where $\mathcal{L}(A_q) < \infty$ denotes the finite Lebesgue measure of the set $A_q \subset \mathbb{R}^{n(q)}$. Assume for simplicity that $A_q \neq \emptyset$ for all $q \in \mathcal{Q}$. Since A is compact, we can introduce a finite partition of each compact set $A_q \subset \mathbb{R}^{n(q)}$, $q \in \mathcal{Q}$, by taking $A_q = \cup_{i=1}^{m_q} A_{q,i}$, where $A_{q,i}$, $i = 1, \ldots, m_q$, are pairwise disjoint Borel sets $A_{q,i} \in \mathcal{B}(\mathbb{R}^{n(q)})$, with $A_{q,i} \cap A_{q,j} = \emptyset$, $\forall i \neq j$. Denote with $\delta_{q,i}$ the diameter of the set $A_{q,i}$, that is $\delta_{q,i} = \sup\{\|x - x'\| : x, x' \in A_{q,i}\}$, and define the *grid size parameter* by $\delta := \max_{i=1,\ldots,m_q, q\in\mathcal{Q}} \delta_{q,i}$.

The collection of sets $\mathcal{G} := \{G_{q,i} := \{q\} \times A_{q,i}, i = 1, \ldots, m_q, q \in \mathcal{Q}\}$ represents a partition of the safe set A. For each element $G_{q,i}$ of the partition, we select a representative point $(q, v_{q,i}) \in G_{q,i}$. The set $A_\delta := \{(q, v_{q,i}), i = 1, \ldots, m_q, q \in \mathcal{Q}\}$ is the discretized version of the safe set A. We denote with $\xi : A \to A_\delta$ the map that associates to $s \in G_{q,i} \subset A$ the corresponding discrete state $(q, v_{q,i}) \in A_\delta$, and with $\Xi : A_\delta \to \mathcal{G}$ the set-valued map that associates to $(q, v_{q,i}) \in A_\delta$ the set $G_{q,i}$ to which $(q, v_{q,i})$ belongs.

We next introduce the state space $\mathcal{Z}_\delta$ and the transition probability function $T_\delta : \mathcal{Z}_\delta \times \mathcal{Z}_\delta \to [0, 1]$ of a stochastic automaton that approximates the original DTSHS for the purpose of probabilistic invariance computation. The state space of the stochastic automaton is defined as $\mathcal{Z}_\delta := A_\delta \cup \{\phi\}$, where ϕ is a discrete state representing the set of all states in the hybrid state space $\mathcal{S}$ that are outside the safe set A. Notice that the compactness assumption on A ensures that the set $\mathcal{Z}_\delta$ is finite.

The transition probability function $T_\delta : \mathcal{Z}_\delta \times \mathcal{Z}_\delta \to [0, 1]$ is defined as follows:

$$T_\delta(z'|z) = \begin{cases} T_s(\Xi(z')|z), & \text{if } z' \in A_\delta \text{ and } z \in A_\delta \\ 1 - \sum_{\bar{z}\in A_\delta} T_s(\Xi(\bar{z})|z), & \text{if } z' = \phi \text{ and } z \in A_\delta \\ 1, & \text{if } z' = z = \phi \\ 0, & \text{if } z' \in A_\delta \text{ and } z = \phi, \end{cases} \tag{8}$$

and satisfies $\sum_{z'\in\mathcal{Z}_\delta} T_\delta(z'|z) = 1$, for all $z \in \mathcal{Z}_\delta$. Note that ϕ is an absorbing state and the probability that the stochastic automaton evolves from a safe state $z \in A_\delta$ to a safe state $z' \in A_\delta$ is defined as the probability that the original DTSHS will enter the safe set $\Xi(z') \subset A$ in one time step starting from z.

The execution during the time horizon $[0, N]$ of the stochastic finite automaton associated with the initial condition $z_0 \in \mathcal{Z}_\delta$ is a Markov chain $\{\mathbf{z}(k), k \in [0, N]\}$ defined on the probability space $(\mathcal{Z}_\delta^{N+1}, \sigma(\mathcal{Z}_\delta^{N+1}), P_{\delta,z_0})$, where $\sigma(\mathcal{Z}_\delta^{N+1})$ is the σ-algebra associated to $\mathcal{Z}_\delta^{N+1}$, and the probability measure P_{δ,z_0} is uniquely defined by the initial condition z_0 and the transition probability function T_δ.

3.2 Convergence result

Consider an initial condition $s_0 \in A$ for the DTSHS. Let $z_0 = \xi(s_0) \in A_\delta$ be the discrete state corresponding to s_0. We show that, under certain regularity conditions on the DTSHS, the probability

$$\mathrm{p}_{\delta,z_0}(A_\delta) := P_{\delta,z_0}\{\mathbf{z}(k) \in A_\delta \text{ for all } k \in [0, N]\} \tag{9}$$

computed on the approximating Markov chain initialized at $z_0 = \xi(s_0)$ converges to the invariance probability of interest $\mathrm{p}_{s_0}(A)$ of the DTSHS initialized at $s_0 \in A$, as the grid size parameter δ tends to zero. We also provide an expression for the rate of convergence.

Suppose that the stochastic kernels T_x and R on the continuous component of the hybrid state admit density t_x and r, and that t_x and r, as well as the stochastic kernel T_q, satisfy the following Lipschitz condition.

Assumption 1

1. $|T_q(\bar{q}|(q, x)) - T_q(\bar{q}|(q, x'))| \leq h_1\|x - x'\|$, *for all* $(q, x), (q, x') \in A$, *and* $\bar{q} \in \mathcal{Q}$,
2. $|t_x(\bar{x}|(q, x)) - t_x(\bar{x}|(q, x'))| \leq h_2\|x - x'\|$, *for all* $(q, x), (q, x'), (q, \bar{x}) \in A$,

3. $|r(\bar{x}|(q,x),\bar{q}) - r(\bar{x}|(q,x'),\bar{q})| \leq h_3\|x - x'\|$*, for all* $(q,x),(q,x'),(\bar{q},\bar{x}) \in A$*, and* $\bar{q} \neq q \in \mathcal{Q}$*,*

where h_1*,* h_2*, and* h_3 *are suitable finite Lipschitz constants.*

Based on Assumption 1, we can prove that the value functions V_k of the original probabilistic invariance problem for the DTSHS satisfy a Lipschitz condition over the set A. An analogous result for a DTSHS with inputs is stated without proof in [1, Theorem 2].

Theorem 1 *Under Assumption 1, the value functions* $V_k : \mathcal{S} \to [0,1]$ *of the probabilistic invariance problem for the DTSHS satisfy the following Lipschitz condition over A:*

$$|V_k((q,x)) - V_k((q,x'))| \leq \mathcal{K}\|x - x'\|,\ \forall\,(q,x),\ (q,x') \in A, \tag{10}$$

for any $k \in [0,N]$*. The constant* $\mathcal{K}$ *is given by* $\mathcal{K} = mh_1 + \lambda\big(h_2 + (m-1)h_3\big)$*, where* m *is the cardinality of* $\mathcal{Q}$ *and* λ *is the Lebesgue measure of the continuous state space within* A*.*

Proof: Since $V_N(s) = V_N(s') = 1$, for all $s, s' \in A$, then, the inequality in (10) is trivially satisfied for $k = N$. For any $k \in [0, N-1]$, (q,x), $(q,x') \in A$, from the backward recursion (7), the definition (1) of T_s, and the fact that $V_{k+1}(s) = 0$ for all $s \in \mathcal{S}\backslash A$, we have:

$$\begin{aligned}
|V_k((q,x)) - V_k((q,x'))| &= \Big|\int_A V_{k+1}(s)T_s(ds|(q,x)) - \int_A V_{k+1}(s)T_s(ds|(q,x'))\Big| \\
&= \Big|T_q(q|(q,x))\int_{A_q} V_{k+1}(q,\bar{x})T_x(d\bar{x}|(q,x)) + \sum_{\bar{q}\neq q} T_q(\bar{q}|(q,x))\int_{A_{\bar{q}}} V_{k+1}(\bar{q},\bar{x})R(d\bar{x}|(q,x),\bar{q}) \\
&\quad - T_q(q|(q,x'))\int_{A_q} V_{k+1}(q,\bar{x})T_x(d\bar{x}|(q,x')) - \sum_{\bar{q}\neq q} T_q(\bar{q}|(q,x'))\int_{A_{\bar{q}}} V_{k+1}(\bar{q},\bar{x})R(d\bar{x}|(q,x'),\bar{q})\Big| \\
&\leq \Big|T_q(q|(q,x))\int_{A_q} V_{k+1}(q,\bar{x})T_x(d\bar{x}|(q,x)) - T_q(q|(q,x'))\int_{A_q} V_{k+1}(q,\bar{x})T_x(d\bar{x}|(q,x'))\Big| \\
&\quad + \sum_{\bar{q}\neq q}\Big|T_q(\bar{q}|(x,q))\int_{A_{\bar{q}}} V_{k+1}(\bar{q},\bar{x})R(d\bar{x}|(q,x),\bar{q}) - T_q(\bar{q}|(q,x'))\int_{A_{\bar{q}}} V_{k+1}(\bar{q},\bar{x})R(d\bar{x}|(q,x'),\bar{q})\Big|.
\end{aligned} \tag{11}$$

We next show two intermediate results that will be useful for proving the Lipschitz property for V_k. The following chain of inequalities can be easily proven using the fact that $|V_{k+1}(\cdot)| \leq 1$ and Assumption 1:

$$\begin{aligned}
&\Big|\int_{A_q} V_{k+1}(q,\bar{x})T_x(d\bar{x}|(q,x)) - \int_{A_q} V_{k+1}(q,\bar{x})T_x(d\bar{x}|(q,x'))\Big| \leq \int_{A_q} |V_{k+1}(q,\bar{x})|\big|T_x(d\bar{x}|(q,x)) - T_x(d\bar{x}|(q,x'))\big| \\
&\quad \leq \int_{A_q} \big|T_x(d\bar{x}|(q,x)) - T_x(d\bar{x}|(q,x'))\big| \leq \lambda h_2\|x - x'\|.
\end{aligned}$$

Similarly, we have that

$$\Big|\int_{A_{\bar{q}}} V_{k+1}(\bar{q},\bar{x})R(d\bar{x}|(q,x),\bar{q}) - \int_{A_{\bar{q}}} V_{k+1}(\bar{q},\bar{x})R(d\bar{x}|(q,x'),\bar{q})\Big| \leq \lambda h_3\|x - x'\|.$$

Recall now that the product of two functions $\alpha, \beta : \mathcal{E} \to \mathbb{R}$ that are Lipschitz continuous over a compact set $\mathcal{E}$ of an Euclidean space, with Lipschitz constants respectively h_α and h_β, satisfies:

$$|\alpha(w_1)\beta(w_1) - \alpha(w_2)\beta(w_2)| \leq \left\{h_\alpha \sup_{w\in\mathcal{E}}|\beta(w)| + h_\beta \sup_{w\in\mathcal{E}}|\alpha(w)|\right\}\|w_1 - w_2\|.$$

Applying this inequality to the two terms in the right-hand side of equation (11) with $\alpha(w) = T_q(q|(q,w))$, and with either $\beta(w) = \int_{A_q} V_{k+1}(q,\bar{x})T_x(d\bar{x}|(q,w))$ or $\beta(w) = \int_{A_{\bar{q}}} V_{k+1}(q,\bar{x})R(d\bar{x}|(\bar{q},w),\bar{q})$, it follows that

$$|V_k((q,x)) - V_k((q,x'))| \leq \big[mh_1 + \lambda\big(h_2 + (m-1)h_3\big)\big]\|x - x'\|,$$

which concludes the proof. ■

Based on Theorem 1, we can finally prove the main convergence result.

Theorem 2 *Under Assumption 1, the invariance probability* $\mathrm{p}_{s_0}(A)$ *for the DTSHS initialized at* $s_0 \in A$ *satisfies*

$$\left|\mathrm{p}_{s_0}(A) - \mathrm{p}_{\delta,z_0}(A_\delta)\right| \leq \gamma\delta,$$

where $\mathrm{p}_{\delta,z_0}(A_\delta)$ *is the invariance probability for the approximating Markov chain with grid size* δ *initialized at the discrete state* $z_0 = \xi(s_0) \in A_\delta$*, and* $\gamma = N\mathcal{K}$*.*

Proof: Fix $\delta > 0$ and consider the MC on $\mathcal{Z}_\delta = A_\delta \cup \{\phi\}$ with transition probability $T_\delta : \mathcal{Z}_\delta \times \mathcal{Z}_\delta \to [0,1]$ defined in (8). Given that ϕ is an absorbing state, the invariance probability $\mathrm{p}_{\delta,z_0}(A_\delta)$ in (9) of the approximating Markov chain can be computed as

$$\mathrm{p}_{\delta,z_0}(A_\delta) = P_{\delta,z_0}\{\mathbf{z}(N) \in A_\delta\}.$$

Let $V_{\delta,k} : \mathcal{Z}_\delta \to [0,1]$, for all $k \in [0,N]$, represent the conditional probability that a Markov chain execution of the automaton that takes the value z at time k will be within the safe set A_δ at time N. Clearly, the invariance probability of interest can be computed as $\mathrm{p}_{\delta,z_0}(A_\delta) = V_{\delta,0}(z_0)$. Moreover, $V_{\delta,N}(z) = \mathbf{1}_{A_\delta}(z)$, $z \in \mathcal{Z}_\delta$, and, for $k \in [0,N-1]$, $V_{\delta,k} : \mathcal{Z}_\delta \to [0,1]$ satisfies the following recursive equation

$$V_{\delta,k}(z) = \sum_{z' \in \mathcal{Z}_\delta} T_\delta(z'|z) V_{\delta,k+1}(z').$$

Given that $V_{\delta,k}(\phi) = 0$, $k \in [0,N]$, we have that

$$\begin{aligned} V_{\delta,k}(z) &= \sum_{z' \in A_\delta} T_\delta(z'|z) V_{\delta,k+1}(z'), \;\; z \in A_\delta,\, k = 0,1,\ldots,N-1, \\ V_{\delta,N}(z) &= 1, \;\; z \in A_\delta, \end{aligned} \tag{12}$$

which is the discretized version of the backward iteration (7).

Let us introduce the piecewise constant function $\hat{V}_k(s) = V_{\delta,k}(\xi(s))$, $s \in A$. We next prove by induction on k that

$$|V_k(s) - \hat{V}_k(s)| \leq (N-k)\mathcal{K}\delta, \tag{13}$$

holds for any $k = 0,1,\ldots,N$. The claim then follows by setting $k = 0$ in equation (13) and recalling that $\mathrm{p}_{s_0}(A) = V_0(s_0)$ and $\mathrm{p}_{\delta,z_0}(A_\delta) = V_{\delta,0}(\xi(s_0))$.

Since $V_N(s) = \hat{V}_N(s) = 1$, $s \in A$, then, equation (13) trivially holds for $k = N$. Let us suppose by induction hypothesis that $\left|V_{k+1}(s) - \hat{V}_{k+1}(s)\right| \leq (N-k-1)\mathcal{K}\delta$, $s \in A$, for $k+1 < N$. Observe that

$$|V_k(s) - \hat{V}_k(s)| = |V_k(s) - \hat{V}_k(\xi(s))| \leq |V_k(s) - V_k(\xi(s))| + |V_k(\xi(s)) - \hat{V}_k(\xi(s))|, \;\; s \in A. \tag{14}$$

By Theorem 1, it is easily seen that the first term in the right hand-side of this equation is bounded by

$$|V_k(s) - V_k(\xi(s))| \leq \mathcal{K}\delta,\; s \in A.$$

For the second term, by the backward recursions (7) and (12), and the definition of the approximating Markov chain transition probability function (8), we get

$$\begin{aligned} |V_k(\xi(s)) - \hat{V}_k(\xi(s))| &= \Big| \int_A V_{k+1}(w) T_s(dw|\xi(s)) - \sum_{z \in A_\delta} T_\delta(z|\xi(s)) \hat{V}_{k+1}(z) \Big| \\ &= \Big| \int_A V_{k+1}(w) T_s(dw|\xi(s)) - \int_A \hat{V}_{k+1}(w) T_s(dw|\xi(s)) \Big| \\ &\leq \int_A \left|V_{k+1}(w) - \hat{V}_{k+1}(w)\right| T_s(dw|\xi(s)) \\ &\leq (N-k-1)\mathcal{K}\delta, s \in A, \end{aligned}$$

where the last inequality follows from the induction hypothesis. By plugging these two bounds into equation (14), the proof of (13) is completed. ■

Notice that the quality of the approximation by the numerical procedure improves as the grid size parameter δ decreases. The rate of convergence is linear in δ and depends on the Lipschitz constants h_1, h_2, and h_3 in Assumption 1 through the $\mathcal{K}$ constant defined in Theorem 1. This is not surprising because the value function V_0 over the set A is approximated by a piecewise constant function through the discretization process, and we expect such a piecewise constant approximation to be more accurate for a smoother V_0 function. As the time horizon grows, the approximation error propagates. This is taken into account by the constant γ in Theorem 2, which grows linearly with the time-horizon length N.

Though the bound in Theorem 2 is conservative, it holds uniformly over A, which allows one to approximate the probabilistic safe set $S(\epsilon)$, $\epsilon \in (0,1]$, defined in (3) by model checking the invariance property of the approximating finite state Markov chain. This is detailed in Section 4.3.

4 Finite state Markov chain model checking

4.1 Logical specification of probabilistic invariance

A powerful and efficient analysis technique for assessing a large variety of properties, including probabilistic invariance, on discrete time and finite state stochastic automata generating Markov chains is *model checking* [6, 14]. Let us consider a stochastic automaton with state space $\mathcal{Z}$ and transition probability function $T : \mathcal{Z} \times \mathcal{Z} \to [0,1]$. The execution of the automaton associated with some initial condition $\bar{z} \in \mathcal{Z}$ is a DTMC whose sample paths $z_0, z_1, z_2, \ldots$, satisfy $z_0 = \bar{z}$ and $T(z_{i+1}|z_i) > 0$, $i = 0, 1, \ldots$. The model checking approach to probabilistic verification consists in specifying the property of interest in terms of a probabilistic temporal logic formula, and in computing the set of initial conditions of the stochastic automaton such that the corresponding DTMC executions satisfy that formula.

According to the Probabilistic Computation Tree Logic (PCTL) [20], the probabilistic invariance property for a DTMC with safe set $D \subset \mathcal{Z}$ can be expressed by the formula

$$\mathbb{P}_{\geq \epsilon} \left(\Box^{\leq N} \Phi_D \right), \tag{15}$$

which holds in the state $\bar{z}$ whenever the DTMC execution associated with the initial condition $\bar{z}$ satisfies the sub-formula $\Box^{\leq N}\ \Phi_D$ with probability at least equal to $\epsilon \in [0,1]$. The *state formula* Φ_D characterizes the safe set D, i.e., Φ_D holds in state z if and only if $z \in D$. A sample path of the DTMC satisfies the *path formula* $\Box^{\leq N} \Phi_D$ if its first N states all belong to D, i.e. they are all safe. The symbol $\Box$ should be read as "always".

The validity of a formula in a state is formally defined by means of a satisfaction relation, denoted by $\models$. For instance,

$$\bar{z} \models \mathbb{P}_{\geq \epsilon} \left(\Box^{\leq N} \Phi_D \right)$$

denotes that state $\bar{z}$ satisfies formula (15).

Similarly, the path formula $\Diamond \Phi$ asserts that at some point a state satisfying formula Φ is reached. Let $\bar{D}$ be the complement of D in $\mathcal{Z}$, i.e., the set of unsafe states. Then, $\Phi_{\bar{D}} = \neg \Phi_D$ with $\neg$ denoting logical negation. A path $z_0, z_1, \ldots$ satisfies $\Diamond \Phi_{\bar{D}}$ if some of its states are unsafe. $\Diamond \Phi_{\bar{D}}$ thus expresses a reachability specification over the unsafe set $\bar{D}$. The duality between probabilistic invariance and probabilistic reachability [2] can be expressed in PCTL as:

$$\mathbb{P}_{\geq \epsilon} \left(\Box^{\leq N} \Phi_D \right) \ \equiv \ \mathbb{P}_{\leq 1-\epsilon} \left(\Diamond^{\leq N} \Phi_{\bar{D}} \right). \tag{16}$$

More complex properties can be stated in a similar manner via logical specifications. For instance, assume that we are interested in the reachability of a desired set C via some set B of admissible states within the bounded time horizon $[0, N]$, with probability at least ϵ. This is expressed by the formula:

$$\mathbb{P}_{\geq \epsilon} \left(\Phi_B \, \mathcal{U}^{\leq N} \Phi_C \right),$$

involving the so-called bounded-until operator ($\mathcal{U}^{\leq \mathcal{N}}$). Intuitively, a path satisfies $\Phi_B \mathcal{U}^{\leq N} \Phi_C$ if it reaches a desired state (in C) within $[0, N]$ while all states prior to this state are admissible (in B). If $N = \infty$, then the unbounded-until operator ($\mathcal{U}$) poses no finite time restriction on reaching set C, as long as it is reached via an admissible path.

In general, the syntax of PCTL is as follows. Let N be a natural number or ∞, $\sim$ a binary comparison operator such as $<$, $\leq$, or, dually, $>$ $\geq$, and $\epsilon \in [0, 1]$. A formula Φ in PCTL is built up from the basic state formula `true` and from the generic atomic proposition a, and can be obtained by combining two PCTL-formulae by conjunction ($\wedge$), by prefixing a PCTL-formula with negation ($\neg$), or by considering a path formula characterized by the operator $\mathcal{U}^{\leq N}$ and contained in a $\mathbb{P}$-context parameterized by a probability (ϵ) and a binary comparison operator ($\sim$):

$$\Phi ::= \texttt{true} \;\Big|\; a \;\Big|\; \Phi \wedge \Phi \;\Big|\; \neg \Phi \;\Big|\; \mathbb{P}_{\sim \epsilon} \left(\Phi \, \mathcal{U}^{\leq N} \, \Phi \right).$$

Note that $\Box^{\leq N}$ is the dual operator to $\Diamond^{\leq N}$, and that $\Diamond^{\leq N} \Phi$ can be defined as $\texttt{true} \mathcal{U}^{\leq N} \Phi$. Based on (16), the probabilistic invariance property can then be expressed in terms of the bounded-until operator as

$$\mathbb{P}_{\geq \epsilon} \left(\Box^{\leq N} \Phi_D \right) \;\equiv\; \mathbb{P}_{\leq 1-\epsilon} \left(\texttt{true} \mathcal{U}^{\leq N} \neg \Phi_D \right). \tag{17}$$

Given that the other usual boolean connectives such as disjunction, implication, and equivalence can be derived in the usual way, e.g., $\Phi \vee \Psi = \neg(\neg \Phi \wedge \neg \Psi)$, this completes the logic. The formal semantics of the logic falls outside the scope of this paper and can be found in [6, 20].

4.2 Model-checking algorithm

In this section, we summarize the model-checking algorithm for PCTL over DTMCs, which is based on [15, 20]. The inputs to the algorithm are a MC with finite state space $\mathcal{Z}$ and transition probability function $T : \mathcal{Z} \times \mathcal{Z} \to [0, 1]$, and a PCTL formula Φ. The states of the DTMC are assumed to be labeled with sets of atomic propositions. The output is the set of states satisfying formula Φ: $Sat(\Phi) = \{z \in \mathcal{Z} \mid z \models \Phi\}$. PCTL model checking is carried out by recursively computing the set $Sat(\Phi)$, in the same way as verifying the non-probabilistic temporal logic CTL [6, 14] from which PCTL originates. This is done by means of a bottom-up recursive algorithm over the parse tree of Φ. Each node of this tree is labeled with a sub-formula of Φ, the root node is labeled with Φ, and the leaves are either labeled with `true` or some atomic proposition a. Starting from the leaves of the tree, the set of states satisfying each sub-formula is computed recursively moving upwards towards the root. For most of the operators in the logic, such as negation and conjunction, this step is straightforward. The main difficulty is represented by the sub-formulas involving the $\mathbb{P}_{\sim \epsilon}(\cdot)$ operator. For the sake of this paper, we concentrate on bounded-until formulas since we are interested in model-checking the probabilistic invariance property (17).

Consider the problem of checking the formula $\mathbb{P}_{\sim \epsilon} \left(\Phi \, \mathcal{U}^{\leq N} \, \Psi \right)$ with $N < \infty$. Let $\pi_k(z)$ denote the probability that the DTMC execution of the stochastic automaton starting from z at time k reaches a Ψ-state within the residual time horizon $[k, N]$ via paths of all Φ-states. The set of states $Sat\big(\mathbb{P}_{\sim \epsilon} \left(\Phi \, \mathcal{U}^{\leq N} \, \Psi \right) \big)$ can then be expressed in terms of $\pi_0(\cdot)$ as: $Sat\big(\mathbb{P}_{\sim \epsilon} \left(\Phi \, \mathcal{U}^{\leq N} \, \Psi \right) \big) = \{z \in \mathcal{Z} : \pi_0(z) \sim \epsilon\}$. In turn, it is easily seen that

$$\pi_k(z) \;=\; \begin{cases} 1, & \text{if } z \models \Psi \\ 0, & \text{if } (k{=}N \text{ and } z \models \neg \Psi) \text{ or } z \models \neg \Phi \wedge \neg \Psi \\ \sum_{z' \in \mathcal{Z}} T(z'|z) \pi_{k+1}(z'), & \text{otherwise,} \end{cases}$$

so that $\pi_0(\cdot)$ can be recursively computed backwards, starting from $\pi_N(\cdot) \;=\; \mathbb{1}_{Sat(\Psi)}(\cdot)$.

The following alternative approach shows how the probability $\pi_0(z)$, $z \in \mathcal{Z}$, can be expressed and computed in terms of the transient probabilities of a suitably defined DTMC. Given a PCTL formula Υ, consider the

transition probability function $T[\Upsilon] : \mathcal{Z} \times \mathcal{Z} \to [0,1]$ defined as:

$$T[\Upsilon](z'|z) \;=\; \begin{cases} T(z'|z), & \text{if } z \models \neg\Upsilon \\ 1, & \text{if } z \models \Upsilon \text{ and } z' = z \\ 0, & \text{otherwise.} \end{cases}$$

Clearly, this modified transition probability function makes all the states satisfying Υ absorbing. For the purpose of model-checking the formula $\mathbb{P}_{\sim\epsilon}\left(\Phi\,\mathcal{U}^{\leq N}\,\Psi\right)$, one can then make all $\neg(\Phi\vee\Psi)$-states and all Ψ-states absorbing by considering $T[\Upsilon](\cdot|\cdot)$ with $\Upsilon = \neg\Phi\vee\Psi$, since $\neg\Phi\vee\Psi = \neg(\Phi\vee\Psi)\vee\Psi$. The $\neg(\Phi\vee\Psi)$-states are defined as absorbing since $\Phi\,\mathcal{U}^{\leq N}\,\Psi$ is violated as soon as some state is visited that neither satisfies Φ nor Ψ; whereas the Ψ-states are defined as absorbing since, once a Ψ-state is reached (along a Φ-path) in at most N steps, then $\Phi\,\mathcal{U}^{\leq N}\,\Psi$ holds, regardless of which states will be visited later on.

As a result of this construction, for any $z \in \mathcal{Z}$, the probability $\pi_0(z)$ can be computed as the probability that the DTMC with transition probability function $T[\neg\Phi\vee\Psi](\cdot|\cdot)$ starting from z at time 0 will be within $Sat(\Psi)$ at time N. The probability distribution at time k of this DTMC can be expressed as

$$\Pi_z^k := e_z \cdot \mathrm{P}[\neg\Phi\vee\Psi]^k,$$

where e_z is a row probability vector whose elements are all equal to 0 except for a single one corresponding to state z, and $\mathrm{P}[\neg\Phi\vee\Psi]$ is the one-step transition probability matrix obtained by appropriately arranging in different columns the sequences $\{T[\neg\Phi\vee\Psi](z'|z),\, z\in\mathcal{Z}\}$ corresponding to the different $z'\in\mathcal{Z}$. Finally, the probability of interest $\pi_0(z)$ can be computed as

$$\pi_0(z) = \Pi_z^N \cdot e_\Psi = e_z \cdot \mathrm{P}[\neg\Phi\vee\Psi]^N \cdot e_\Psi,$$

where e_Ψ is a column vector that characterizes $Sat(\Psi)$, i.e., each element of e_Ψ takes values in $\{0,1\}$ and is equal to 1 if it corresponds to $z \models \Psi$, and 0 otherwise.

The complexity of model checking the PCTL formula $\mathbb{P}_{\sim\epsilon}\left(\Phi\,\mathcal{U}^{\leq N}\,\Psi\right)$ then mainly depends on the size of the one-step transition probability matrix $\mathrm{P}[\neg\Phi\vee\Psi]$. Determining the set of states that satisfy $\mathbb{P}_{\sim\epsilon}\left(\Phi\,\mathcal{U}^{\leq N}\,\Psi\right)$ in fact amounts to computing $\mathrm{P}[\neg\Phi\vee\Psi]^N\cdot e_\Psi$. In order to exploit the possible sparsity of $\mathrm{P}[\neg\Phi\vee\Psi]$, i.e. the presence of many zero elements in such a matrix, the product $\mathrm{P}[\neg\Phi\vee\Psi]^N\cdot e_\Psi$ is typically computed in an iterative fashion: $\mathrm{P}[\neg\Phi\vee\Psi]\cdot(\ldots(\mathrm{P}[\neg\Phi\vee\Psi]\cdot e_\Psi))$.

4.3 Approximation of the probabilistic safe set

We next describe a computational procedure to determine a conservative approximation of the safe set $S(\epsilon) = \{s_0 \in \mathcal{S} : \mathrm{p}_{s_0}(A) \geq \epsilon\}$, $\epsilon \in (0,1)$, of a DTSHS by model checking the invariance property of the approximating finite state Markov chain.

Algorithm 1 (probabilistic safe set approximation)

select $\eta > 0$ such that $\frac{\eta}{2} \in (0, 1-\epsilon)$;
select $\delta > 0$ such that $\gamma\delta \leq \frac{\eta}{2}$ (cf. Theorem 2 for the definition of γ);
construct the approximating Markov chain with grid size δ according to the procedure in Section 3.1;
use the model checker to compute $Z_\delta(\epsilon+\frac{\eta}{2}) = \left\{z_0 \in \mathcal{Z}_\delta :\; z_0 \models \mathbb{P}_{\leq 1-(\epsilon+\frac{\eta}{2})}\left(\mathtt{true}\,\mathcal{U}^{\leq N}\,\neg\Phi_{A_\delta}\right)\right\}$;
define the approximating safe set as $\hat{S}_\eta(\epsilon) = \left\{s_0 \in \mathcal{S} :\; \xi(s_0) \in Z_\delta(\epsilon+\frac{\eta}{2})\right\}$.

Theorem 3 *Under Assumption 1, for any $\epsilon \in (0,1)$, the safe set approximation $\hat{S}_\eta(\epsilon)$ obtained through Algorithm 1 satisfies $S(\epsilon+\eta) \subseteq \hat{S}_\eta(\epsilon) \subseteq S(\epsilon)$.*

Proof: By Theorem 2 and the condition $\gamma\delta \leq \frac{\eta}{2}$ in the second step of Algorithm 1, we have that

$$|\mathrm{p}_{s_0}(A) - \mathrm{p}_{\delta,z_0}(A_\delta)| \leq \gamma\delta \leq \frac{\eta}{2}, \quad z_0 = \xi(s_0), s_0 \in A. \tag{18}$$

Let $s_0 \in \hat{S}_\eta(\epsilon)$. Then, by construction, $z_0 = \xi(s_0) \in Z_\delta(\epsilon + \frac{\eta}{2})$ and, hence, $z_0 \models \mathbb{P}_{\leq 1-(\epsilon+\frac{\eta}{2})}\left(\texttt{true}\ \mathcal{U}^{\leq N}\ \neg\Phi_{A_\delta}\right)$. Since, according to the PCTL syntax, $\mathbb{P}_{\leq 1-(\epsilon+\frac{\eta}{2})}\left(\texttt{true}\ \mathcal{U}^{\leq N}\ \neg\Phi_{A_\delta}\right)$ is equivalent to the probabilistic invariance formula $\mathbb{P}_{\geq \epsilon+\frac{\eta}{2}}\left(\Box^{\leq N}\Phi_{A_\delta}\right)$, this implies that

$$\mathrm{p}_{\delta,z_0}(A_\delta) \geq \epsilon + \frac{\eta}{2}. \tag{19}$$

Bound (19) combined with (18) leads to $\mathrm{p}_{s_0}(A) \geq \epsilon$; hence, $\hat{S}_\eta(\epsilon) \subseteq S(\epsilon)$.

Suppose now that $s_0 \in S(\epsilon + \eta)$. Then, $\mathrm{p}_{s_0}(A) \geq \epsilon + \eta$ and, by (18), $\mathrm{p}_{\delta,z_0}(A_\delta) \geq \epsilon + \frac{\eta}{2}$ with $z_0 = \xi(s_0)$. This in turn implies that $z_0 = \xi(s_0) \in Z_\delta(\epsilon + \frac{\eta}{2})$, and, by the last step in Algorithm 1, that $s_0 \in \hat{S}_\eta(\epsilon)$. Hence, $S(\epsilon + \eta) \subseteq \hat{S}_\eta(\epsilon)$. ■

Theorem 3 is easy to interpret based on the approximation result in Theorem 2. It simply states that, in order to guarantee a certain safety level $\epsilon \in (0, 1)$ for the original DTSHS, we have to require a higher safety level $\epsilon + \frac{\eta}{2}$ for the approximating Markov chain so as to compensate for the approximation error $\frac{\eta}{2}$ introduced by the gridding procedure.

Note that η can be made arbitrarily small at the cost of decreasing the grid size parameter δ. However, the gap between the two sets $S(\epsilon+\eta)$ and $S(\epsilon)$ (measured e.g. by $\max_{q\in\mathcal{Q}} \mathcal{L}(\Delta X_q)$ with $\Delta X_q = \{x \in A_q : (q, x) \in S(\epsilon) \setminus S(\epsilon + \eta)\}$) may still be arbitrarily large if $\mathrm{p}_{s_0}(A)$ defining $S(\epsilon)$ happens to be flat around those values of s_0 mapping into ϵ.

Remark 1 *In the case when $\epsilon = 1$, one can obtain an over-approximation $\hat{S}_\eta(\epsilon)$ of $S(\epsilon)$ satisfying $S(\epsilon) \subseteq \hat{S}_\eta(\epsilon) \subseteq S(\epsilon - \gamma\delta)$ by choosing $\delta > 0$ and following the last three steps of Algorithm 1 with $\eta = 0$.*

5 Case study: The multi-room heating problem

We present the results of a computational study for a multi-room heating benchmark introduced in [18], based on a model proposed by [28]. The objective is to assess the performance of approximate model checking for the verification of probabilistic invariance of DTSHSs.

5.1 Description of the multi-room heating system

We study a DTSHS model for the temperature evolution in a building with h rooms. Each room is equipped with a heater and each heater switches between the `ON` and `OFF` conditions depending on the temperature in the room. The state of the system is hybrid, with the discrete state component representing the status of the h heaters and the continuous state component representing the average temperatures in the h rooms. The discrete state space is given by $\mathcal{Q} = \{\texttt{ON}, \texttt{OFF}\}^h$. The continuous state space is $\mathbb{R}^h$, irrespectively of the discrete state value (that is, $n(q) = h$ for all $q \in \mathcal{Q}$). Thus, the hybrid state space is $\mathcal{S} = \mathcal{Q} \times \mathbb{R}^h$.

We suppose that the average temperature of each room, say room i, evolves according to the following stochastic difference equation during the finite time horizon $[0, N]$:

$$\mathbf{x}_i(k+1) = \mathbf{x}_i(k) + b_i(x_a - \mathbf{x}_i(k)) + \sum_{i\neq j} a_{ij}(\mathbf{x}_j(k) - \mathbf{x}_i(k)) + c_i \mathbb{1}_{Q_i}(\mathbf{q}(k)) + \mathbf{w}_i(k), \tag{20}$$

where x_a represents the ambient temperature (assumed to be constant and equal for the whole building) and $\mathbb{1}_{Q_i}(\cdot)$ is the indicator function of set $Q_i = \{(q_1, \ldots, q_h) \in \mathcal{Q} : q_i = \texttt{ON}\}$. Equation (20) is the discrete time version of the stochastic differential equation in [28] with sample time interval Δt. The quantities b_i, a_{ij}, and c_i are non-negative constants representing the average heat transfer rate from room i to the ambient (b_i) and to room $j \neq i$ (a_{ij}), and the heat rate supplied to room i by the heater in room i (c_i), all rescaled by Δt and normalized with respect to the thermal capacity of room i. The values taken by the a_{ij} constants reflect the

room layout, for instance $a_{ij} = 0$ if rooms i and j are not adjacent. The disturbance $\{\mathbf{w}_i(k),\, k = 0, \ldots, N\}$ affecting the temperature evolution in room i is assumed to be a sequence of independent identically distributed Gaussian random variables with zero mean and variance ν^2 proportional to Δt. Furthermore, with no loss of generality we suppose that the disturbances $\mathbf{w}_i$ and $\mathbf{w}_j$ affecting the temperature of different rooms ($i \neq j$) are independent.

It is worth noticing that while a state-dependent disturbance could be easily modeled through the DTSHS in Definition 1, modeling a disturbance with time-varying characteristics would require to extend the DTSHS model and consider time-varying transition kernels. Though conceptually simple, this extension is not included in this paper to keep the notations simple while explaining the approximate model checking approach.

The continuous transition kernel T_x describing the evolution of the continuous state $\mathbf{x} = (\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_h)$ can be easily derived from (20). To this purpose, let $\mathcal{N}(\cdot; \mu, V)$ denote the Gaussian measure over $(\mathbb{R}^h, \mathcal{B}(\mathbb{R}^h))$, with mean $\mu \in \mathbb{R}^h$ and covariance matrix $V \in \mathbb{R}^{h \times h}$. Then, $T_x : \mathcal{B}(\mathbb{R}^h) \times \mathcal{S} \to [0, 1]$ can be expressed as

$$T_x(\cdot\,|(q, x)) = \mathcal{N}(\cdot; x + \Sigma x + \Gamma(q), \nu^2 I), \tag{21}$$

where $\Sigma \in \mathbb{R}^{h \times h}$, $\Gamma(q) \in \mathbb{R}^h$, and $I \in \mathbb{R}^{h \times h}$ is the identity matrix. For $i = 1, \ldots, h$, the element in row i and column j of matrix Σ is given by $[\Sigma]_{ij} = a_{ij}$, if $j \neq i$, and $[\Sigma]_{ij} = -(b_i + \sum_{k \neq i, k \in \mathcal{Q}} a_{ik})$, if $j = i$. For $i = 1, \ldots, h$, the i^{th} element of vector $\Gamma(q)$, $q = (q_1, q_2, \ldots, q_h) \in \mathcal{Q}$, is given by $[\Gamma(q)]_i = b_i x_a + c_i$, if $q_i = \texttt{ON}$, and $[\Gamma(q)]_i = b_i x_a$, if $q_i = \texttt{OFF}$. The reset kernel is set to coincide with the transition kernel in the current mode, irrespectively of the status to which the heaters possibly switch: $R(\cdot\,|(q, x), q') = T_x(\cdot\,|(q, x))$, for any $q, q' \in \mathcal{Q}$, and any $x \in \mathbb{R}^h$.

As for the discrete state evolution, we suppose that each heater switches status based on the temperature of the room where it is located, and independently of the other heaters. This is modeled taking the discrete transition kernel $T_q : \mathcal{Q} \times \mathcal{S} \to [0, 1]$ as the product of h conditional stochastic kernels $T_{q,i} : \{\texttt{ON}, \texttt{OFF}\} \times (\{\texttt{ON}, \texttt{OFF}\} \times \mathbb{R}) \to [0, 1]$ governing the switching of each single heater. More precisely, we set

$$T_q(q'|(q, x)) = \prod_{i=1}^{h} T_{q,i}(q_i'|(q_i, x_i)), \tag{22}$$

$q = (q_1, q_2, \ldots, q_h)$, $q' = (q_1', q_2', \ldots, q_h') \in \mathcal{Q}$, $x = (x_1, x_2, \ldots, x_h) \in \mathbb{R}^h$, where

$$T_{q,i}(q_i'|(q_i, x_i)) = \begin{cases} \sigma_i(x_i), & q_i' = \texttt{OFF}, \\ 1 - \sigma_i(x_i), & q_i' = \texttt{ON} \end{cases} \tag{23}$$

with $\sigma_i : \mathbb{R} \to [0, 1]$ a sigmoidal function given by

$$\sigma_i(y) = \frac{y^{d_i}}{\alpha_i^{d_i} + y^{d_i}},\ y \in \mathbb{R}. \tag{24}$$

Function $\sigma_i(y)$, $y \in \mathbb{R}$, is parameterized by a "threshold" parameter α_i and a "steepness" parameter $d_i > 0$. α_i is the value of y at which the probability of the heater changing status becomes equal to 0.5, whereas d_i is related to the slope of the sigmoidal function at $y = \alpha_i$ (which amounts to $d_i/(4\alpha_i)$).
Three possible sets of values for α_i and d_i are reported in Table 1. We shall refer to the three possible values for the steepness parameter d_i respectively as *flat*, *gradual*, and *steep*, in increasing order, and, similarly, to those for the threshold α_i respectively as *low*, *medium*, and *high*, again in increasing order. The values for the threshold α_i are determined as a convex combination of the temperatures x_i^l and x_i^u, $x_i^l < x_i^u$, defining the desired temperature range $[x_i^l, x_i^u]$ in room i. In the examples to follow, temperature is measured in degrees Celsius and $\Delta t = 0.25$ minutes.

Figure 1 represents a sample path of the Markov process generated by the DTSHS model of a 2-room heating system, when $\mathbf{q}(0)$ and $\mathbf{x}(0)$ are extracted uniformly from $\mathcal{Q}$ and $[x_1^l, x_1^u] \times [x_2^l, x_2^u] = [17, 22] \times [16, 23] \subseteq \mathbb{R}^2$, respectively. The sample paths over the time horizon $[0, N]$ of length $N = 100$ of both the discrete and the continuous components of the hybrid execution are plotted: the blue/darker color is associated with room 1,

d_i	1 (flat)	10 (gradual)	100 (steep)
α_i	$\frac{3}{4}x_i^l + \frac{1}{4}x_i^u$ (low)	$\frac{1}{2}(x_i^l + x_i^u)$ (medium)	$\frac{1}{4}x_i^l + \frac{3}{4}x_i^u$ (high)

Table 1: Possible choices for the steepness and threshold parameters d_i and α_i.

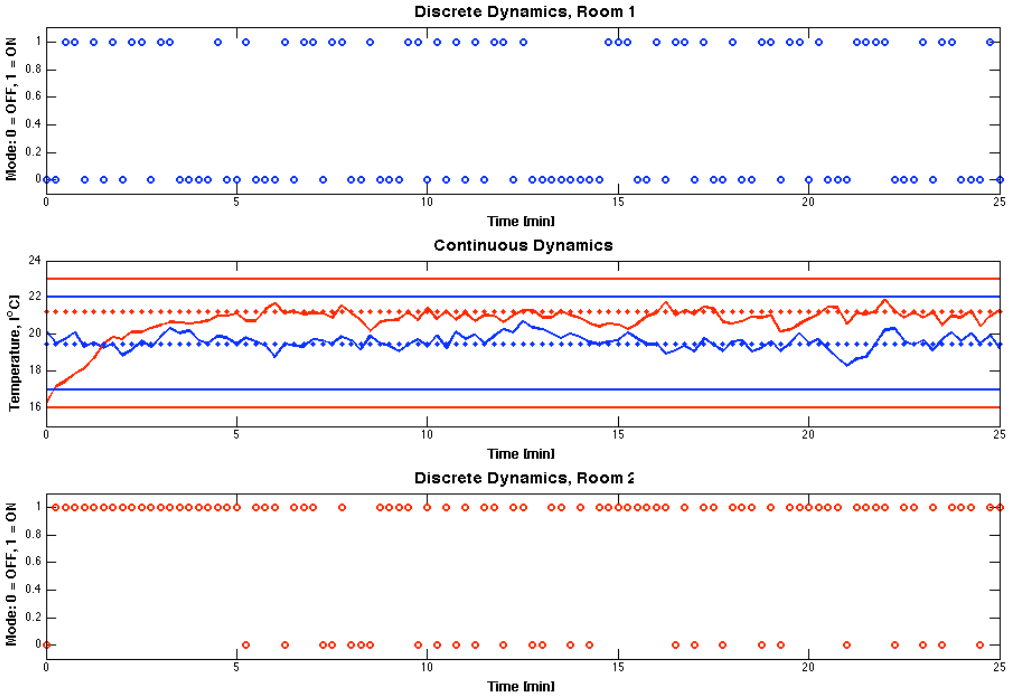


Figure 1: A sample path of the DTSHS modeling the 2-room heating system during the time horizon $[0, 100]$ min. The blue/darker color is associated with the dynamics of room 1, whereas the red/lighter color with those of room 2. The evolution of both the discrete and the continuous components is plotted.

whereas the red/lighter color with room 2. The adopted functions $\sigma_1(\cdot)$ and $\sigma_2(\cdot)$ are reported in Figure 2, and correspond to steep d_1 and medium α_1 parameters for the first room ($d_1 = 100, \alpha_1 = 19.5$), and to steep d_2 and high α_2 parameters for the second room ($d_2 = 100, \alpha_2 = 21.25$).
The parameters involved in the definition (21) of the continuous transition kernel T_x (and, hence, the reset kernel R) are set equal to $a_{12} = a_{21} = 0.0625$, $b_1 = 0.0375$, $b_2 = 0.025$, $c_1 = 0.65, c_2 = 0.6$, and $x_a = 6$, and the noise standard deviation is set equal to $\nu = 0.25$.

5.2 Model checking probabilistic invariance

In this section, we analyze the probabilistic invariance property of the h-room heating system by applying model checking to an approximation of the DTSHS in the form of a Markov chain.

We consider a safe set $A \subset \mathcal{S}$ as $A = \mathcal{Q} \times [x_1^l, x_1^u] \times \cdots \times [x_h^l, x_h^u]$, where $[x_i^l, x_i^u]$ is the desired temperature range in room i. The approximating Markov chain has state space $\mathcal{Z}_\delta = A_\delta \cup \{\phi\}$, where A_δ is the discretized version of set A with grid size δ and state ϕ represents the set of all states outside A. Set A_δ is determined by partitioning $[x_1^l, x_1^u] \times \cdots \times [x_h^l, x_h^u]$ into rectangular regions, uniformly dividing each interval $[x_1^l, x_1^u]$ into

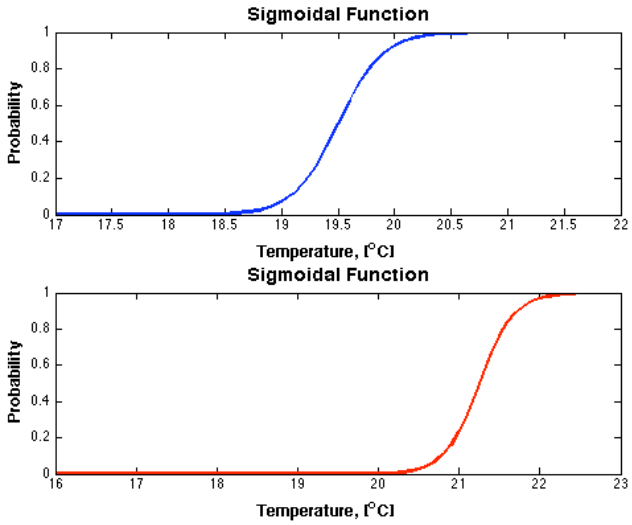


Figure 2: Plots of the sigmoidal functions $\sigma_1(\cdot)$ (top) and $\sigma_2(\cdot)$ (bottom) with $d_1 = 100, \alpha_1 = 19.5$, and $d_2 = 100, \alpha_2 = 21.25$.

l equal parts. The cardinality of the resulting state space $\mathcal{Z}_\delta = A_\delta \cup \{\phi\}$ is then $2^h l^h + 1$. The transition probability function $T_\delta : \mathcal{Z}_\delta \times \mathcal{Z}_\delta \to [0, 1]$ of the approximating Markov chain is defined in (8) based on the transition kernels of the DTSHS modeling the multi-room heating system.

The PCTL formula for the approximating Markov chain that has to be verified is given by

$$\mathbb{P}_{\leq 1-(\epsilon+\frac{\eta}{2})} \left(\texttt{true}\ \mathcal{U}^{\leq N}\ \neg\Phi_{A_\delta} \right). \tag{25}$$

where Φ_{A_δ} characterizes the safe set A_δ. This probabilistic specification can be model-checked by computing $\mathrm{p}_{z_0}(A_\delta)$ for all $z_0 \in A_\delta$. As pointed out at the end of Section 4.2, the complexity involved in model checking formula (25) is then mainly determined by the size and sparsity of the one-step transition probability matrix $\mathrm{P}_\delta[\neg\texttt{true} \vee \neg\Phi_{A_\delta}] = \mathrm{P}_\delta[\neg\Phi_{A_\delta}]$ built from the modified transition probability function $T_\delta[\neg\Phi_{A_\delta}]$ of the approximating Markov chain. Since $\neg\Phi_{A_\delta} = \Phi_{\{\phi\}}$ and ϕ is an absorbing state for the approximating Markov chain, then $T_\delta[\neg\Phi_{A_\delta}]$ coincides with T_δ and, in turn, $\mathrm{P}_\delta[\neg\Phi_{A_\delta}]$ is equal to the one-step transition probability matrix of the approximating Markov chain. The size of this square matrix is given by the cardinality $2^h l^h + 1$ of $\mathcal{Z}_\delta$, whereas its sparsity depends on the kernels T_x (through the noise variance ν^2) and T_q (through the parameters α_i, d_i). The sparsity can also be tuned according to the choice of a tolerance parameter, henceforth denoted with `tol`, by setting equal to zero all the transition probabilities smaller than `tol` (and re-normalizing the resulting probability matrix).

The results reported in this section refer to computations performed on a 3.4 GHz Intel Pentium 4 with a 1GB RAM. The Markov chain approximation scheme is implemented in MATLAB and the one-step transition probability matrix is passed to the MRMC software [22] for model checking. For performance assessment, we consider $\mathrm{p}_{z_0}(A_\delta)$, $z_0 \in A_\delta$, as output of the MRMC software, and plot an interpolated version of it, namely the corresponding estimate of $\mathrm{p}_{s_0}(A)$, $s_0 \in A$, for illustrative purposes.

We start considering the case of $h = 2$ rooms with the parameters listed at the end of Section 5.1. By choosing a discretization level $l = 10$, we obtain an approximating Markov chain with $2^2 \cdot 10^2 + 1 = 401$ states and a one-step transition probability matrix with $401^2 = 160801$ elements. Table 2 reports the number of non-zero one-step transition probabilities for different combinations of the steepness and threshold parameters with $d_1 = d_2$ and $\alpha_1 = \alpha_2$, and $\texttt{tol} = 10^{-5}$. Note that the sparsity of the one-step transition matrix is particularly sensitive to the value taken by d_i. In particular, the matrix sparsity increases with d_i. However, the computational benefits of increased sparsity may be offset by the requirements for finer gridding in this case, since the Lipschitz constant of T_q is directly proportional to d_i (see equation (26)).

$\alpha_i \backslash d_i$	flat	gradual	steep
low	21285	20696	10602
medium	21303	20975	11475
high	21306	20704	11112

Table 2: Number of non-zero one-step transition probabilities out of 160801 ($h = 2$, $l = 10$), when the tolerance parameter is $\texttt{tol} = 10^{-5}$.

Figures 3 and 4 represent the estimate of the invariance probability $\mathrm{p}_{s_0}(A)$ with $s_0 = (q_0, x_0)$. In Figure 3, the steepness and threshold parameters are chosen as gradual and medium for both rooms. The plots represent the invariance probability as a function of x_0 over $[17, 22] \times [16, 23]$, for the four possible values of the initial mode q_0. The running time of the overall approximate model checking procedure was 1.534 seconds, which includes both the time spent for building the approximating Markov chain and that for running the model checking algorithm. Figure 4 represents the average of $\mathrm{p}_{(q_0,x_0)}(A)$ with respect to q_0, as a function of x_0 over $[17, 22] \times [16, 23]$, for the 9 possible combinations of α_i and d_i reported in Table 1. Visual inspection suggests that the invariance probability is maximized when α_i is medium and d_i is steep (second line, right-most plot): this configuration corresponds to sigmoidal functions defining the discrete transition kernel that rapidly saturate (to zero or to one) as soon as the temperature differs from the middle ones.

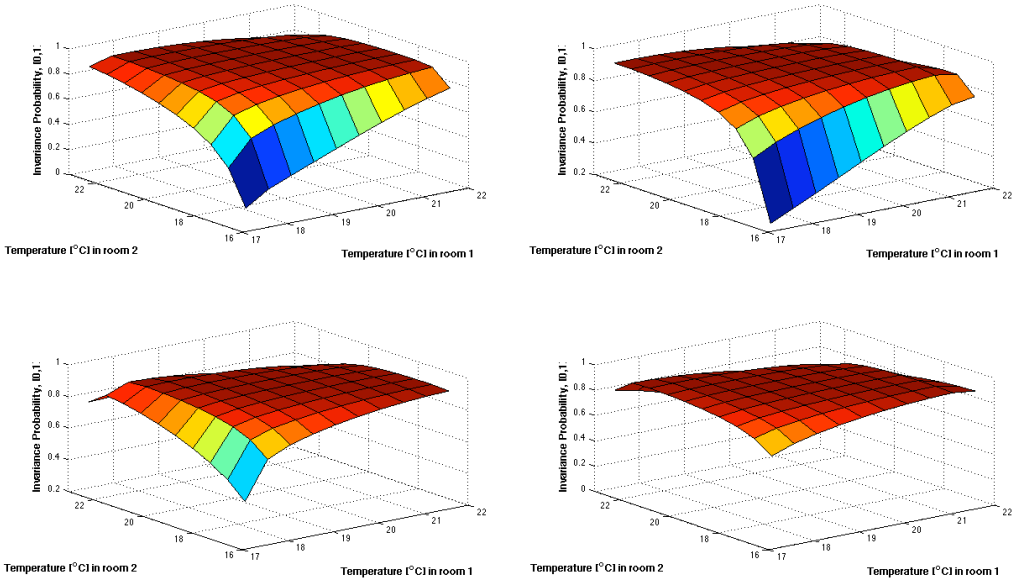
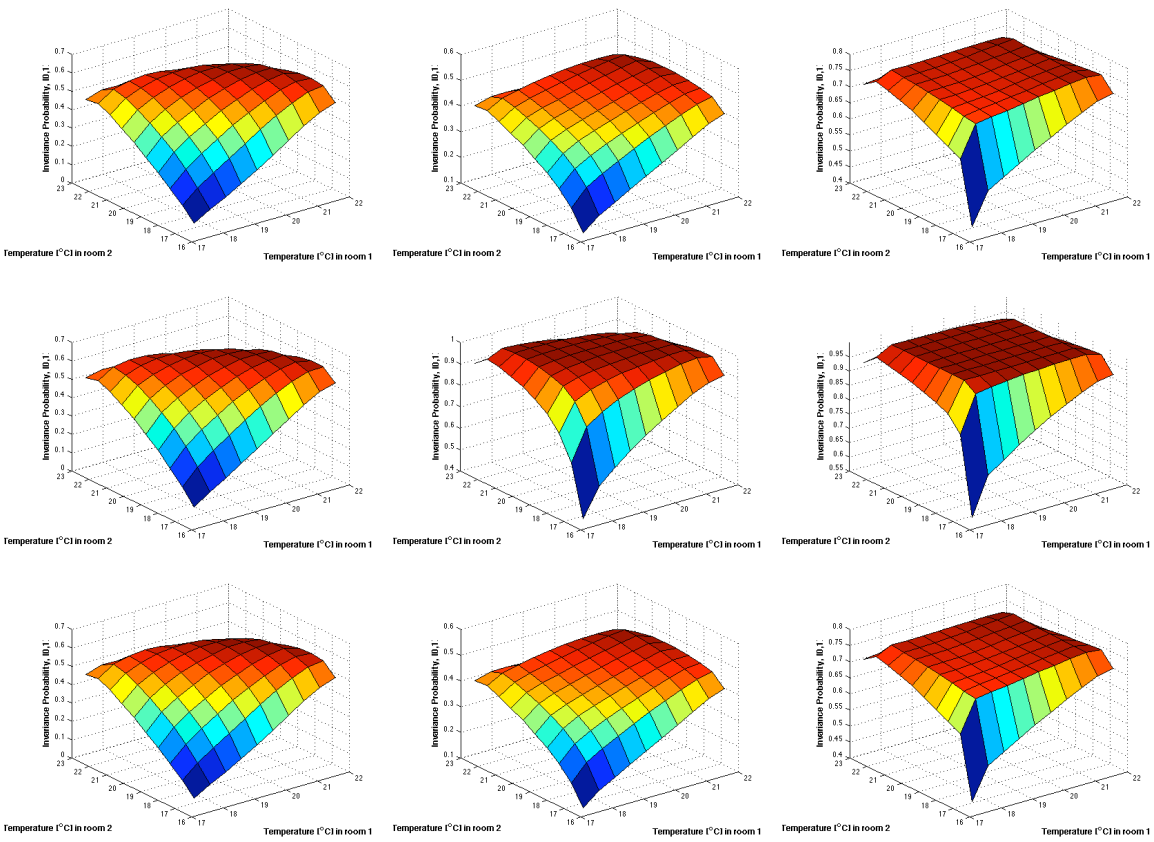


Figure 3: Invariance probability in the time horizon $[0, 50]$ for a 2-room heating system with medium threshold and gradual steepness parameters, discretization level $l = 10$, and $\texttt{tol} = 10^{-5}$. The plots represent the estimate of $\mathrm{p}_{(q_0,x_0)}(A)$ as a function of x_0 over $[17, 22] \times [16, 23]$, for mode $q_0 = (\texttt{OFF}, \texttt{OFF})$ (top-left plot), $q_0 = (\texttt{OFF}, \texttt{ON})$ (top-right plot), $q_0 = (\texttt{ON}, \texttt{OFF})$ (bottom-left plot), and $q_0 = (\texttt{ON}, \texttt{ON})$ (bottom-right plot).

We next evaluate the scalability of the two-stage approximate model checking procedure in terms of both computational time and memory usage as determined by three fundamental parameters: the time horizon length N, the number of rooms h, and the discretization level l. More specifically, we fix two parameters and examine how computing time scales with the other one. Note that if we fix h and l and let N grow, only the computing time involved in the model checking stage increases, since the size of the Markov chain is fixed. The value taken by h and l instead affects the computing times of both the Markov chain approximation and the model checking stages. The computing time heavily depends on the size of the approximating Markov chain, but also on the tolerance parameter `tol` and on the steepness and threshold parameters. The results are reported for $\texttt{tol}=10^{-5}$, and for steepness and threshold parameters respectively set equal to gradual and



low α_i, flat d_i	low α_i, gradual d_i	low α_i, steep d_i
medium α_i, flat d_i	medium α_i, gradual d_i	medium α_i, steep d_i
high α_i, flat d_i	high α_i, gradual d_i	high α_i, steep d_i

Figure 4: Invariance probability in the time horizon $[0, 50]$ for a 2-room heating system with discretization level $l = 10$ and $\texttt{tol} = 10^{-5}$. The plots represent the average of $\mathrm{p}_{(q_0,x_0)}(A)$ with respect to q_0 as a function of x_0 over $[17, 22] \times [16, 23]$, for different values of α_i and d_i (the pairs are chosen to be the same in both rooms), as reported in the corresponding position of the table below.

medium in all rooms.

Table 3 reports the computing time for the two-stage approximate model checking procedure as a function of the time horizon length N in the case of $h = 2$ rooms and a discretization level $l = 10$. Given that, for fixed l and h, N affects only the computing time of the model checking stage, it appears that the bottleneck of the overall approach resides in the Markov chain approximation stage.

Table 4 reports computing time and memory usage as a function of the discretization level l ($N = 50$ and $h = 2$). Memory usage is evaluated in terms of multiples of 8 Bytes since MATLAB uses 8 Bytes to store a double type associated to each element of the transition probability matrix. The results show how the computing time deteriorates as the discretization level l increases.

Table 5 reports computing time and memory usage as a function of the number of rooms h ($N = 50$ and $l = 5$). The parameters defining the transition kernels for the additional rooms have the same values of those used in the 2-room case. The results show a stronger dependence of the computing time on the number of room h than on the discretization level l. This is due to the fact that the size $2^h l^h + 1$ of the approximating Markov chain scales exponentially with h and only polynomially with l. Interestingly, if one considers combinations of

l and h mapping into the same size of the Markov chain, then, the computing time is larger for larger values of l. This is for instance the case for $l = 50$, $h = 2$ and $l = 5$, $h = 4$, respectively corresponding to a computing time of 54.3 min (see Table 4) and 32 min (see Table 5). The reason for this behavior can be found in the different sparsity of the one-step transition probability matrices.

Time horizon length N	10	50	100	500	1000	5000
Computing time	1.532 [sec]	1.534 [sec]	1.535 [sec]	1.544 [sec]	1.554 [sec]	1.656 [sec]

Table 3: Computing time versus time horizon length N ($h = 2$, $l = 10$).

Discretization level l	5	10	20	50
Computing time	0.21[sec]	1.534 [sec]	39.87 [sec]	54.3 [min]
Memory usage ($\times$8 Bytes)	101^2	401^2	1601^2	10001^2

Table 4: Computing time and memory usage versus discretization level l ($h = 2$, $N = 50$).

Number of rooms h	1	2	3	4	5
Computing time	0.08 [sec]	0.21 [sec]	12.6 [sec]	32 [min]	10.8 [hr]
Memory usage ($\times$8 Bytes)	11^2	101^2	1001^2	10001^2	100001^2

Table 5: Computing time and memory usage versus number of rooms h ($N = 50$, $l = 5$).

5.3 Computational complexity

In this section, we determine the size of the approximating Markov chain needed to approximate the probabilistic safety set $S(\epsilon)$ of the multi-room heating system up to some given tolerance η. The size of the approximating Markov chain determines the memory usage requirements posed on the approximate model checking procedure. It represents also a good indicator of the computational effort involved in such a procedure, though the computing time depends not only on the size but also on the sparsity of the one-step transition matrix of the approximating Markov chain. Computations are performed with reference to the case when the temperature limits x_i^l and x_i^u, and the steepness and threshold parameters d_i and α_i are the same for all rooms $i = 1, 2, \ldots, h$. We shall denote these common values as x^l, x^u, d, and α, respectively, and set $\Delta := x^u - x^l$.

The size of the approximating Markov chain for the multi-room heating system is given by $2^h l^h + 1$ where h is the number of rooms and l is the number of intervals in which the temperature range $[x^l, x^u]$ is uniformly divided. We shall now express l as a function of η and of the parameters characterizing the DTSHS model of the multi-room heating system.

Fix $\eta > 0$. Then, according to the second step of Algorithm 1 the grid size parameter δ should satisfy $\delta \leq \frac{\eta}{2\gamma}$ with γ given by Theorem 2. Each region A_q of the safe set A is given by the hypercube $[x^l, x^u]^h$ and is discretized uniformly in l^h hypercubes. Then δ is the diameter of an h-dimensional hypercube of size $\frac{\Delta}{l}$ and satisfies $\delta \leq \sqrt{h}\frac{\Delta}{l}$, which leads to $l \geq \frac{\sqrt{h}\Delta}{\delta} = \frac{2\sqrt{h}\Delta\gamma}{\eta}$. The parameter γ in Theorem 2 is given by $\gamma = N\mathcal{K} = N\big[mh_1 + \lambda\big(h_2 + (m-1)h_3\big)\big]$, where $m = 2^h$ is the cardinality of $\mathcal{Q}$ and $\lambda = \Delta^h$ is the Lebesgue measure of the hypercube $[x^l, x^u]^h$. As for the Lipschitz constants h_1, h_2 and h_3 in Assumption 1, h_2 and h_3 take the same value since $t_x(x'|(q,x)) = r(x'|(q,x),q')$, and they satisfy the following bounds, which are proven in Appendix A:

$$h_1 \leq \sqrt{h}\ \frac{d}{4\alpha}, \tag{26}$$

$$h_2 = h_3 \leq \frac{1}{(2\pi)^{h/2} e^{1/2} \nu^{h+1}} \left\| I + \Sigma \right\|_2 . \tag{27}$$

We then get that

$$\gamma \leq 2^h N \left[\sqrt{h} \frac{d}{4\alpha} + \Delta^h \ \frac{\|I + \Sigma\|_2}{(2\pi)^{h/2} e^{1/2} \nu^{h+1}} \right],$$

which leads to the following conservative bound on l:

$$l \geq \frac{2^{h+1} \sqrt{h} N \Delta}{\eta} \left[\sqrt{h} \frac{d}{4\alpha} + \Delta^h \ \frac{\|I + \Sigma\|_2}{(2\pi)^{h/2} e^{1/2} \nu^{h+1}} \right].$$

As a consequence, the memory usage is of the order of $\backsimeq \frac{2^{h^2} h^{h/2} N^h \Delta^h}{\eta^h} \left[\sqrt{h} \frac{d}{4\alpha} + \Delta^h \ \frac{\|I+\Sigma\|_2}{(2\pi)^{h/2} e^{1/2} \nu^{h+1}} \right]^h$.

From this expression, it is evident that the proposed approach suffers from the curse of dimensionality. In particular, the dependency of the memory requirements on the number of room h is super-exponential. This is not surprising, given that the approximating Markov chain is obtained by gridding the safe set $A = 2^h \times [x^l, x^u]^h$.

6 Concluding remarks

In this paper, we take an initial step toward the development of a fully automatic procedure for the approximate verification of stochastic hybrid systems. We showed how the probabilistic invariance of discrete time stochastic hybrid systems can be studied by building an approximating discrete time Markov chain which can be analyzed using model checking methods. Under certain regularity conditions on the transition and reset kernels of the stochastic hybrid system, the proposed procedure for approximate model checking provides an estimate of the invariance probability together with a certificate of guaranteed accuracy.

The results proposed here are a first step in the desired direction. Several problems need to be overcome to make the method applicable in practice. Some of these problems are practical. In our current implementation, the construction of the approximating Markov chain is done using rather crude, custom-made approximation methods. An efficient implementation of this process which works seamlessly with the highly optimized model checking tools would definitely improve the applicability of the method. On the theoretical front, several challenges have to be addressed, among them developing similar procedures that work with a wider range of properties (beyond reachability and invariance) and methods for dealing with continuous time stochastic hybrid systems. The former will most likely require an appropriate characterization of the properties of interest as the expected value of appropriate cost functions. For the latter, a numerical scheme based on a Markov chain approximation [25] has been recently introduced in [32] for the purpose of reachability analysis of continuous time stochastic hybrid systems; however, these results are confined to a particular class of stochastic hybrid systems and are only asymptotic.

Acknowledgments: Research supported by the European Commission under project iFly FP6-TREN-037180, and by the Swiss National Science Foundation under grant 200021-122072.

A Appendix

A.1 Proof of equation (26)

The Lipschitz constant h_1 in Assumption 1 is bounded by the maximum of the gradient norm of the stochastic kernel $T_q(q'|(q,x))$ in (22) as a function of $x \in \mathbb{R}^h$. The gradient of $T_q(q'|(q,x))$ with respect to x is given by

$$\frac{\partial T_q(q'|(q,x))}{\partial x} = \begin{bmatrix} \dfrac{\partial T_{q,1}(q_1'|(q_1,x_1))}{\partial x_1} \displaystyle\prod_{i=2}^{h} T_{q,i}(q_i'|(q_i,x_i)) \\ \dfrac{\partial T_{q,2}(q_2'|(q_2,x_2))}{\partial x_2} \displaystyle\prod_{i=1,i\neq 2}^{h} T_{q,i}(q_i'|(q_i,x_i)) \\ \vdots \\ \dfrac{\partial T_{q,h}(q_h'|(q_h,x_h))}{\partial x_h} \displaystyle\prod_{i=1}^{h-1} T_{q,i}(q_i'|(q_i,x_i)) \end{bmatrix},$$

$q = (q_1, q_2, \ldots, q_h)$, $q' = (q_1', q_2', \ldots, q_h') \in \mathcal{Q}$, $x = (x_1, x_2, \ldots, x_h) \in \mathbb{R}^h$.

From the definition of $T_{q,i}(q_i'|(q_i,x_i))$ in equation (23) we get

$$\left| \frac{\partial T_{q,i}(q_i'|(q_i,x_i))}{\partial x_i} \right| \leq \max_{y\in\mathbb{R}} \left\{ \frac{d}{dy} \left(\frac{y^d}{\alpha^d + y^d} \right) \right\} = \frac{d}{4\alpha}.$$

Since $\left| \prod_{i=1,i\neq j}^{h} T_{q,i}(q_i'|(q_i,x_i)) \right| \leq 1$, then

$$\left\| \frac{\partial T_q(q'|(q,x))}{\partial x} \right\| \leq \sqrt{\sum_{i=1}^{h} \left(\frac{\partial T_{q,i}(q_i'|(q_i,x_i))}{\partial x_i} \right)^2} \leq \sqrt{h} \frac{d}{4\alpha},$$

which concludes the proof of equation (26).

A.2 Proof of equation (27)

The Lipschitz constants $h_2 = h_3$ are bounded by the maximum of the gradient norm of the density $t_x(x'|(q,x))$ as a function of $x \in \mathbb{R}^h$. From (21) we have that

$$t_x(x'|(q,x)) = \frac{1}{(2\pi)^{h/2}\nu^h} e^{-\frac{\|x' - (x + \Sigma x + \Gamma(q))\|^2}{2\nu^2}}.$$

Then,

$$\frac{\partial t_x(x'|(q,x))}{\partial x} = \frac{1}{(2\pi)^{h/2}\nu^{h+2}} (I + \Sigma)\big(x' - (x + \Sigma x + \Gamma(q))\big) e^{-\frac{\|x' - (x + \Sigma x + \Gamma(q))\|^2}{2\nu^2}}.$$

and, hence,

$$\left\| \frac{\partial t_x(x'|(q,x))}{\partial x} \right\| \leq \frac{1}{(2\pi)^{h/2}\nu^{h+2}} \|I + \Sigma\|_2 \|w\| e^{-\frac{\|w\|^2}{2\nu^2}},$$

where we have set $w := x' - (x + \Sigma x + \Gamma(q))$. Based on the observation that

$$\max_{\beta\in\mathbb{R}} \left\{ \beta e^{-\frac{\beta^2}{2\nu^2}} \right\} = \left[\beta e^{-\frac{\beta^2}{2\nu^2}} \right]_{\beta=\nu} = \nu e^{-\frac{1}{2}},$$

we can conclude that

$$h_2 = h_3 \leq \max_{x', x\in\mathbb{R}^h, q\in\mathcal{Q}} \frac{\partial t_x(x'|(q,x))}{\partial x} \leq \frac{1}{(2\pi)^{h/2} e^{1/2} \nu^{h+1}} \|I + \Sigma\|_2 .$$

References

[1] A. Abate, S. Amin, M. Prandini, J. Lygeros, and S. Sastry. Computational approaches to reachability analysis of stochastic hybrid systems. In A. Bemporad, A. Bicchi, and G. Buttazzo, editors, *Hybrid Systems: Computation and Control*, number 4416 in Lecture Notes in Computer Sciences, pages 4–17. Springer-Verlag, Berlin, 2007.

[2] A. Abate, M. Prandini, J. Lygeros, and S. Sastry. Probabilistic reachability and safety for controlled discrete time stochastic hybrid systems. *Automatica*, 44(11):2724–2734, November 2008.

[3] R. Alur, C. Courcoubetis, N. Halbawachs, T. A. Henzinger, P. H. Ho, X. Nicollin, A. Olivero, J. Sifakis, and S. Yovine. The algorithmic analysis of hybrid systems. *Theoretical Computer Science*, 138:3–34, 1995.

[4] R. Alur and D. Dill. A theory of timed automata. *Theoretical Computer Science*, 126:183–235, 1994.

[5] R. Alur, T.A. Henzinger, G. Lafferriere, and G.J. Pappas. Discrete abstractions of hybrid systems. *Proceedings of the IEEE*, 88(7):971–984, July 2000.

[6] C. Baier and J.-P. Katoen. *Principles of Model Checking*. MIT Press, 2008.

[7] C. Baier, J.-P. Katoen, and H. Hermanns. Approximate symbolic model checking of continous-time Markov chains. In J.C.M. Baeten and S. Mauw, editors, *Concurrency Theory*, number 1664 in Lecture Notes in Computer Sciences, pages 146–162. Springer-Verlag, Berlin, 1999.

[8] D. P. Bertsekas. Convergence of discretization procedures in dynamic programming. *IEEE Transactions on Automatic Control*, 20(3):415–419, 1975.

[9] D. P. Bertsekas and S. E. Shreve. *Stochastic Optimal Control: The Discrete-Time Case*. Athena Scientific, 1996.

[10] H.A.P. Blom and J. Lygeros (Eds.). *Stochastic Hybrid Systems: Theory and Safety Critical Applications*. Number 337 in Lecture Notes in Control and Information Sciences. Springer-Verlag, Berlin, 2006.

[11] M. Bujorianu. Extended stochastic hybrid systems and their reachability problem. In R. Alur and G.J. Pappas, editors, *Hybrid Systems: Computation and Control*, number 2993 in Lecture Notes in Computer Sciences, pages 234–249. Springer-Verlag, Berlin, 2004.

[12] M.L. Bujorianu and J. Lygeros. Reachability questions in piecewise deterministic Markov processes. In O. Maler and A. Pnueli, editors, *Hybrid Systems: Computation and Control*, number 2623 in Lecture Notes in Computer Sciences, pages 126–140. Springer-Verlag, Berlin, 2003.

[13] C.G. Cassandras and J. Lygeros (Eds.). *Stochastic Hybrid Systems*. Number 24 in Control Engineering. CRC Press, Boca Raton, 2006.

[14] E. M. Clarke, O. Grumberg, and D. Peled. *Model Checking*. MIT Press, 1999.

[15] C. Courcoubetis and M. Yannakakis. The complexity of probabilistic verification. *Journal of the ACM*, 42(4):857–907, 1995.

[16] M. H. A. Davis. *Markov Models and Optimization*. Chapman & Hall/CRC Press, London, 1993.

[17] L. de Alfaro. *Formal Verification of Probabilistic Systems*. PhD thesis, Department of Computer Science, Stanford University, 1998.

[18] A. Fehnker and F. Ivančić. Benchmarks for hybrid systems verifications. In R. Alur and G.J. Pappas, editors, *Hybrid Systems: Computation and Control*, Lecture Notes in Computer Science 2993, pages 326–341. Springer Verlag, 2004.

[19] Martin Fränzle, Holger Hermanns, and Tino Teige. Stochastic satisfiability modulo theory: A novel technique for the analysis of probabilistic hybrid systems. In M. Egerstedt and B. Misra, editors, *Hybrid Systems: Computation and Control*, pages 172–186, Berlin, Heidelberg, 2008. Springer-Verlag.

[20] H. Hansson and B. Jonsson. A logic for reasoning about time and reliability. *Formal Aspects of Computing*, 6(5):512–535, 1994.

[21] T. Henzinger, P. Kopke, A. Puri, and P. Varaiya. What's decidable about hybrid automata. *Journal of Computer and System Sciences*, 57:94–124, 1998.

[22] J.-P. Katoen, M. Khattri, and I. S. Zapreev. A Markov reward model checker. In *Quantitative Evaluation of Systems (QEST)*, pages 243–244, Los Alamos, CA, USA, 2005. IEEE Computer Society.

[23] K. Koutsoukos and D. Riley. Computational methods for reachability analysis of stochastic hybrid systems. In J. Hespanha and A. Tiwari, editors, *Hybrid Systems: Computation and Control*, number 3927 in Lecture Notes in Computer Sciences, pages 377–391. Springer-Verlag, Berlin, 2006.

[24] R. P. Kurshan. *Computer-aided verification of coordinating processes; the automata-theoretic approach.* Princeton University Press, 1994.

[25] H. J. Kushner and P.G. Dupuis. *Numerical Methods for Stochastic Control Problems in Continuous Time.* Springer-Verlag, New York, 2001.

[26] M. Kwiatkowska, G. Norman, R. Segala, and J. Sproston. Verifying quantitative properties of continuous probabilistic timed automata. In *CONCUR'00*, number 1877 in Lecture Notes in Computer Sciences, pages 123–137. Springer-Verlag, Berlin, 2000.

[27] A. Lecchini, W. Glover, J. Lygeros, and J. Maciejowski. Monte Carlo optimization for conflict resolution in air traffic control. *IEEE Transactions on Intelligent Transportation Systems*, 7(4):470–482, December 2006.

[28] R. Malhame and C.-Y. Chong. Electric load model synthesis by diffusion approximation of a high-order hybrid-state stochastic system. *IEEE Transactions on Automatic Control*, 30(9):854–860, 1985.

[29] Z. Manna and A. Pnueli. *The Temporal Logic of Reactive and Concurrent Systems: Specification.* Springer-Verlag, Berlin, 1992.

[30] Z. Manna and A. Pnueli. *Temporal Verification of Reactive Systems: Safety.* Springer-Verlag, New York, 1995.

[31] S. Prajna, A. Jadbabaie, and G.J. Pappas. A framework for worst-case and stochastic safety verification using barrier certificates. *IEEE Transactions on Automatic Control*, 52(8):1415–1428, 2007.

[32] M. Prandini and J. Hu. Stochastic reachability: Theory and numerical approximation. In C.G. Cassandras and J. Lygeros, editors, *Stochastic hybrid systems*, Automation and Control Engineering Series 24, pages 107–138. Taylor & Francis Group/CRC Press, 2006.