A two-step scheme for approximate model checking of stochastic hybrid systems ⋆

A. Abate * **J.P. Katoen** ** **J. Lygeros** *** **M. Prandini** ****

* *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

Abstract: In this paper we describe a two-step scheme for approximate model checking of discrete time stochastic hybrid systems. In the first step, the stochastic hybrid system is approximated by a finite state Markov chain. In the second step, the Markov chain is model checked for the desired property. In particular, we consider the probabilistic invariance property and show that, under certain regularity conditions, the invariance probability computed using the approximating Markov chain converges 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.

1. INTRODUCTION

Stochastic hybrid systems allow one to model the interaction between continuous dynamics, discrete dynamics and probabilistic uncertainty. Because of their versatility, stochastic hybrid systems have emerged as a powerful framework for capturing the intricacies of complex systems. Various classes of stochastic hybrid models have been introduced and numerous case studies in the literature have illustrated their potential in diverse application domains such as control of telecommunication networks, air traffic, manufacturing, biology and finance (see Blom and J. Lygeros (Eds.) [2006], Cassandras and J. Lygeros (Eds.) [2006] for an overview). Motivated by this, considerable research effort has been devoted to the development of modelling, analysis and control methods for stochastic hybrid systems. Part of the research in this area has been driven by computer scientists, in the area of formal methods. Different approaches have originated in the engineering literature, in particular in the area of automatic control. Each of the approaches has its own advantages and disadvantages and has been applied successfully to several application areas. However, synergies still need to be deeply explored and fully exploited in order to address the challenges posed by real-life, large scale applications.

In particular, the development of computational tools is crucial for the application to practical problems of the theoretical results that have emerged in the area 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 and, at the same time, should be versatile and efficient enough to be used on realistic applications.

Many of the computational tools proposed in the area of stochastic hybrid systems are based on numerical techniques. These involve either imposing a grid on the state space, thus turning an infinite state problem into an approximate finite state one (see, e.g., Abate et al. [2007], Prandini and Hu [2006]), or carrying out Monte-Carlo simulations to obtain empirical estimates of quantities such as expected values of reach probabilities (Lecchini et al. [2006]). Even though computational tools based on numerical methods typically come with explicit approximation guarantees, 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 with symbolic computation techniques such as model-checking methods that can be used to test a wider range of properties and that have been optimized for computational efficiency.

Model checking methods are model-based verification techniques that provide the means to algorithmically check whether the model of a system satisfies different kinds of properties related to its evolution in time and expressed in an appropriate temporal logic (Baier and Katoen [2008], Clarke et al. [1999]). In particular, probabilistic model checking has been introduced quite recently as an automated technique for verifying properties specified in terms of probability that a certain condition is satisfied. The adopted stochastic models are discrete and continuous time Markov chains and the properties are specified in terms of some probabilistic logic like Probabilistic Computation Tree Logic (PCTL, Hansson and Jonsson [1994]). A detailed overview is provided in Baier and Katoen [2008]. During the last decade, effective and efficient probabilis-

⋆ Research supported by the European Commission under the MoVeS project, FP7-ICT-2009-257005.

tic model-checking algorithms for Markov chains have been developed and implemented in software tools such as PRISM (Hinton et al. [2006]) and MRMC (Katoen et al. [2009]). Model checking for stochastic systems with a hybrid state space is a rather unexplored field.

The aim of the present paper is to take a first step toward the automatic verification of complex properties for stochastic hybrid systems by combining numerical computation based on gridding and finite approximation with model checking tools developed to test temporal logic properties for finite state Markov chains. The idea is as simple as follows. 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 are then encoded in PCTL and analyzed using a model checker. Guarantees on the probability that the original stochastic hybrid system satisfies the property of interest are provided. Scalability of the approach can be further enhanced by constructing first an approximate abstraction of the stochastic hybrid system as in Julius and Pappas [2009], and then performing the proposed automatic verification procedure on the simpler abstracted model.

Results in this paper are confined to discrete time stochastic hybrid systems and the finite time invariance property. Theoretical and practical challenges related to the extension and the application of the approximate model checking approach are discussed in the conclusions.

2. STOCHASTIC HYBRID SYSTEM MODEL

We consider a discrete time stochastic hybrid system (DTSHS) as defined in Abate et al. [2008] but 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 a 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 $\mathbb{R}^{n(q)}$. $\mathcal{S}$ 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 (Davis [1993]).

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. Definitions 1 and 2 formalize this description.

Definition 1. (DTSHS). A discrete time stochastic hybrid system is a collection $\mathcal{H} = (\mathcal{Q}, n, \mathit{Init}, T_x, T_q, R)$, 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 $\mathbb{R}^{n(q)}$;
- $\mathit{Init} : \mathcal{B}(\mathcal{S}) \to [0, 1]$ 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 the DTSHS over a finite time horizon $[0, N]$ and define its semantics algorithmically.

Definition 2. (Execution of a DTSHS). Consider a DTSHS $\mathcal{H} = (\mathcal{Q}, n, \mathit{Init}, T_x, T_q, R)$ 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 $\{s_k = (q_k, x_k),\ k \in [0, N]\}$ are obtained according to the following algorithm:

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

Interestingly, a DTSHS $\mathcal{H} = (\mathcal{Q}, n, \mathit{Init}, T_x, T_q, R)$ defines a Markov process with state space $\mathcal{S}$ and transition probability kernel $T_s : \mathcal{B}(\mathcal{S}) \times \mathcal{S} \to [0, 1]$ given by

$$T_s(\{q'\} \times A_{q'}\,|s) = \begin{cases} T_x(A_{q'}|s)T_q(q'|s), & q' = q \\ R(A_{q'}|s, q')T_q(q'|s), & q' \neq q, \end{cases} \quad (1)$$

for all $s = (q, x) \in \mathcal{S}$, $q' \in \mathcal{Q}$, and $A_{q'} \in \mathcal{B}(\mathbb{R}^{n(q')})$.

Thus, 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 [Bertsekas and Shreve, 1996, Proposition 7.45]. In the sequel, we shall use the notation P_{s_0} for the probability measure associated with the deterministic initial condition $s_0 \in \mathcal{S}$, i.e., $\mathrm{Init}(\cdot) = \delta_{s_0}(\cdot)$.

3. PROBABILISTIC INVARIANCE PROBLEM

Consider a compact Borel set $A \in \mathcal{B}(S)$, representing a 'safe' set. The probability that the execution associated with the initial condition $s_0 \in \mathcal{S}$ will remain within A during the time horizon $[0, N]$ is defined as

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

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. For a given $\epsilon \in (0, 1]$, we can define as the *probabilistic safe set* with safety level ϵ the set

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

of those initial conditions s_0 that are safe with an ϵ probabilistic guarantee. Our goal is setting up a computation procedure to determine $S(\epsilon)$.

Let $\mathbf{1}_C : \mathcal{S} \to \{0, 1\}$ denote the indicator function of set $C \subseteq \mathcal{S}$: $\mathbf{1}_C(s) = 1$, if $s \in C$, and $\mathbf{1}_C(s) = 0$, if $s \notin C$. Then, it is easily seen that $\mathrm{p}_{s_0}(A)$ in (2) can be expressed as $\mathrm{p}_{s_0}(A) = E_{s_0}\left[\prod_{k=0}^{N} \mathbf{1}_A(\mathbf{s}(k))\right]$, where E_{s_0} denotes the expectation with respect to the probability measure P_{s_0}.

Define the functions $V_k : \mathcal{S} \to [0, 1]$, $k = 0, 1, \ldots, N$, such that $V_k(s)$ 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. Then, $V_0(s) = E_s[\prod_{l=0}^{N} \mathbf{1}_A(\mathbf{s}(l))]$, $s \in \mathcal{S}$, 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 $S(\epsilon) = \{s_0 \in \mathcal{S} : V_0(s_0) \geq \epsilon\}$.

According to the dynamic programming terminology, we call $V_k(s)$ the value function at time k. The following result states that the value functions can be determined through a backward recursive procedure.

Proposition 1. (Abate et al. [2008], Lemma 1). The value functions $V_k : \mathcal{S} \to [0, 1]$, $k = 0, 1 \ldots, N - 1$, can be computed through the following backward recursion:

$$V_k(s) = \mathbf{1}_A(s) \int_{\mathcal{S}} V_{k+1}(s_{k+1}) T_s(ds_{k+1}|s), \; s \in \mathcal{S}, \quad (4)$$

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

Since an explicit analytic solution to the recursion in equation (4) is generally impossible to find, the computational aspects to the problem are of key importance to its implementation. In Section 4 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 $V_k(s) = 0$, $s \in \mathcal{S} \setminus A$, for any $k \in [0, N]$. As a consequence, the recursive equation (4) in Proposition 1 can be restricted to the compact set A:

$$\begin{aligned} &V_k(s) = \int_A V_{k+1}(s_{k+1}) T_s(ds_{k+1}|s), \; s \in A, \; k < N, \\ &V_N(s) = 1, s \in A. \end{aligned} \quad (5)$$

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 of $\mathcal{H}$, 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 approximate numerical solution.

4. APPROXIMATE MODEL CHECKING

In this section, we describe a computational procedure to determine a conservative approximation of the safe set $S(\epsilon)$ defined in (3) of a DTSHS. This procedure involves two steps: 1. building an approximating finite state Markov chain for the DTSHS, and 2. applying the model checking algorithm for verifying the logical specification of the probabilistic invariance property for the Markov chain.

4.1 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 $A_q \in \mathcal{B}(\mathbb{R}^{n(q)})$. 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.

The state space of the stochastic automaton that approximates the original DTSHS 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:

$$T_\delta(z'|z) = \begin{cases} T_s(\Xi(z')|z), & z' \in A_\delta \text{ and } z \in A_\delta \\ 1 - \sum_{\bar{z} \in A_\delta} T_s(\Xi(\bar{z})|z), & z' = \phi \text{ and } z \in A_\delta \\ 1, & z' = z = \phi \\ 0, & z' \in A_\delta \text{ and } z = \phi, \end{cases} \quad (6)$$

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 discrete time Markov chain (DTMC) $\{\mathbf{z}(k), k \in [0, N]\}$ 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.

4.2 Finite state Markov chain model checking

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 such that the corresponding DTMC executions satisfy that formula.

Logical specification of probabilistic invariance: According to the PCTL, the probabilistic invariance property for a DTMC with safe set $D \subset \mathcal{Z}$ can be expressed by the formula

$$\mathbb{P}_{\geq\epsilon}\left(\Box^{\leq N}\Phi_D\right), \tag{7}$$

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". 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 (see Abate et al. [2008]) can be expressed as:

$$\mathbb{P}_{\geq\epsilon}\left(\Box^{\leq N}\Phi_D\right) \ \equiv\ \mathbb{P}_{\leq 1-\epsilon}\left(\Diamond^{\leq N}\Phi_{\bar{D}}\right). \tag{8}$$

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 ($\mathcal{U}^{\leq N}$). 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). Then, $\Diamond^{\leq N}\Phi$ can be defined as $\texttt{true}\,\mathcal{U}^{\leq N}\Phi$, and, based on (8), the probabilistic invariance property can be expressed in terms of the bounded-until operator as

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

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 a state $\bar{z}$ satisfying formula (7).

Model-checking algorithm: The inputs to the model-checking algorithm for PCTL over DTMCs are a stochastic automaton 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 $Sat(\Phi)$. 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, $\sim$ being a binary comparison operator such as $<$, $\leq$, $>$, $\geq$. We concentrate on 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: $Sat\big(\mathbb{P}_{\sim\epsilon}\left(\Phi\,\mathcal{U}^{\leq N}\,\Psi\right)\big) = \{z \in \mathcal{Z} : \pi_0(z) \sim \epsilon\}$. We next show 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:

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

Clearly, this modified transition probability function makes all the states satisfying Υ absorbing. For model-checking 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 $\mathrm{P}[\neg\Phi \vee \Psi]$. Deter-

mining 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: $\mathrm{P}[\neg\Phi\vee\Psi]\cdot(\ldots(\mathrm{P}[\neg\Phi\vee\Psi]\cdot e_\Psi))$.

Approximation of the probabilistic safe set: The probabilistic safe set $S(\epsilon)$ in (3) can be approximated through the following procedure.

Algorithm 1. (probabilistic safe set approximation).

(1) select $\eta>0$ such that $\frac{\eta}{2}\in(0,1-\epsilon)$;
(2) construct the approximating Markov chain with grid size δ according to the procedure in Section 4.1;
(3) use the model checker to compute $Z_\delta(\epsilon+\frac{\eta}{2})=\left\{z_0\in\mathcal{Z}_\delta:\ z_0\models\mathbb{P}_{\leq 1-(\epsilon+\frac{\eta}{2})}\left(\texttt{true}\,\mathcal{U}^{\leq N}\,\neg\Phi_{A_\delta}\right)\right\}$;
(4) define the approximating safe set as

$$\hat{S}_\eta(\epsilon)=\left\{s_0\in\mathcal{S}:\ \xi(s_0)\in Z_\delta(\epsilon+\frac{\eta}{2})\right\}.$$

We next show that δ can be chosen so as to guarantee a certain quality of the approximated safe set.

5. ANALYSIS OF THE APPROXIMATION 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

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

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. The proof is inspired by Abate et al. [2007], Bertsekas [1975].

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.

Theorem 1. Under Assumption 1, the value function $V_k:\mathcal{S}\to[0,1]$ satisfies the Lipschitz condition over A:
$|V_k((q,x))-V_k((q,x'))|\leq\mathcal{K}\|x-x'\|$, $\forall\,(q,x),\,(q,x')\in A$, 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.

The proof is omitted due to space limitations.

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 stochastic automaton 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 (6). 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 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

$$\begin{aligned}&V_{\delta,k}(z)=\sum_{z'\in A_\delta}T_\delta(z'|z)V_{\delta,k+1}(z'),\ \ z\in A_\delta,\,k<N,\\&V_{\delta,N}(z)=1,\ \ z\in A_\delta,\end{aligned}\quad(10)$$

which is the discretized version of (5).

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

$$|V_k(s)-\hat{V}_k(s)|\leq(N-k)\mathcal{K}\delta,\quad(11)$$

holds for any $k=0,1,\ldots,N$. The claim then follows by setting $k=0$ in equation (11) 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 (11) 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

$$\begin{aligned}|V_k(s)-\hat{V}_k(s)|=|V_k(s)-\hat{V}_k(\xi(s))|\leq|V_k(s)-V_k(\xi(s))|\\+|V_k(\xi(s))-\hat{V}_k(\xi(s))|,\ s\in A.\end{aligned}\quad(12)$$

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 (5) and (10), and the definition of the approximating Markov chain transition probability function (6), 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 (12), the proof of (11) is completed. ∎

Note that the quality of the approximation improves as the grid size 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 length N grows, the approximation error propagates. This is taken into account by the constant γ in Theorem 2, which grows linearly with N. Though the bound in Theorem 2 is conservative, it holds uniformly over A. This allows one to approximate the probabilistic safe set $S(\epsilon)$ defined in (3) by model checking the invariance property of the approximating finite state Markov chain.

Theorem 3. Under Assumption 1, for any $\epsilon \in (0,1)$, the safe set approximation $\hat{S}_\eta(\epsilon)$ obtained through Algorithm 1 with $\delta \leq \frac{\eta}{2\gamma}$ satisfies $S(\epsilon+\eta) \subseteq \hat{S}_\eta(\epsilon) \subseteq S(\epsilon)$.

Proof: By Theorem 2 and $\gamma\delta \leq \frac{\eta}{2}$, we have that

$$|\mathrm{p}_{s_0}(A) - \mathrm{p}_{\delta,z_0}(A_\delta)| \leq \gamma\delta \leq \frac{\eta}{2}, \quad z_0 = \xi(s_0), s_0 \in A. \quad (13)$$

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, $z_0 \models \mathbb{P}_{\leq 1-(\epsilon+\frac{\eta}{2})}\left(\mathtt{true}\ \mathcal{U}^{\leq N} \neg\Phi_{A_\delta}\right)$. Since $\mathbb{P}_{\leq 1-(\epsilon+\frac{\eta}{2})}\left(\mathtt{true}\ \mathcal{U}^{\leq N} \neg\Phi_{A_\delta}\right)$ 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 $\mathrm{p}_{\delta,z_0}(A_\delta) \geq \epsilon + \frac{\eta}{2}$. This bound combined with (13) 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 (13), $\mathrm{p}_{\delta,z_0}(A_\delta) \geq \epsilon + \frac{\eta}{2}$ with $z_0 = \xi(s_0)$. This 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 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.

6. CONCLUDING REMARKS

In this paper, 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.

An efficient implementation of the construction of the approximating Markov chain which works seamlessly with the highly optimized model checking tools has to be conceived for an effective application 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 and for continuous time stochastic hybrid systems. Work in the former direction has been performed in Ramponi et al. [2010] by developing a dynamic programming approach to encode PCTL properties through 'value functions'. For the latter, a numerical scheme has been introduced in Prandini and Hu [2006] for reachability analysis of a class of continuous time stochastic hybrid systems; however, results are only asymptotic.

REFERENCES

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.

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.

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

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

D. P. Bertsekas and S. E. Shreve. *Stochastic optimal control: the discrete-time case*. Athena Scientific, 1996.

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.

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

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

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

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

A. Hinton, M. Kwiatkowska, G. Norman, and D. Parker. PRISM: A tool for automatic verification of probabilistic systems. In *Tools and Algorithms for the Construction and Analysis of Systems*, volume 3920 of *LNCS*, pages 441–444. Springer, 2006.

A.A. Julius and G.J. Pappas. Approximate abstraction of stochastic hybrid systems. *IEEE Trans. Automatic Control*, 54(6):1193–1203, 2009.

J.-P. Katoen, I. S. Zapreev, E. M. Hahn, H. Hermanns, and D. N. Jansen. The ins and outs of the probabilistic model checker MRMC. In *Conference on Quantitative Evaluation of Systems*, pages 167–176. IEEE Computer Society, 2009.

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.

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.

F. Ramponi, D. Chatterjee, S. Summers, and J. Lygeros. On the connections between PCTL and dynamic programming. In *Hybrid Systems: Computation and Control*, pages 253–262. ACM, 2010.