Verifying Stochastic Hybrid Systems with Temporal Logic Specifications via Model Reduction

YU WANG, University of Florida
NIMA ROOHI, Amazon
MATTHEW WEST, MAHESH VISWANATHAN, and GEIR E. DULLERUD,
University of Illinois at Urbana-Champaign

We present a scalable methodology to verify stochastic hybrid systems for inequality linear temporal logic (iLTL) or inequality metric interval temporal logic (iMITL). Using the Mori–Zwanzig reduction method, we construct a finite-state Markov chain reduction of a given stochastic hybrid system and prove that this reduced Markov chain is approximately equivalent to the original system in a distributional sense. Approximate equivalence of the stochastic hybrid system and its Markov chain reduction means that analyzing the Markov chain with respect to a suitably strengthened property allows us to conclude whether the original stochastic hybrid system meets its temporal logic specifications. Based on this, we propose the first statistical model checking algorithms to verify stochastic hybrid systems against correctness properties, expressed in iLTL or iMITL. The scalability of the proposed algorithms is demonstrated by a case study.

CCS Concepts: • **Computer systems organization** → **Embedded and cyber-physical systems**; • **Security and privacy** → *Formal methods and theory of security*; • **Theory of computation** → *Abstraction;*

Additional Key Words and Phrases: Cyber-physical systems, statistical model checking, approximate bisimulation

ACM Reference format:
Yu Wang, Nima Roohi, Matthew West, Mahesh Viswanathan, and Geir E. Dullerud. 2021. Verifying Stochastic Hybrid Systems with Temporal Logic Specifications via Model Reduction. *ACM Trans. Embedd. Comput. Syst.* 20, 6, Article 113 (November 2021), 27 pages.
https://doi.org/10.1145/3483380

1 INTRODUCTION

Modern control systems achieve high-level autonomy by controlling complex physical processes with powerful embedded computers [46]. These integrated systems typically have hybrid dynamics due to interaction between continuous-state physical dynamics, discrete-state cyber dynamics, and random system/environment noise. Common dynamic models for these cyber-physical systems

Authors' addresses: Y. Wang, University of Florida, 633 Gale Lemerand Dr, Gainesville, FL 32603; email: yu.wang094@duke.edu; N. Roohi, Amazon, 2127 7th Ave, Seattle, WA 98121; email: nroohi@amazon.edu; M. West, University of Illinois at Urbana-Champaign, 1206 W Green St, Urbana, IL 61801; email: mwest@illinois.edu; M. Viswanathan, University of Illinois at Urbana-Champaign, 201 Goodwin Ave, Urbana, IL, 61801; email: vmahesh@illinois.edu; G. E. Dullerud, University of Illinois at Urbana-Champaign, 1308 W Main St, Urbana, IL, 61801; email: dullerud@illinois.edu.

1539-9087/2021/11-ART113 $15.00
https://doi.org/10.1145/3483380

are discrete-time or continuous-time stochastic hybrid systems [13, 35, 45, 68]. Such models are widely used in various applications such as automobile powertrains [38, 58], smart grids [63], and chemical [34] or biological systems [36, 60]. To assure functionality in these applications, a foundational problem is the verification of general specifications [3, 20, 27, 42, 65].

For automated system verification, a common approach is model checking [18]. The idea is to use temporal logics (e.g., **computation tree logic** (**PCTL**) [33] and **continuous stochastic logic** (**CSL**) [10] to formally express the specifications of interest and develop computer algorithms to verify these formal specifications. This article focuses on verifying specifications in **inequality linear temporal logic** (**iLTL**) [43] or **inequality metric interval temporal logic** (**iMITL**) on discrete- or continuous-time stochastic hybrid systems, respectively. These logics can express many important specifications for distributed systems and networks [44]. In them, the atomic propositions are functional inequalities over the distributions on the system state space, and the temporal operators are the same as the standard **linear temporal logic** (**LTL**) [52] or **metric interval temporal logic** (**MITL**) [9]. The logics iLTL and iMITL are incomparable to PCTL and CSL. The latter reasons about properties of a random system path, e.g., whether the probability of never reaching an unsafe state is >0.5. Whereas iLTL and iMITL view the random path of the stochastic hybrid system as a time-evolving distribution and reason about properties about this distribution, e.g., whether the probability of the (current) state being safe is always >0.5. Such a specification is not expressible in PCTL and CSL (nor is the example specification of PCTL and CSL expressible in iLTL and iMITL).

For model checking temporal logic specifications, there are two approaches: analytic and statistical. Analytic model checking computes the probabilities of the properties of interest using the system dynamics [10]. Previously, iLTL specifications are checkable on finite-state Markov chains by this approach [44]. But such a method is not scalable to stochastic hybrid systems due to their hybrid state space and stochasticity. On the other hand, statistical model checking infers the probabilities by sampling with provable probabilistic guarantees such as the confidence/significance level of the verification results [6, 47, 48]. Such probabilistic guarantees cannot be attained by Bayesian statistical model checking [37, 82]. Previously, it has been shown how PCTL and CSL specifications are checkable on finite-state Markov chains by this approach [20, 58, 81]. However, statistical model checking is not directly applicable to iLTL and iMITL, since these logics reason about properties of distributions. Checking them on stochastic hybrid systems requires approximating time-evolving hybrid distributions (particularly in continuous time) with finite samples. Although direct simulation of single executions of stochastic hybrid systems with bounded error is possible [62], using finite samples to approximate these hybrid distributions with provable probabilistic guarantees remains challenging.

Our main contribution is to propose a statistical model checking method for iMITL/iLTL specifications on continuous-/discrete-time stochastic hybrid systems with provable probabilistic guarantees. Using the Mori–Zwanzig model reduction method [12, 15], we build an approximate equivalence relation between continuous-/discrete-time stochastic hybrid systems and finite-state continuous-/**discrete-time Markov chains** (**DTMC**)for iMITL/iLTL specifications. To verify whether such a specification is satisfied (or violated) on the stochastic hybrid system, it suffices to verify whether a slightly strengthened (or weakened) specification is satisfied (or violated) on the Markov chain. (We also prove the converse, although this is not our major purpose.) Then, considering that the Markov chain can be large and unscalable for previous analytic model checking methods [44], we propose a new statistical model checking method that can verify iMITL/iLTL on continuous-/DTMCs. Since the Markov chains have finite states, we can use statistical inference methods from [16] to provide provable probabilistic guarantees on the verification results.

The approximate equivalence relation via the Mori–Zwanzig method is similar in spirit to the simulation/bisimulation relation [19, 31], where a complex system model is abstracted by a simple one while exactly or approximately preserving the truth value of the specifications of interest. This approach is applicable to both non-stochastic [7, 17, 41, 51, 56, 66, 76] and stochastic [14, 30, 49, 50, 70, 71] system models. Previous studies have shown that approximate simulation can be built between finite-state Markov chains, Markov processes, and discrete-time stochastic hybrid systems for PCTL specifications [2–4, 22, 39, 70, 71, 79, 80]. However, building simulation relation between continuous-time stochastic hybrid systems and finite-state Markov chains is still challenging in general for CSL specifications [30] because PCTL and CSL reason about properties over paths. The paths of stochastic hybrid systems (particularly continuous-time ones) can exhibit many more complex behaviors than finite-state models (e.g., Markov chains), thus building (approximate) simulations between them is difficult. On the other hand, since iLTL and iMITL only care about properties on distributions, building simulation relations between stochastic hybrid systems and finite-state Markov chains is possible using the Mori–Zwanzig method, even though the systems are very different on the paths.

Similar to previous methods [2, 3], the Mori–Zwanzig model reduction is performed via partitioning the state space, although the metric used for defining equivalence is different. Our model reduction method can be viewed as a generalization of [5, 26] to continuous-time and to temporal logic specifications in iLTL and iMITL. The approximate equivalence by Mori–Zwanzig reduction is similar in spirit to the results first established for non-stochastic, stable, hybrid systems [29, 53, 66], and later extended to stochastic dynamical systems [79, 80]. When compared to [79, 80], we consider a more general class of stochastic hybrid systems that have multiple modes and jumps with guards and resets. Second, our reduced system is a Markov chain, whereas in [79, 80] the stochastic system is approximated by a non-stochastic model. Accordingly, our notion of distance between the stochastic hybrid system and the reduced system is different. Finally, our Mori–Zwanzig reduction method is different from the model order reduction in classic control theory [23]. The former directly reduces the continuous part of the state space to finite states, while the latter only reduces the state dimension.

Since the reduced system, even though finite-state, is likely to have a large number of states, we use a statistical approach to verification [78] as opposed to a symbolic one. In statistical model checking, the model is simulated multiple times, and the drawn simulations are analyzed to see if they constitute statistical evidence for the correctness of the model. Statistical model checking algorithms have been developed for logic that reasons about the probability of path properties [6, 47, 48]. However, since our logic iLTL/iMITL reasons about the properties of time-evolving distributions, we cannot leverage these algorithms. Thus, we develop new statistical model checking algorithms for temporal logics (over discrete and continuous time) that reason about sequences of distributions.

For the scalability of our approach, the main complexity is to perform the integrations of the system dynamic function on the partitions during the Mori–Zwanzig reduction. For common non-linear dynamics (e.g., polynomial, sinusoidal, and exponential), the integrations typically have closed-form solutions, and so are easy to compute. Also, due to the density of polynomial dynamics on any compact domain, we can use them to approximate general non-linear dynamics with error bounds. Alternatively, general non-linear dynamics can also be computed by Monte-Carlo integrations with bounded statistical errors [55]. Our approach successfully verified [58] the Toyota powertrain system with 10 variables [38]; furthermore, the case study of this work demonstrates that our approach can handle general iMITL specifications on stochastic hybrid systems with up to 40 variables.

Our approach combines the Mori–Zwanzig model reduction method with statistical model checking and applies to both continuous-/discrete-time stochastic hybrid systems. It unifies our previous papers [73–75], where discrete-time stochastic hybrid systems are studied in [74]; continuous-time non-hybrid systems are studied in [73]; and continuous-time stochastic hybrid systems are studied in [75] without numerical evaluations. This work also provides a case study to demonstrate the scalability of our statistical verification algorithms.

The rest of the article is organized as follows. In Section 2, we introduce the problem setup. In Section 3, we introduce the Mori–Zwanzig method to reduce continuous-time stochastic hybrid systems into Markov chains. In Section 4, we propose a statistical model checking algorithm for iMITL specifications on the **continuous-time Markov chains** (**CTMC**). Then, we apply the same procedure of Sections 3 and 4 to verify iLTL specification on discrete-time stochastic hybrid systems in Section 5. The scalability of the proposed algorithms is demonstrated by a case study in Section 6. Finally, we conclude in Section 7.

2 PROBLEM FORMULATION

We denote the set of natural, rational, non-negative rational, real, positive real, and non-negative real numbers by $\mathbb{N}$, $\mathbb{Q}$, $\mathbb{Q}_{\geq 0}$, $\mathbb{R}$, $\mathbb{R}_{>0}$ and $\mathbb{R}_{\geq 0}$, respectively. We denote the essential supremum by ess sup. For $n \in \mathbb{N}$, let $[n] = \{1, 2, \dots, n\}$. For any set $\mathbb{S}$, let $\mathbb{S}^{\omega}$ be the set of infinite sequences in $\mathbb{S}$. For $s \in \mathbb{S}^{\omega}$, let s_i be the ith element in the sequence. For a finite set A, we denote the cardinality by $|A|$ and its power set by 2^A. The empty set is denoted by $\emptyset$. For $X \subseteq \mathbb{R}^d$, we denote the boundary of X by ∂X. The symbols $\mathbb{P}$ and $\mathbb{E}$ are used for the probability and expected value, respectively.

2.1 Stochastic Hybrid System

We follow the formal definitions of continuous-time stochastic hybrid systems in [45, 64, 67–69] as shown in Figure 1 with a Fokker–Planck formulation and interpretation of the model.

2.1.1 Continuous-time Stochastic Hybrid System. We denote the continuous and discrete states by $x \in \mathbb{R}^d$ and $q \in Q$, respectively, where $Q = \{q_1, \dots, q_m\}$ is a finite set. We call the combination (q, x) the state of the system, and the product set $\mathbb{X} \subseteq Q \times \mathbb{R}^d$ the state space. For each $q \in Q$, the state of the system flows in $\mathbf{A}_q \subseteq \mathbb{R}^d$ and jumps forcedly on hitting the boundary $\mathbf{A}_q$. We assume that each $\mathbf{A}_q$ is open and bounded, and the boundaries $\partial\mathbf{A}_q$ are second-order continuously differentiable. On the flow set, the state x of the system evolves by a stochastic differential equation

$$\mathrm{d}x = f(q, \mathbf{x})\mathrm{d}t + g(q, \mathbf{x})\mathrm{d}B_t, \tag{1}$$

where q and $\mathbf{x}$ are random processes describing the stochastic evolution of the discrete and continuous states, and B_t is the standard n-dimensional Brownian motion. The vector-valued function f specifies the drift of the state, and the matrix-valued function g describes the intensity of the diffusion [40, 54]. In (1), we assume that $f(q, \cdot)$ and $g(q, \cdot)$ are locally Lipschitz continuous. Meanwhile, the system jumps spontaneously by a non-negative integrable rate function $r(q, x)$ inside $\mathbf{A}_q$. The probability distribution of the target of both spontaneous and forced jumps (as they happened on different domains) is given by a non-negative integrable target distribution $h(q', x', q, x)$, satisfying

$$\sum_{q \in Q} \int_{\mathbf{A}_q} h(q', x', q, x)\mathrm{d}x' = 1. \tag{2}$$

2.1.2 Fokker–Planck Equation. The probability distribution $F(t, q, x)$ of the state of the system in the flow set is determined by the Fokker–Planck equation, which can be derived in the same way as that for jump-diffusion processes [32],

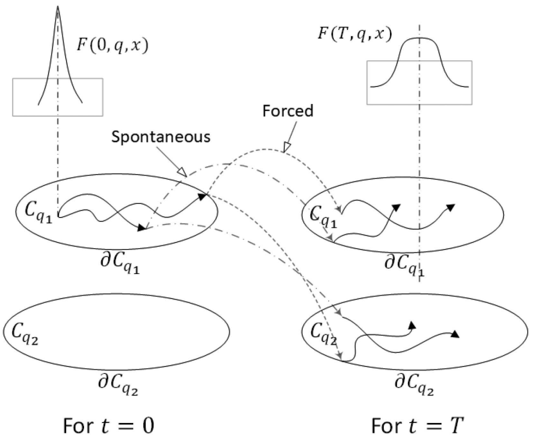


Fig. 1. A continuous-time stochastic hybrid system with two discrete states at time 0 and T.

$$\begin{aligned}\frac{\partial F(t,q,x)}{\partial t} &= L(F(t,q,x)) = \underbrace{-\sum_{a=1}^{d} \frac{\partial}{\partial x_a}(f_a(q,x)F(t,q,x))}_{\text{drift}} \\ &\underbrace{+\sum_{a=1}^{d}\sum_{b=1}^{d}\frac{\partial^2}{\partial x_a \partial x_b}\sum_{c=1}^{d}\frac{g_{ac}(q,x)g_{cb}(q,x)F(t,q,x)}{2}}_{\text{diffusion}} \\ &\underbrace{-r(q,x)F(t,q,x)}_{\text{jump out}} \underbrace{+\sum_{q\in Q}\int_{x\in \mathrm{A}_q} h(q,x,q',x')r(q',x')F(t,q',x')\mathrm{d}x',}_{\text{spontaneous jump in}} \\ &\underbrace{+\sum_{q\in Q}\int_{x\in \partial\mathrm{A}_q} h(q,x,q',x')(\mathbf{n}\cdot\mathbf{F})\mathrm{d}x',}_{\text{forced jump in}}\end{aligned} \tag{3}$$

where f_a is the ath element of f from (1), $\mathbf{n}$ is the unit vector pointing out of the flow set and the inner product $\mathbf{n}\cdot\mathbf{F}$ is the corresponding outgoing flow. Here, $\mathbf{F}$ is a matrix (more precisely a second-order tensor) whose components are given by

$$\mathbf{F}_{ab} = \frac{\partial}{\partial x_b}\sum_{c=1}^{d}\frac{g_{ac}(q,x)g_{cb}(q,x)F(t,q,x)}{2}, \tag{4}$$

where $a, b \in [d]$. In (3), L is the Fokker–Planck operator for the system, and we write symbolically that $F(t,q,x) = e^{tL}F(0,q,x)$. On the boundary, we have

$$F(t,q',x') = 0, \tag{5}$$

as it is absorbing (paths jump away immediately after hitting the boundary). In the rest of the article, we assume that the stochastic hybrid system given in this section is well defined in the sense that it gives a Fokker–Planck equation with a unique solution [40, 54].

2.1.3 *Invariant Distribution.* An invariant distribution of the continuous-time stochastic hybrid system $F_{\text{inv}}(q,x)$ is defined by

$$L(F_{\text{inv}}(q,x)) = 0. \tag{6}$$

In this work, when handling temporal logic specifications of an infinite time horizon, we assume that $F(t,q,x)$ converges to the invariant distribution function $F_{\text{inv}}(q,x)$ to ensure that the truth value of the specifications will not change after a finite time.

2.1.4 System Observables. The state of the system is only partially observable. Here, we are interested in observables of the system given by

$$y(t) = \mathbb{E}[y(q(t),x(t))] = \sum_{q\in Q}\int_{\mathrm{A}_q} \gamma(q,x)F(t,q,x)\mathrm{d}x, \tag{7}$$

where $\gamma(q,x)$ is a weight function on $\mathbb{X}$, which is integrable in x for each $q \in Q$.

Example 2.1. Throughout the article, we use the following example to illustrate the theorems. Consider a continuous-time stochastic hybrid system with two discrete states on $\mathbb{X} = \{1\}\times[0,1]\cup\{2\}\times[2,4]$. It jumps uniformly to $[2,4]$ when hitting $x = 0$ or $x = 1$, and jumps uniformly to $[0,1]$ when hitting $x = 2$ or $x = 4$. It can jump spontaneously at any $x \in \mathbb{X}$ with the rate $h(x) = \mathbf{I}_{\mathbb{X}}(x)/3$, where $\mathbf{I}_{\mathbb{X}}(\cdot)$ is the indicator function of the set $\mathbb{X}$. In each location, the state of the system is governed by the stochastic differential equation

$$\mathrm{d}\mathbf{x} = \mathrm{d}t + \mathrm{d}B_t,$$

The probability distribution $F(t,q,x)$ of the state evolves by the Fokker–Planck equation

$$\frac{\partial F(t,q,x)}{\partial t} = -\frac{\partial F(t,q,x)}{\partial x} + \frac{1}{2}\frac{\partial^2 F(t,q,x)}{\partial x^2} + \frac{\partial F(t,q,0)}{\partial x} - \frac{\partial F(t,q,1)}{\partial x} + \frac{1}{2}\frac{\partial F(t,q,2)}{\partial x} - \frac{1}{2}\frac{\partial F(t,q,4)}{\partial x}$$

with the boundary conditions

$$F(t,q,0) = F(t,q,1) = F(t,q,2) = F(t,q,4) = 0.$$

Initially, the state of the system is uniformly distributed on $[0,1/2]$.

2.2 Inequality Metric Interval Temporal Logic

We are interested in verifying specifications in iMITL of the continuous-time stochastic hybrid systems. In iMITL, the atomic propositions are inequalities of the form $y \sim c$ ($c \in \mathbb{Q}$, $\sim\in \{<,\leq,\geq,>\}$), where y is an observable of the system given by (7); and these atomic propositions are concatenated by the syntax of MITL [9].

Definition 2.2 (iMITL Syntax). An iMITL formula is defined using the following BNF form:

$$\varphi ::= \bot \mid \top \mid y\sim c \mid \varphi\wedge\varphi \mid \varphi\vee\varphi \mid \varphi\mathcal{U}_I\varphi \mid \varphi\mathcal{R}_I\varphi,$$

where $c \in \mathbb{Q}$, $\sim\in \{<,\leq,\geq,>\}$ and I is a non-singleton interval on $\mathbb{R}_{\geq 0}$.

We note that the syntax does not contain negation ($\neg$), since $\{<,\leq,\geq,>\}$ is closed under negation. For a standard iMITL formula, negation on non-atomic formulas can always be pushed inside as part of the atomic propositions. For example, $\neg(y > 0)$ is defined as $y \leq 0$, $\neg(\varphi_1 \vee \varphi_2)$ is defined as $(\neg\varphi_1)\wedge(\neg\varphi_2)$, and $\neg(\varphi\mathcal{U}_I\psi)$ is defined as $(\neg\varphi)\mathcal{R}_I(\neg\psi)$.

The continuous-time stochastic hybrid system induces a signal $f(t) : \mathbb{R}_{\geq 0} \to 2^{\mathsf{AP}}$ by $(y \sim c) \in f(t)$ iff $y \sim c$ holds at time t. The semantics of iMITL are defined with respect to the signal $f(t)$ as follows.

Definition 2.3 (iMITL Semantics). Let φ be an iMITL formula and f be a signal $f : \mathbb{R}_{\geq 0} \to 2^{\mathsf{AP}}$. The satisfaction relation $\models$ between f and φ is defined according to the following inductive rules:

$f \models \bot$ iff false
$f \models \top$ iff true
$f \models y \sim c$ iff $(y \sim c) \in f^0$
$f \models \varphi \wedge \psi$ iff $(f \models \varphi) \wedge (f \models \psi)$
$f \models \varphi \vee \psi$ iff $(f \models \varphi) \vee (f \models \psi)$
$f \models \varphi \mathcal{U}_I \psi$ iff $\exists t \in I, (f^t \models \psi) \wedge \forall t' \in (0, t), f^{t'} \models \varphi$
$f \models \varphi \mathcal{R}_I \psi$ iff $\forall t \in I, (f^t \models \psi)$ or $\exists t \in \mathbb{R}_{>0}, (f^t \models \varphi \wedge \forall t' \in [0, t] \cap I, f^{t'} \models \psi)$ or
$\exists t \in I', t' \in I \cap (t, \infty), \forall t'' \in I, (t'' \leq t \to f^{t''} \models \psi) \wedge (t < t'' \leq t' \to f^{t''} \models \varphi)$,

where $f^r(\cdot) = f(r + \cdot)$ and $I' = I \cup \{\underline{I}\}$ in the semantics of $\varphi \mathcal{R}_I \psi$ with $\underline{I}$ being the lower bound of I. We define $[\![\varphi]\!]$ to be the set of signals that satisfy φ.

Our semantics of $\mathcal{R}$ in iMITL is different from standard MITL [9]. This is because it has recently been shown that the common semantics of MITL cannot ensure that the formulas $\neg(\varphi \mathcal{U}_I \psi)$ and $(\neg\varphi)\mathcal{R}_I(\neg\psi)$ are equivalent for the continuous-time domain (see [57] for details). The satisfiability/model checking problems for iMITL with abstract atomic propositions are known to be EXPSPACE-complete [9, 57]. The corresponding decision procedure has a close connection with timed automata.

Definition 2.4 (Timed Automata [8]). Timed automaton A is a tuple $(\mathtt{Q}, \mathtt{X}, \Sigma, \mathtt{L}, \mathtt{I}, \mathtt{E}, \mathtt{Q}^{\mathtt{init}}, \mathtt{Q}^{\mathtt{final}})$ where

- $\mathtt{Q}$ is a finite non-empty set of *locations.*
- $\mathtt{X}$ is a finite set of *clocks.*
- Σ is a finite *alphabet.*
- $\mathtt{L} : \mathtt{Q} \to \Sigma$ maps each location to the *label* of that location.
- $\mathtt{I} : \mathtt{Q} \to (\mathtt{X} \to \mathbb{I}_{\geq 0})$ maps each location to its *invariant* which is the set of possible values of variables in that location, where $\mathbb{I}_{\geq 0}$ is the set of intervals on $\mathbb{R}_{\geq 0}$.
- $\mathtt{E} \subseteq \mathtt{Q} \times \mathtt{Q} \times 2^{\mathtt{X}}$ is a finite set of *edges* of the form $e = (s, d, j)$, where $s = \mathtt{S}e$ is *source* of the edge; $d = \mathtt{D}e$ is *destination* of the edge; and $j = \mathtt{J}e$ is the set of clocks that are *reset* by the edge.
- $\mathtt{Q}^{\mathtt{init}} \subseteq \mathtt{Q}$ is the set of *initial locations.*
- $\mathtt{Q}^{\mathtt{final}} \subseteq \mathtt{Q}$ is the set of *final locations.*

A *run* of the timed automaton A is a sequence of tuples $(\rho, \tau, \zeta) \in \mathtt{Q}^\omega \times \mathbb{I}^\omega_{\geq 0} \times \mathtt{E}^\omega$ in which the following conditions holds: (i) $\rho_0 \in \mathtt{Q}^{\mathtt{init}}$, *i.e.*, ρ starts from an initial location $\mathtt{Q}^{\mathtt{init}}$; (ii) $(\mathtt{S}\zeta_n = \rho_n) \wedge (\mathtt{D}\zeta_n = \rho_{n+1})$, *i.e.*, the source and destination of edge ζ_n is ρ_n and ρ_{n+1}, respectively; (iii) $\tau_0, \tau_1, \ldots$ is an ordered and disjoint partition of the time horizon $\mathbb{R}_{\geq 0}$; and (iv) $\forall t \in \tau_n, x \in \mathtt{X}$, we have $\varrho_n(x) + t - \underline{\tau}_n \in \mathtt{I}(\rho_n, x)$, where $\varrho_0(x) = 0$ and $\varrho_{n+1}(x)$ is inductively defined by

$$\varrho_{n+1}(x) = \begin{cases} 0, & \text{if } x \in \mathtt{J}\zeta_n \\ \varrho_n(x) + \overline{\tau}_n - \underline{\tau}_n, & \text{otherwise} \end{cases}$$

i.e., clocks must satisfy the invariant of the current location. Here, $\underline{\tau}$ and $\overline{\tau}$ are the lower and upper bound of the interval.

A run satisfying the condition $\mathtt{inf}(\rho) \cap \mathtt{Q}^{\mathtt{final}} \neq \emptyset$, *i.e.*, some location from $\mathtt{Q}^{\mathtt{final}}$ has been visited infinitely many times by ρ, is called an *accepting run* of A. Note that every run of A induces a function f of type $\mathbb{R}_{\geq 0} \to \Sigma$ that maps t to $\mathtt{L}(\rho_n)$, where n is uniquely determined by the condition $t \in \tau_n$. We define the *language* of A, denoted by $\mathtt{Lang}(A)$, to be the set of all functions that are induced by accepting runs of A. The language of timed automata is closely related to MITL as follows.

Lemma 2.5 (MITL to Timed Automata [9]). *For any MITL formula φ, a timed automaton A_φ can be constructed such that* $\mathsf{Lang}(A_\varphi) = [\![\varphi]\!]$, i.e., *the set of functions that satisfy φ is exactly those that are induced by accepting runs of A_φ.*

Example 2.6. Following Example 2.1, we want to check the following iMITL formula

$$\varphi_1 = \top\, \mathcal{U}_{[0,\infty]} \left(y_2(t) > \frac{1}{4}\right), \quad \varphi_2 = \left(y_1(t) > \frac{1}{2}\right) \mathcal{U}_{[0,\infty]} \left(y_2(t) > \frac{1}{4}\right),$$

where

$$y_1(t) = \sum_{q \in Q} \int_{\mathbf{A}_q} I_{[0,1]} F(t,q,x)\mathrm{d}x, \quad y_2(t) = \sum_{q \in Q} \int_{\mathbf{A}_q} I_{[2,4]} F(t,q,x)\mathrm{d}x.$$

3 MODEL REDUCTION OF CONTINUOUS-TIME HYBRID SYSTEMS

The model reduction procedure for a continuous-time stochastic hybrid system follows the three steps: (i) reduce the dynamics by partitioning the state space; (ii) reduce the temporal logic specifications accordingly; and (iii) estimate the model reduction error.

3.1 Reducing the Dynamics

The Mori–Zwanzig model reduction method reduces a continuous-time stochastic hybrid system to a CTMC.

Definition 3.1. A CTMC is a tuple $(\mathbb{S}, p_0, A, \mathsf{S}^{\mathrm{init}}, \mathsf{F})$ where $\mathbb{S}$ is a finite set of *states*, $p_0 \in \mathbb{S}$ is an *initial distribution* on $\mathbb{S}$, and A is a *transition rate matrix* with $A_{ii} = -\sum_{j \neq i} A_{ij}$. At the state $s_i \in \mathbb{S}$, the probability of jumping to any other state s_j follows an (independent) exponential distribution with the rate A_{ij}.

To implement the Mori–Zwanzig model reduction method [15] for continuous-time stochastic hybrid systems, we partition the continuous state space into finitely many partitions $\mathbb{S} = \{s_1, \ldots, s_n\}$, and treat each of them as a discrete state. The idea of partitioning is similar to [2, 3] for the discrete-time stochastic hybrid systems. The partition is called an equipartition if they are hypercubes with the same size η. We assume that for each s_i, there exists $q \in Q$ such that $s_i \subseteq \{q\} \times \mathbf{A}_q$, and denote its measure by $\mu(s_i)$. Let $m(\mathbb{X})$ and $m(\mathbb{S})$ be sets of probability distribution functions on $\mathbb{X}$ and $\mathbb{S}$, respectively. Then we can define a projection $P : m(\mathbb{X}) \to m(\mathbb{S})$ and an injection $R : m(\mathbb{S}) \to m(\mathbb{X})$ between $m(\mathbb{X})$ and $m(\mathbb{S})$ by

$$p_j = (PF(q,x))_j = \int_{s_j} F(q,x)\mathrm{d}x, \tag{8}$$

where p_j is the jth element of p, and

$$Rp = \sum_{j=1}^{n} p_j \mathbf{U}_{s_j}, \tag{9}$$

where $\mathbf{U}_{s_j}$ is the uniform distribution on s_j:

$$\mathbf{U}_{s_j}(x) = \begin{cases} \frac{1}{\mu(s_j)}, & \text{if } x \in s_j \\ 0, & \text{otherwise.} \end{cases} \tag{10}$$

Here the projection P and the injection R are defined for probability distributions. But they extend naturally to L_1 functions on $\mathbb{X}$ and $\mathbb{S}$, respectively. The projection P is the left inverse of the injection R but not *vice versa*, namely $PR = I$ but $RP \neq I$.

This projection P and injection R can reduce the Fokker–Planck operator to a transition rate matrix on $\mathbb{S}$, and hence reduce the continuous-time stochastic hybrid system into a CTMC. Following [15], the Fokker–Planck operator given in (3) reduces to the transition rate matrix A by

$$A = PLR. \tag{11}$$

In practice, we are usually interested in a continuous state space $\mathbb{X}$ that is partitioned into hypercubes of edge length η. In this case, the transition rate matrix A is explicitly expressed as follows.

Theorem 3.2. *Let $\mathbb{S} = \{s_1, s_2, \dots, s_n\}$ be a partition*[1] *of the d-dimensional continuous state space $\mathbb{X}$ into hypercubes of edge length η, and P and R be the corresponding projection and injection given by* (8)–(10)*, the transition rate from the state s_i to the state s_j ($i \neq j$) at time t is given by*

$$A_{ij} = \mathbf{n} \cdot \left(\mathbf{N} + \frac{\mathbf{M} \cdot \mathbf{n}(p_i - p_j)}{\eta} \right) + \mathbf{R}, \tag{12}$$

for $a, b \in [n]$, where $\mathbf{n}$ is (if exists) the unit vector of the boundary $s_i \cap s_j$ pointing from s_i to s_j, $\mathbf{N}$ is a d dimensional vector with components

$$\mathbf{N}_a = \int_{\partial s_i \cap \partial s_j} f_a(q, x) \mathrm{d}x, \tag{13}$$

$\mathbf{M}$ is a $d \times d$ matrix with components

$$\mathbf{M}_{ab} = \int_{\partial s_i \cap \partial s_j} \sum_{c=1}^{d} \frac{g_{ac}(q, x) g_{cb}(q, x)}{2} \mathrm{d}x, \tag{14}$$

and for an inner cell s_i,

$$\mathbf{O} = \int_{s_i \times s_j} \frac{\mathbf{I}_{s_j}(q, x) h(q, x, q', x') r(q', x') \mathbf{I}_{s_i}(q', x')}{\eta^d} \mathrm{d}x' \mathrm{d}x, \tag{15}$$

for a boundary cell s_j,

$$\mathbf{O} = \int_{s_i \times s_j} \frac{\mathbf{I}_{s_j}(q, x) h(q, x, q', x') \mathbf{n}' \cdot \mathbf{M} \cdot \mathbf{n}' p_i}{\eta/2;} \mathrm{d}x' \mathrm{d}x, \tag{16}$$

with $\mathbf{I}_{s_i}$ being the indicator function of s_i and $\mathbf{n}'$ being the vector pointing out of the boundary of the flow set.

Proof. For simplicity, we first show the proof for the 1D case. Specifically, for fixed q, we integrate both sides of (3) on the cell $I = [p, p + \Delta p]$, and apply the Stokes theorem for the first two terms, we derive

$$\begin{aligned} \int_I \frac{\partial F(t, q, x)}{\partial t} \mathrm{d}x = & -f(q, x) F(t, q, x) \Big|_p^{p+\Delta_p} + \frac{\partial}{\partial x} \frac{g^2(q, x) F(t, q, x)}{2} \Big|_p^{p+\Delta_p} - \int_I r(q, x) F(t, q, x) \mathrm{d}x \\ & + \sum_{q \in Q} \int_{x \in \mathbb{A}_{||}} \int_I h(q, x, q', x') r(q', x') F(t, q', x') \mathrm{d}x \mathrm{d}x' + \sum_{q \in Q} \int_{x \in \partial \mathbb{A}_{||}} \int_I h(q, x, q', x') (\mathbf{n} \cdot \mathbf{F}) \mathrm{d}x \mathrm{d}x'. \end{aligned} \tag{17}$$

The left-hand side of (17) is the rate of probability change in the cell I. On the right-hand side of (17), (i) the combination of the first two terms $f(q, x) F(t, q, x) - \frac{\partial}{\partial x} \frac{g^2(q, x) F(t, q, x)}{2}$ is the probability flow

[1] The partitions can be labeled by S arbitrarily.

on the boundary; (ii) the other terms correspond to average probability jumps inside the cell I. The same is true for multidimensional cases.

By applying (11), it is easy to check the probability flow between adjacent cells sharing a boundary is (14) and (13). The probability of jumping from one inner cell to another cell has the rate (15). Finally, the probability of jumping from one boundary cell to another cell has the rate (16). Thus, (12) holds. □

Roughly speaking, the transition rate between two partitions in the same location is the flux of $f(q,x)$ across the boundary and the transition rate between two different locations is the flux of $r(q,x)$.

3.2 Reducing iMITL Formulas

The observables on the continuous-time stochastic hybrid system reduce to the corresponding CTMC using the projection P. Let y be an observable on the continuous-time stochastic hybrid system with weight function $\gamma(q,x)$. To facilitate further discussion, we assume that $\gamma(q,x)$ is invariant under the projection P,

$$\gamma(q,x) = RP\gamma(q,x), \tag{18}$$

which means that the function $\gamma(q,x)$ can be written as a linear combination of indicator functions of the partitions of P (sometimes called a *simple function*). This assumption can be lifted by approximating a general function $\gamma(q,x)$ with a simple function and considering the approximation error. As we refine the partition, the approximation error converges to 0.

We define a corresponding observable y' on the CTMC that derives from the model reduction procedure by

$$y'(0) = \sum_{q\in Q}\int_{\mathrm{A}_q}\gamma(q,x)PF(0,q,x)\mathrm{d}x = \sum_{i=1}^{n}\left(\int_{s_i}\gamma(q,x)\mathrm{d}x\right)\left(\int_{s_i}F(0,q,x)\mathrm{d}x\right) = \sum_{i=1}^{n}r_i p(i). \tag{19}$$

From now on, we will always denote the corresponding observable on the CTMC by y' for any observable y on the continuous-time stochastic hybrid system.

3.3 Reduction Error Estimation

For a given observable y with weight function $\gamma(q,x)$, the error of the projection P with respect to the observable y is defined by the maximal possible difference between y and y',

$$\Delta_y = \left|\sum_{q\in Q}\int_{\mathrm{A}_q}\gamma(q,x)(F(0,q,x) - RPF(0,q,x))\mathrm{d}x\right|. \tag{20}$$

Remark 1. When refining the partition of $\mathbb{X}$, $RP \to I$; that is, any distribution function $F(q,x)$ on the state space, $|\sum_{q\in Q}\int_{\mathrm{A}_q}\gamma(q,x)(F(q,x) - RPF(q,x))| \to 0$ holds for any measurable weight function $\gamma(q,x)$. Accordingly for (20), $\Delta_y \to 0$ for any given y.

By the definition of Δ_y, we know that, at the initial time, the atomic propositions on the continuous-time stochastic hybrid system and the CTMC have the relations

$$y(0) > c \Longrightarrow y'(0) > c - \Delta_y, \quad y(0) < c \Longrightarrow y'(0) < c + \Delta_y,$$

and similarly,

$$y'(0) > c + \Delta_y \Longrightarrow y(0) > c, \quad y'(0) < c - \Delta_y \Longrightarrow y(0) < c.$$

$$\begin{tikzcd}
F(0,q,x) \arrow[r, "e^{Lt}"'] & F(t,q,x) \arrow[d, dashed, "P"] \\
p(0) \arrow[u, dashed, "R"] \arrow[r, "e^{At}"'] & p(t)
\end{tikzcd}$$

Fig. 2. Diagram for reduction error. It is non-commutative due to the errors in the projection P.

To derive the relations of the observables between the continuous-time stochastic hybrid system and the CTMC at any time, we define the reduction error of the observable y at time t due to the model reduction process by

$$\Theta_y(t) = |y(t) - y'(t)| = \left| \sum_{q \in Q} \int_{\mathrm{A}_q} \gamma(q,x)(e^{Lt} - Re^{At}P)F(0,q,x)\mathrm{d}x \right|, \tag{21}$$

where $F(0,q,x)$ is an initial distribution of the continuous-time stochastic hybrid system and $y'(t)$ is the corresponding observable of $y(t)$ on the CTMC. This reduction error is illustrated in Figure 2. Note that the diagram is not commutative; the difference between going along the two paths is related to the reduction error.

For a finite time horizon T, the supremum $\sup_{t \le T} \Theta(t)$ provides a uniform bound of the reduction error. For an infinite time horizon $T \to \infty$, the supremum may go unbounded. Below, we provide a sufficient condition for boundedness. We define the reduction error of the Fokker–Planck operator L by

$$\delta(t,q,x) = (L - RPL)e^{tRPL}F(0,q,x). \tag{22}$$

Accordingly, we define the integration of $\delta(t,q,x)$ with respect to the weight function $\gamma(q,x)$ by

$$\Lambda_y = \sup_{t \ge 0} \left| \sum_{q \in Q} \int_{\mathrm{A}_q} \gamma(q,x)(L - RPL)e^{tRPL}F(0,q,x)\mathrm{d}x \right|, \tag{23}$$

which captures the maximal change of the time derivative of observable y. When the reduction error $\delta(t,q,x)$ converges exponentially in time, an upper bound of the reduction error $\Theta(t)$ can be obtained.

Definition 3.3. For $\alpha > 0$, $\beta \ge 1$ and a given observable y, the continuous-time stochastic hybrid system is α-contractive with respect to y, if for any initial distribution function $F(0,q,x)$ on the state space, we have

$$\left| \sum_{q \in Q} \int_{\mathrm{A}_q} \gamma(q,x)e^{tL}\delta(t,q,x)\mathrm{d}x \right| \le \beta e^{-\alpha t} \left| \sum_{q \in Q} \int_{\mathrm{A}_q} \gamma(q,x)\delta(t,q,x)\mathrm{d}x \right|. \tag{24}$$

where $\delta(t,q,x)$ is given by (22).

This contractivity condition is to ensure that the model reduction error is bounded for all time, which is required for approximately keeping the truth value of temporal logic specifications of an infinite time horizon. Although the condition seems restrictive, it is valid for a relatively wide range of systems including asymptotically stable systems. It is a commonly-used sufficient condition to guarantee the existence and uniqueness of an invariant measure for general dynamical systems, and the contractivity factor α is usually derived case-by-case. Using Definition 3.3, we obtain the following theorem.

Theorem 3.4. *If the continuous-time stochastic hybrid system from Section 2.1.1 is α-contractive, then for any $t \geq 0$, the reduction error $\Theta_y(t)$ for an observable y satisfies*

$$\Theta_y(t) \leq \frac{\beta\Lambda_y}{\alpha} + \Delta_y. \tag{25}$$

Proof. The discrepancy in evolving the system by the original dynamics L and the reduced dynamics RPL can be captured by Dyson's formula [15]

$$e^{tL} = e^{tRPL} + \int_{[0,t]} e^{(t-\tau)L}(L - RPL)e^{\tau RPL}\mathrm{d}\tau. \tag{26}$$

This formula can be verified by taking time derivatives on both sides. Substituting (26) into (21) gives

$$\begin{aligned}\Theta_y(t) \leq & \left|\sum_{q\in Q}\int_{\mathrm{A}_q} \gamma(q,x)(e^{tRPL} - Re^{tA}P)F(0,q,x)\mathrm{d}x\right| \\ & + \left|\sum_{q\in Q}\int_{\mathbb{R}^d\times[0,t]} \gamma(q,x)e^{(t-\tau)L}(L - RPL)e^{\tau RPL}F(0,q,x)\mathrm{d}\tau\mathrm{d}x\right|.\end{aligned} \tag{27}$$

Since the projection P and the injection R preserve the L_1 norm, RPL is also a Fokker–Planck operator. Noting $Re^{tA}PF(0,q,x) = e^{tRPL}PF(0,q,x)$, by (20), we see that the first term on the right-hand side of (27) is less than Δ_y.

For the second term on the right-hand side of (27), by (23)–(24), we have

$$\begin{aligned}\Theta_y(t) &\leq \Delta_y + \left|\sum_{q\in Q}\int_{\mathrm{A}_q}\int_{[0,t]} \gamma(q,x)e^{(t-\tau)L}\delta(\tau,q,x)\mathrm{d}\tau\mathrm{d}x\right| \\ &\leq \Delta_y + \left|\sum_{q\in Q}\int_{\mathrm{A}_q}\int_{[0,t]} \beta e^{-\alpha(t-\tau)}\gamma(q,x)\delta(\tau,q,x)\mathrm{d}\tau\mathrm{d}x\right| \leq \frac{\beta\Lambda_y}{\alpha} + \Delta_y.\end{aligned} \tag{28}$$

□

Theorem 3.4 implies the following relations between the atomic propositions on the continuous-time stochastic hybrid system and the CTMC.

Theorem 3.5. *If the continuous-time stochastic hybrid system given in Section 2.1.1 is α-contractive, then we have*

$$y(t) > c \Longrightarrow y'(t) > c - \left(\frac{\beta\Lambda_y}{\alpha} + \Delta_y\right), \tag{29}$$

$$y(t) < c \Longrightarrow y'(t) < c + \left(\frac{\beta\Lambda_y}{\alpha} + \Delta_y\right), \tag{30}$$

and similarly,

$$y'(t) > c + \left(\frac{\beta\Lambda_y}{\alpha} + \Delta_y\right) \Longrightarrow y(t) > c, \tag{31}$$

$$y'(t) < c - \left(\frac{\beta\Lambda_y}{\alpha} + \Delta_y\right) \Longrightarrow y(t) < c. \tag{32}$$

In Theorem 3.5, the term Δ_y bounds the initial model reduction error and the term $\frac{\beta\Lambda_y}{\alpha}$ bounds the model reduction error accumulated over time. Following Theorem 3.5, to verify an iMITL formula φ for an α-contractive continuous-time stochastic hybrid system introduced in Section 2.1.1, we can strengthen φ to ψ by replacing the atomic propositions according to (31)–(32). If ψ holds for the CTMC derived from the continuous-time stochastic hybrid system following the model reduction procedure of Sections 3.1 and 3.2, then φ holds for the continuous-time stochastic hybrid system.

The main complexity of the Mori–Zwanzig reduction method is in performing integration of the system dynamic function on the partitions. This process can be significantly simplified if the integration has a closed-form solution, e.g., linear, polynomial, exponential, and so on. For more general dynamics, we may use Monte-Carlo integrations with considerations on extra statistical errors.

Example 3.6. Following Example 2.1 and 2.6, the invariant distribution of this process is $F_{\text{inv}} = \mathbf{U}_{\mathbb{X}}/3$. We partition $\mathbb{X}$ into intervals of length $1/N$. By the above model reduction procedure it reduces to a CTMC with transition rate matrix M given by

$$M_{ij} = \frac{\delta_{ij}}{4} + \frac{1}{4N}$$

where $i \in [3N]$ and $j \in [3N]$. The invariant distribution F_{inv} remains unchanged, and the iMITL formula to check is

$$\varphi_1' = \top\, \mathcal{U}_{[0,\infty]} \left(y_2'(t) > \frac{1}{4} + \Theta_y(t) \right)$$
$$\varphi_2' = \left(y_1'(t) > \frac{1}{2} + \Theta_y(t) \right) \mathcal{U}_{[0,\infty]} \left(y_2'(t) > \frac{1}{4} + \Theta_y(t) \right),$$

where $\Theta_y(t)$ is the model reduction error and

$$y_1'(t) = \sum_{i=1}^{N} p_i(t), \quad y_2'(t) = \sum_{i=2N+1}^{3N} p_i(t).$$

When $N = 30$, we have $\Theta_y(t) \leq 0.02$ from (8) and (23).

4 STATISTICAL MODEL CHECKING OF IMITL

We now propose a statistical model checking method to verify the reduced iMITL formula φ on the CTMC C derived from the model reduction (Section 3). We denote the set of atomic propositions in φ by AP_φ. The pair C, φ can generate a *signal* by evaluating the truth value of the atomic propositions in AP_φ on C for each time; the singleton set containing this signal is denoted by $[\![C, \mathsf{AP}_\varphi]\!]$. Let $T_{C,\varphi}$ be the timed automaton such that $[\![C, \mathsf{AP}_\varphi]\!] \subseteq \mathsf{Lang}(T_{C,\varphi})$. Using Lemma 2.5, we construct two timed automata T_φ and $T_{\neg\varphi}$ such that their languages are signals accepted and rejected by φ, respectively. If the intersection of $\mathsf{Lang}(T_{C,\varphi})$ and $\mathsf{Lang}(T_\varphi)$ is empty then C violates φ. Similarly, if the intersection of $\mathsf{Lang}(T_{C,\varphi})$ and $\mathsf{Lang}(T_{\neg\varphi})$ is empty then C satisfies φ. This emptiness problem for the intersection of timed automata is known to be PSPACE-complete [8]. However, none of the two intersections may be empty. To avoid this situation, we assume that each signal of $\mathsf{Lang}(T_{C,\varphi})$ remains close to the signal in $[\![C, \mathsf{AP}_\varphi]\!]$. That is, if the signal in $[\![C, \mathsf{AP}_\varphi]\!]$ satisfies/violates φ, then there is a close signal that violates/satisfies φ.

We use a statistical method to construct the timed automaton $T_{C,\varphi}$. Let $p(t)$ be the state distribution of the CTMC C at the time t. For each atomic proposition $(y \sim c)$ from AP_φ, where y is of the form $r \cdot p(t)$, we assume *wlog.* that

ALGORITHM 1: Truncating Time Horizon

```
Data: CTMC $(C, p_0)$, estimation of invariant distribution $p^*$, Atomic formula $(y \sim c)$, parameters $\alpha'$, and $\delta'$
Function DurationOfSimulation
    $t \leftarrow 1$
    while Close$\left(p(t), p^*, \frac{1}{2}\alpha', \frac{\delta'}{3}\right)$ = failed do
        $t \leftarrow 2 \times t$
        $\alpha' \leftarrow \frac{1}{2}\alpha'$
    end
    return $t$+1
```

— r is not identical to $\mathbf{0}$ (otherwise, $(y \sim c)$ can be replaced with $\top$ or $\bot$); and

— the maximum absolute value in r is exactly 1 (by scaling the parameters in $(y \sim c)$).

Since the CTMC converges to a unique invariant distribution p^{inv}, there exists a known constant $\delta' \in \mathbb{R}$ and a known estimation p^* of p^{inv} such that

— $\forall (r \cdot p(t) \sim c) \in \mathsf{AP}_\varphi, |r \cdot p^{\text{inv}} - c| > \delta'$, and

— $\|p^{\text{inv}} - p^*\|_1 < \frac{\delta'}{3}$, where $\|\cdot\|_1$ is the ℓ_1 norm.

Furthermore, let T be a time such that $\|p(T) - p^*\|_1 < \frac{\delta'}{3}$ holds (we will show how to find T later in this section). For any $t \geq T$, we have $\|p(t) - p^{\text{inv}}\|_1 < \frac{2\delta'}{3}$. Also, we assume that $r \cdot p^{\text{inv}} - c > \delta'$ holds (the discussion for $r \cdot p^{\text{inv}} - c < -\delta'$ is similar). By $|r \cdot p(t) - r \cdot p^{\text{inv}}| \leq \|p(t) - p^{\text{inv}}\|_1 < \frac{2\delta'}{3}$, we know $r \cdot p(t) - c > \frac{\delta'}{3}$. Then by $|r \cdot p(t) - r \cdot p^*| \leq \|p(t) - p^*\|_1 < \frac{\delta'}{3}$, we have $r \cdot p^* > c$. Therefore, the truth value of $(y \sim c)$ is fixed for any $t > T$ and can be determined by looking at p^*.

We use Algorithm 1 to find a time T such that $p(T)$ is $\frac{\delta'}{3}$-close to p^* (the estimation of the invariant distribution). Our statistical algorithm compares $p(T)$ and p^* for successively larger values of T until $\|p(T) - p^*\|_1 < \frac{\delta'}{3}$ holds. To check if two distributions are close, we employ Lemma 4.1. When $\|p(t) - p^*\|_1 > \frac{\delta'}{3}$, starting from the iteration $i = 1$, the probability of Lemma 4.1 not rejecting t is at most $\alpha' \times 2^{-i}$. Thus, the probability of returning a wrong time T is at most α'.

Lemma 4.1 ([11]). *For any $\alpha, \delta > 0$, and any two distributions p and p' on n discrete values, there is a test* Close(p, p', α, δ) *which runs in time $O(n^{2/3}\delta^{-8/3}\log(n/\alpha))$ such that (i) if $\|p - p'\|_1 \leq \max(\frac{\delta^{4/3}}{32\sqrt[3]{n}}, \frac{\alpha}{4\sqrt{n}})$, then the test accepts with probability at least $1 - \alpha$; and (ii) if $\|p - p'\|_1 > \delta$, then the test rejects with probability at least $1 - \alpha$.*

Before constructing the timed automaton for times within $[0, T]$, we first explain how to statistically verify if $p(t)$ satisfies an atomic proposition $(y \sim c)$. For now, assume that elements of r are from $\{0, 1\}$. Then, $p(t)$ satisfies $(y \sim c)$ iff the probability of drawing a state s from $p(t)$ with $r(s) = 1$ is great than c. This can be statistically checked by drawing samples from $p(t)$ and using the **sequential probability ratio test (SPRT)** [59, 72, 77]. It requires as input an indifference parameter $\delta \in (0, 1)$, and the error bounds $\alpha, \gamma \in (0, 1)$. The output of this test, called $\mathcal{A}_0$, is yes, no, or unknown with the following guarantees:

$$\mathbb{P}[\mathsf{res} = \mathsf{no} \quad | \; r \cdot p(t) > c \quad] \leq \alpha, \tag{33a}$$

$$\mathbb{P}[\mathsf{res} = \mathsf{yes} \quad | \; r \cdot p(t) \not> c \quad] \leq \alpha, \tag{33b}$$

$$\mathbb{P}[\mathsf{res} = \mathsf{unknown} \; | \; |r \cdot p(t) - c| > \delta] \leq \gamma. \tag{33c}$$

The parameters α, γ, δ can be made arbitrarily small at the cost of requiring more samples. For the general case that the elements of r are real numbers, the SPRT is not applicable. Instead, we can use a method from Chow and Robbins [16] to estimate unknown distributions on finite states from finite samples with bounded error as follows.

Given that T is known, we construct the timed automaton for the time interval $[0, T]$. For simplicity, we focus on constructing $T_{C,\{AP\}}$ for an atomic proposition $AP : y = \sum_{i=1}^{n} r_i p_i > c$, denoted by the pair (r, c). Then, at every time t, $f(t)$ is either the emptyset or $\{(r, c)\}$. Let $T_{C,\{AP\}}(t)$ be the set of reachable locations of $T_{C,\{AP\}}$ at time t. Given the parameters $\delta > 0$, let $\Delta > 0$ be a value at most $\frac{\delta}{3} \max\{|\frac{\mathrm{d}}{dt}(r \cdot p)(t)| \mid t \in [0, T]\}^{-1}$ (Δ can be set to $\frac{\delta}{3}\|r\|_\infty \|M\|_1$, where $\|\cdot\|_\infty$ and $\|\cdot\|_1$ are, respectively, ℓ_∞ and ℓ_1 induced norms). For any $t \in [0, T]$ and $t' \in [t - \Delta, t + \Delta] \cap [0, T]$, we have

(1) if $r \cdot p(t) - c > \frac{\delta}{3}$ then $r \cdot p(t') > c$,
(2) if $r \cdot p(t) - c < -\frac{\delta}{3}$ then $r \cdot p(t') < c$,
(3) if $|r \cdot p(t) - c| \leq \frac{2\delta}{3}$ then $|r \cdot p(t') - c| \leq \delta$.

We partition $[0, T)$ into $\left\lfloor \frac{T}{2\Delta} \right\rfloor + 1$ intervals, each of size strictly less than 2Δ. Let $[t_1, t_2)$ be one of these intervals and define $t = \frac{1}{2}(t_1 + t_2)$. We then run $\mathcal{A}_0$ twice as follows, where α' and γ' are obtained by dividing input parameters α and γ over $\lceil \frac{T}{2\Delta} \rceil$.

$$\mathsf{res}_1 = \mathcal{A}_0\left(r \cdot p(t), c + \frac{\delta}{3}, \frac{1}{|\mathsf{AP}_\varphi|}\alpha', \frac{1}{|\mathsf{AP}_\varphi|}\gamma', \frac{\delta}{3}\right),$$
$$\mathsf{res}_2 = \mathcal{A}_0\left(r \cdot p(t), c - \frac{\delta}{3}, \frac{1}{|\mathsf{AP}_\varphi|}\alpha', \frac{1}{|\mathsf{AP}_\varphi|}\gamma', \frac{\delta}{3}\right),$$

If $\mathsf{res}_1 = \mathsf{yes}$, then $\forall t' \in [t_1, t_2), (r \cdot p(t') > c)$ holds with a bounded error α', so we set $T_{C,\{AP\}}(t) = \{AP\}$. If $\mathsf{res}_2 = \mathsf{no}$, then $\forall t' \in [t_1, t_2), (r \cdot p(t') < c)$ holds with a bounded error α', so we set $T_{C,\{AP\}}(t) = \{\emptyset\}$. Otherwise, for any time t' in the interval, $|r \cdot p(t') - c| \leq \delta$ with a bounded error γ', so we set

— $T_{C,\{AP\}}(t) = \{q, q'\}$,
— $\mathsf{L}(q) = \{AP\}$ and $\mathsf{L}(q') = \emptyset$,
—entry to q or q', and
— switches between q and q' when their common invariant permits.

This ensures that within $[t_1, t_2)$, both states q, q' can be reached and they can switch arbitrary many times. Intuitively, this means the atomic propositions within this interval are unknown and not fixed. The algorithm to construct $T_{C,\varphi}$ is given by Algorithm 2.

Based on Algorithms 1 and 2, the complete algorithm $\mathcal{A}$ to statistically verify the iMITL formula φ for the CTMC C with the parameters $\delta, \delta', \alpha, \gamma$. The parameters δ' and $\frac{1}{2}\min\{\alpha, \gamma\}$ are given to Algorithm 1, and the parameters δ, $\frac{1}{2}\alpha$, $\frac{1}{2}\gamma$ are given to Algorithm 2. We have the following guarantee on the return res of the complete algorithm $\mathcal{A}$:

$$\mathbb{P}[\mathsf{res} = \mathsf{no} \mid C \models \varphi] \leq \alpha, \tag{34}$$
$$\mathbb{P}[\mathsf{res} = \mathsf{yes} \mid C \not\models \varphi] \leq \alpha. \tag{35}$$

As for the unknown output, let $B^\delta(r \cdot p)$ be the tube of functions that are point-wise δ-close to $r \cdot p$ (formally, a function $f : \mathbb{R}_{\geq 0} \to \mathbb{R}$ is in $B^\delta(r \cdot p)$ iff for any $t \in \mathbb{R}_{\geq 0}$, $|f(t) - r \cdot p(t)| \leq \delta$). The algorithm guarantees that

$$\left(\forall \sigma \in B^\delta(r \cdot p),\ \sigma \models \varphi\right) \implies \mathbb{P}[\mathsf{res{=}unknown}] \leq \alpha + \gamma, \tag{36}$$
$$\left(\forall \sigma \in B^\delta(r \cdot p),\ \sigma \not\models \varphi\right) \implies \mathbb{P}[\mathsf{res{=}unknown}] \leq \alpha + \gamma. \tag{37}$$

ALGORITHM 2: Constructing the timed automaton $T_{C,\varphi}$

```
$h \leftarrow \max\{|\frac{d}{dt}(r \cdot p)(t)| \mid t \in [0,T]\}$
$\Delta \leftarrow \frac{\delta}{3h}$
$n \leftarrow |\mathsf{AP}_\varphi|\lceil\frac{T}{2\Delta}\rceil$
$T_{C,\{AP\}}$ ← an empty automaton
$\mathsf{X} \leftarrow \{t\}$, $q_{\text{last}} \leftarrow \bot$
forall the $i \leftarrow 0$ to $\lfloor\frac{T}{2\Delta}\rfloor$ do
    $\mathtt{res}_1 \leftarrow \mathcal{A}_0(r \cdot p((i+\frac{1}{2})2\Delta), c+\frac{\delta}{3}, \frac{\alpha}{2n}, \frac{\beta}{2n}, \frac{\delta}{3})$
    $\mathtt{res}_2 \leftarrow \mathcal{A}_0(r \cdot p((i+\frac{1}{2})2\Delta), c-\frac{\delta}{3}, \frac{\alpha}{2n}, \frac{\beta}{2n}, \frac{\delta}{3})$
    add a new location $q$ to $\mathsf{Q}$
    if $\mathtt{res}_1 = \mathsf{yes}$ then
        $\mathsf{L}(q) \leftarrow \{AP\}$
    else if $\mathtt{res}_2 = \mathsf{no}$ then
        $\mathsf{L}(q) \leftarrow \emptyset$
    else
        $\mathsf{L}(q) \leftarrow \mathsf{unknown}$
    $\mathtt{I}(q) \leftarrow 2i\Delta \le t < 2(i+1)\Delta$
    if $q_{\text{last}} \neq \bot$ then
        $\mathsf{E} \leftarrow \mathsf{E} \cup \{(q_{\text{last}}, q, \emptyset)\}$
    else
        $\mathsf{Q}^{\text{init}} \leftarrow \{q\}$
    $q_{\text{last}} = q$
end
add a new location $q$ to $\mathsf{Q}$
$\mathtt{I}(q) \leftarrow \mathtt{true}$, $\mathsf{Q}^{\mathtt{final}} \leftarrow \{q\}$
$\mathsf{E} \leftarrow \mathsf{E} \cup \{(q_{\text{last}}, q, \emptyset), (q, q, \emptyset)\}$
if $r \cdot p^{\text{inv}} > c$ then
    $\mathsf{L}(q) \leftarrow \{AP\}$
else
    $\mathsf{L}(q) \leftarrow \emptyset$
$T_{C,\{AP\}}$ ← replace any unknown location in $\mathsf{Q}$ with $q$ and $q'$ labeled $\{AP\}$ and $\emptyset$. Duplicate edges from/to $q$ and $q'$ accordingly
Add $(q, q', \emptyset)$ and $(q', q, \emptyset)$ to $\mathsf{E}$ for every split locations in the previous step.
return $T_{C,\{AP\}}$
```

Intuitively, if all the functions that are close to $r \cdot p$ satisfy φ or none of them does then the probability of returning unknown is at most $\alpha + \gamma$.[2]

Example 4.2. Following Example 3.6, we run our algorithm on the CTMC and derive that both φ_1' and φ_2' are true. This implies that the formulas φ_1 and φ_2 given in Example 2.6 are true on the system given in Example 2.1.

5 DISCRETE HYBRID SYSTEMS

In this section, we study the verification of temporal properties for discrete-time stochastic hybrid systems. We follow the formulation of the discrete-time stochastic hybrid systems from [2, 3] and use the iLTL [43] to capture the temporal properties of interest. The iLTL specifications are verified

[2] There is a slight abuse of notation in (36) and (37). They use a function of type $\mathbb{R}_{\geq 0} \to \mathbb{R}$. However, $\models$ requires a signal (function of type $\mathbb{R}_{\geq 0} \to 2^{\mathsf{AP}}$). The signal contains atomic proposition $y \sim c$ at time t iff $y(t) \sim c$ holds.

on the discrete-time stochastic hybrid systems by model reduction and statistical model checking in a similar way as Sections 3 and 4.

Discrete-time stochastic hybrid systems. Following the formulation of [3], we focus on a Fokker–Planck formulation and interpretation of the model. Using the notations from Section 2.1.1, the dynamics of the system is captured by the initial distribution $F(0, q, x)$ on the state space $\mathbb{X} \subseteq Q \times \mathbb{R}^d$ and the transition function $T(q', x', q, x)$, which satisfies

$$\sum_{q \in Q} \int_{\mathrm{A}_q} T(q', x', q, x) \mathrm{d}x' = 1, \tag{38}$$

for any $(q, x) \in \mathbb{X}$. The transition function $T(q', x', q, x)$ can be derived from the dynamics of the continuous-time stochastic hybrid systems given in Section 2.1.1 by time discretization [2, 3]. The observable y of the system is defined in the same way as in the continuous-time case.

We call the transition function $T(q', x', q, x)$ α-contractive, if for any two distributions $F(q, x)$ and $G(q, x)$, it holds that

$$\left\| \sum_{q \in Q} \int_{\mathrm{A}_q} T(q', x', q, x) \left(F(q, x) - G(q, x) \right) \mathrm{d}x \right\| \leq \alpha \| F(q, x) - G(q, x) \|, \tag{39}$$

where $\| \cdot \|$ is the L_1-norm. This α-contractive condition is different from its continuous-time counterpart (Definition 3.3) in two aspects. First, the parameter α of (39) is the contractive factor for one discrete time step, while the parameter α of (24) is the contractive rate for the continuous time. Second, the contractivity of (24) is defined with respect to the given observable, while the contractivity of (39) is independent of the observables. For the discrete time, the contractivity of (39) generally holds for many common stochastic dynamics, such as (discrete-time) diffusion processes.

Inequality linear temporal logic (iLTL). We use the iLTL [43] to capture the temporal properties of interest for the discrete-time stochastic hybrid systems. The iLTL can be viewed as the discrete-time version of the iMITL introduced in Section 2.2. It is a variation of the common LTL [28] by setting the atomic propositions AP to be inequalities of the form $y \sim c$, where $c \in \mathbb{Q}$, $\sim \in \{<, \leq, \geq, >\}$, and y is an observable of the system given by (7). (This is similar to the case of iMITL in Definition 2.2.) Again in the syntax of iLTL, we drop the negation operator $\neg$ by pushing it inside and using completeness of $\{<, \leq, \geq, >\}$.

Definition 5.1 (iLTL Syntax). The syntax of iLTL formulas is defined using the BNF rule:

$$\varphi = \bot \mid \top \mid y \sim c \mid \varphi \wedge \varphi \mid \varphi \vee \varphi \mid \mathcal{X} \varphi \mid \varphi \mathcal{U} \varphi \mid \varphi \mathcal{R} \varphi,$$

where $c \in \mathbb{Q}$ and $\sim \in \{<, \leq, \geq, >\}$.

The discrete-time stochastic hybrid system induces a signal $f : \mathbb{N} \to 2^{\mathsf{AP}}$ by $(y \sim c) \in f(t)$ iff $y \sim c$ holds at time t. According, we define the semantics of iLTL on the system by Definition 5.2.

Definition 5.2 (iLTL Semantics). Let φ be an iLTL formula and f be a discrete-time signal. The satisfaction relation $\models$ between f and φ is inductively defined according to the rules:

$$\begin{array}{lll} f \models \bot & \text{iff} & \text{false} \\ f \models \top & \text{iff} & \text{true} \\ f \models y \sim c & \text{iff} & (y \sim c) \in f(0) \\ f \models \varphi \vee \psi & \text{iff} & (f \models \varphi) \vee (f \models \psi) \\ f \models \varphi \wedge \psi & \text{iff} & (f \models \varphi) \wedge (f \models \psi) \\ f \models \mathcal{X}\varphi & \text{iff} & f^1 \models \varphi \\ f \models \varphi \mathcal{U} \psi & \text{iff} & \exists i \in \mathbb{N}, (f^i \models \psi \wedge \forall j \in [i], f^j \models \varphi) \\ f \models \varphi \mathcal{R} \psi & \text{iff} & \forall i \in \mathbb{N}, f^i \models \psi \text{ or } \exists i \in \mathbb{N}, (f^i \models \varphi \wedge \forall j \in [i], f^j \models \psi), \end{array}$$

where $f^i(\cdot) = f(\cdot + i)$. Let $[\![\varphi]\!]$ be the set of signals that satisfy φ.

Verifying the signals can be done by transforming them to Büchi automata [28], which is the discrete-time version of timed automata in Definition 2.4.

Definition 5.3. A Büchi automaton B is a tuple $(\mathsf{S}, \Sigma, \Gamma, \mathsf{S}^{\mathsf{init}}, \mathsf{F})$ where S is a finite non-empty set of *states*, Σ is a finite *alphabet*, $\Gamma \subseteq \mathsf{S} \times \Sigma \times \mathsf{S}$ is a *transition relation*, $\mathsf{S}^{\mathsf{init}} \subseteq \mathsf{S}$ is a set of *initial* states, $\mathsf{F} \subseteq \mathsf{S}$ is a set of *final* states. We write $s_1 \xrightarrow{a} s_2$ instead of $(s_1, a, s_2) \in \Gamma$.

The Büchi automaton B takes an infinite sequence $w \in \Sigma^\omega$ as an input and accepts it, iff there exists an infinite sequence of states $\rho \in \mathsf{S}^\omega$ such that (1) $\rho_0 \in \mathsf{S}^{\mathsf{init}}$, (2) $\forall n \in \mathbb{N}, \rho_n \xrightarrow{w_n} \rho_{n+1}$, and (3) $\mathtt{inf}(\rho) \cap \mathsf{F} \neq \emptyset$, where $\mathtt{inf}(\rho)$ is the set of states that appear infinitely often in $\{\rho_n\}_{n=1}^{\infty}$. An infinite sequence of states is called a *run* of B if it satisfies 1 and 2, and an *accepting run* if it satisfies 1, 2, and 3. We define the *language* of B, denoted by $\mathsf{Lang}(B)$, to be the set of all infinite sequences in Σ^ω that are accepted by B.

Similar to the relation between MITL and timed automata (Lemma 2.5), we introduce the following result on the conversion between LTL and Büchi automata.

Lemma 5.4 (LTL to Büchi Automata [24, 25, 28]). *For any LTL formula φ, a Büchi automaton B_φ can be constructed such that* $\mathsf{Lang}(B_\varphi) = [\![\varphi]\!]$, i.e., *the set of infinite words that satisfy φ is exactly those that are accepted by B_φ.*

5.1 Model Reduction

The model reduction for the discrete-time stochastic hybrid systems is similar to that for the continuous-time ones discussed in Sections 3.1 to 3.3, following the three steps of (i) reducing the dynamics by partitioning the state space, (ii) reducing the temporal logic specifications accordingly, and (iii) estimating the model reduction error.

5.1.1 Reducing the Dynamics. For a discrete-time stochastic hybrid system, we can reduce it to a finite-state DTMC by the set-oriented method [21] which can be viewed as a discrete-time variation of the Mori–Zwanzig method [15]. Similar to Section 3, let $S = \{s_1, s_2, \ldots, s_n\}$ be a partition of the continuous state space $\mathbb{X}$, and P, R be the corresponding projection and injection operators as given by (8)–(10). As shown in Figure 3 and Theorem 5.5, they induce a projection from the Markov kernel $T : m(\mathbb{X}) \to m(\mathbb{X})$ to a Markov kernel $T_r : m(S) \to m(S)$ by

$$T_r = PTR. \tag{40}$$

For multiple steps, the diagram for projection is shown by the non-commutative diagram in Figure 4.

$$\begin{tikzcd}
F(t,q,x) \arrow[r, "T"'] & F(t+1,q,x) \arrow[d, dashed, "P"] \\
p(t) \arrow[u, dashed, "R"] \arrow[r, "T_r"'] & p(t+1)
\end{tikzcd}$$

Fig. 3. Diagram for single-step reduction.

$$\begin{tikzcd}
F(0,q,x) \arrow[r, "T"'] & F(1,q,x) \arrow[r, dotted] & F(t-1,q,x) \arrow[r, "T"'] & F(t,q,x) \arrow[d, "P"] \\
p(0) \arrow[u, "R"] \arrow[r, dashed, "T_r"'] & p(1) \arrow[r, dotted] & p(t-1) \arrow[r, dashed, "T_r"'] & p(t)
\end{tikzcd}$$

Fig. 4. Diagram for multiple-step reduction.

Theorem 5.5. *Let $S = \{s_1, \ldots, s_n\}$ be a measurable partition of the state space $\mathbb{X}$. Then the discrete-time stochastic hybrid system reduces to a DTMC (T_r, p_0) by*

$$p_0(i) = \int_{s_i} F(0,q,x)\mathrm{d}x, \quad T_r(i,j) = \int_{s_i}\int_{s_j} T(q',x',q,x)\mathrm{d}x'\mathrm{d}x.$$

5.1.2 Reduced iLTL. Similar to Section 3.2, an observable $y(t)$ from (7) can be reduced approximately to an observable $y'(t)$ on the DTMC by (19). Initially, the discrepancy between $y(0)$ and $y'(0)$ and is given by (5.6).

Lemma 5.6. *For any $F(q,x) \in m(\mathbb{X})$ and projection operator P, we have*

$$\begin{aligned}
y(0) > b + \delta_P\|F\|_\infty \implies y'(0) > b, \quad & y'(0) > b + \delta_P\|F\|_\infty \implies y(0) > b,\\
y(0) < b - \delta_P\|F\|_\infty \implies y'(0) < b, \quad & y'(0) < b - \delta_P\|F\|_\infty \implies y(0) < b,
\end{aligned}$$

where

$$\delta_P = \|F(0,q,x) - RPF(0,q,x)\|_{TV}, \tag{41}$$

is the error of projection operator P in total variance, where $\|\cdot\|_{TV}$ is the total variation distance.

5.1.3 Reduction Error Estimation. To compute the discrepancy between $y(t)$ and $y'(t)$ for any $t \in \mathbb{N}$, we first note that the projection operator P is contractive.

Lemma 5.7. *Let $\mathbb{S} = \{s_1, \ldots, s_n\}$ be a measurable partition of $\mathbb{X}$ and P be the projection operator associated with $\mathbb{S}$. For any $F(q,x), F'(q,x) \in m(\mathbb{X})$,*

$$\|PF(q,x) - PF'(q,x)\|_{TV} \leq \|F(q,x) - F'(q,x)\|_{TV}.$$

As shown in the non-commutative diagram in Figure 4, the discrepancy for any $t \in \mathbb{N}$ can be written as

$$\Delta_t = \|PT^{(t)}F(0,q,x) - T_r^{(t)}PF(0,q,x)\|_{\mathrm{TV}} = \|PT^{(t)}F(0,q,x) - P(TRP)^{(t)}F(0,q,x)\|_{\mathrm{TV}}.$$

So, its error bound can be derived as follows.

Theorem 5.8. *Given a discrete-time stochastic hybrid system and a projection operator P, the t-step ($t \geq 1$) error of projection*

$$\Delta_t \leq \sum_{i=0}^{t-1} \delta_P((TRP)^{(i)} F(0, q, x)), \tag{42}$$

where δ_P is given in (41).

Proof. For $t = 1$, we have,

$$\begin{aligned}\Delta_1 &= \|PTF(0, q, x) - P(TRP)F(0, q, x)\|_{\mathrm{TV}} \leq \|TF(0, q, x) - TRPF(0, q, x)\|_{\mathrm{TV}} \\ &\leq \|F(0, q, x) - RPF(0, q, x)\|_{\mathrm{TV}} = \delta_P(F(0, q, x)).\end{aligned}$$

For $t > 1$, with F denoting $F(0, q, x)$, we have

$$\begin{aligned}\Delta_t &= \|PT^{(t)}F - P(TRP)^{(t)}F\|_{\mathrm{TV}} \leq \|T^{(t)}F - (TRP)^{(t)}F\|_{\mathrm{TV}} \leq \|T^{(t)}F - T^{(t-1)}(TRP)F\|_{\mathrm{TV}} \\ &+ \|T^{(t-1)}(TRP)F - T^{(t-2)}(TRP)^{(2)}F\|_{\mathrm{TV}} \ldots + \|T(TRP)^{(t-1)}F - (TRP)^{(t)}F\|_{\mathrm{TV}} \leq \sum_{i=0}^{t-1} \delta_P((TRP)^{(i)}F).\end{aligned}$$

□

For a finite time horizon T, the supremum $\sup_{t \leq T} \Delta(t)$ provides a uniform bound of the reduction error. For an infinite time horizon $T \to \infty$, when T is strictly contractive (39), we can derive a uniform error bound for Δ_t as we did for the continuous-time case.

Theorem 5.9. *Given a discrete-time stochastic hybrid system, a projection operator P and the corresponding injection R, if the Markov kernel T is strictly contractive by factor $\alpha \in (0, 1)$, then the t-step ($t \geq 1$) error of projection*

$$\Delta_t \leq \frac{\delta_P}{1 - \alpha}, \tag{43}$$

where

$$\delta_P = \sup_{i \in \mathbb{N}} \delta_P((TRP)^{(i)} F(0, q, x)). \tag{44}$$

Proof. For $t = 1$, clearly $\Delta_t = \delta_P$. For $t \geq 2$, by (5.1.3) and with F denoting $F(0, q, x)$, we have

$$\begin{aligned}\Delta_t &\leq \|T^{(t)}F - T^{(t-1)}(TRP)F\|_{\mathrm{TV}} + \|T^{(t-1)}(TRP)F - T^{(t-2)}(TRP)^{(2)}F\|_{\mathrm{TV}} \\ &+ \ldots + \|T(TRP)^{(t-1)}F - (TRP)^{(t)}F\|_{\mathrm{TV}} \leq (1 + \alpha + \ldots + \alpha^t)\delta_P \leq \frac{\delta_P}{1 - \alpha}.\end{aligned} \tag{45}$$

□

By combining Lemma 5.6 and Theorem 5.9, we can derive the following theorem on the relationship between linear inequalities on the original Markov process and linear inequalities on the reduced Markov process.

Theorem 5.10. *Given a measurable partition $\mathbb{S} = \{s_1, \ldots, s_n\}$ and the corresponding projection operator P, a discrete-time stochastic hybrid system and its reduction (T_r, p_0) satisfies the equations:*

$$y(t) > b + \frac{\delta_P \|F\|_\infty}{1 - \alpha} \Longrightarrow y'(t) > b, \quad y'(t) > b + \frac{\delta_P \|F\|_\infty}{1 - \alpha} \Longrightarrow y(t) > b, \tag{46}$$

$$y(t) < b - \frac{\delta_P \|F\|_\infty}{1 - \alpha} \Longrightarrow y'(t) < b, \quad y'(t) < b - \frac{\delta_P \|F\|_\infty}{1 - \alpha} \Longrightarrow y(t) < b, \tag{47}$$

for any $t \geq 0$, where δ_p is given by (44), respectively.

Theorem 5.10 can be viewed as the discrete-time counterpart of Theorem 3.5. In Theorem 3.5, the model reduction error is bounded by two term: one for the initial error, and the other for the error accumulated over time. In Theorem 5.10, these two terms are combined into one, due to the difference between the contractivity condition (39) and (24).

Following Theorem 5.10, to verify an iLTL formula φ for an α-contractive discrete-time stochastic hybrid system introduced in Section 2.1.1, we can strengthen φ to ψ by replacing the atomic propositions according to Theorem 5.10. If ψ holds for the DTMC derived from the discrete-time stochastic hybrid system following the aforementioned model reduction procedure, then φ holds for the discrete-time stochastic hybrid system.

5.2 Statistical Model Checking of iLTL

Similar to Section 4, we introduce a statistical model checking procedure for iLTL specifications on the reduced systems. Again, we denote the atomic proposition $p = \sum_{i=1}^{n} r_i p_i = r \cdot p > c$ by a pair (r, c). For an iLTL formula φ and a DTMC generating a sequence of distributions $w = p_0 p_1 p_2 \ldots$, define $u = u_0 u_1 u_2 \ldots$ where $u_t = \{(r, c) \in \mathsf{AP}_\varphi \mid r \cdot p_t > c\}$ is the set of atomic propositions that are true at time t. Similar to Section 4, our algorithm in this section has four steps:

- Construct the Büchi automata B_φ and $B_{\neg\varphi}$.
- Find a time step T at which $p(T)$ is very close to our estimation of the invariant distribution.
- Construct $B_{M,\varphi}$,
- If $\mathsf{Lang}(B_{M,\varphi}) \cap [\![B_\varphi]\!] = \emptyset$ then return no, if $\mathsf{Lang}(B_{M,\varphi}) \cap [\![B_{\neg\varphi}]\!] = \emptyset$ then return yes, otherwise, return unknown.

These steps are similar to their corresponding step in Section 4. For example, the first step is carried out using Lemma 5.4. Simulation of discrete and continuous Markov chains are different procedures, but they both can be performed efficiently, and that is what we need for the second and third steps. Similarly, checking emptiness of intersection of timed automata and Büchi automata are different procedures, but they are both known to be decidable [61]. The main difference with Algorithm 2 is that since in Lemma 5.4 time is discrete, to find labels of $B_{M,\varphi}$, we only run one instance of $\mathcal{A}_0$ at each step. Algorithm 3 shows the pseudocode for different steps. Again, similar to Algorithm 2, unknown labels are modeled using two locations; one labeled by $\{(y \sim c)\}$ and the other labeled by $\emptyset$. However, since the time is discrete for Büchi automata, there will be no extra transition between these two locations.

Similar to our previous algorithm, in addition to a Markov chain M, iLTL formula φ, and p^*, an estimation of the invariant distribution p^{inv}, Algorithm 3 takes two error parameters $\alpha, \gamma \in (0, 1)$ and two indifference parameters $\delta, \delta' \in (0, 1)$. The parameters δ' and $\frac{1}{2}\min\{\alpha, \gamma\}$ are used to find the time bound T, and the parameters δ, $\frac{1}{2}\alpha$, and $\frac{1}{2}\gamma$ are used to construct labels of Büchi automaton $B_{M,\varphi}$ before reaching step T. We have the following guarantee about the algorithm:

$$\mathbb{P}[\mathsf{res} = \mathsf{no} \ \mid M \models \varphi] \leq \alpha, \quad \mathbb{P}[\mathsf{res} = \mathsf{yes} \mid M \not\models \varphi] \leq \alpha,$$
$$\left(\forall \sigma \in B^\delta(r \cdot p),\ \sigma \models \varphi\right) \implies \mathbb{P}[\mathsf{res{=}unknown}] \leq \alpha + \gamma,$$
$$\left(\forall \sigma \in B^\delta(r \cdot p),\ \sigma \not\models \varphi\right) \implies \mathbb{P}[\mathsf{res{=}unknown}] \leq \alpha + \gamma,$$

where $B^\delta(r \cdot p)$ is the tube of discrete functions that are δ-close to $r \cdot p$.

6 CASE STUDY

We provide an example to illustrate the application of our approach as well as its scalability. We consider a non-linear jump system with the continuous state $x(t) \in \mathbb{R}^n$ and the discrete state

ALGORITHM 3: Model Checking Markov Chains Against iLTL Formulas

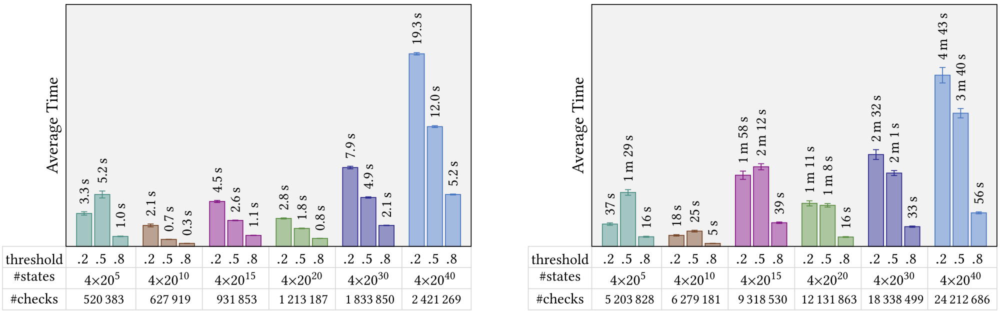
```
Data: Markov chain $(M, p_0)$, estimation of invariant distribution $p^*$, iLTL formula $\varphi$, parameters $\alpha, \gamma, \delta, \delta'$
Result: yes, no, or unknown
Function NumberOfSamplingSteps()
    $t \leftarrow 1$
    $\alpha' \leftarrow \frac{1}{2}\min\{\alpha, \gamma\}$
    while Close$(p(t), p^*, \frac{1}{2}\alpha', \frac{\delta'}{3})$ = failed do
        $t \leftarrow 2 \times t$
        $\alpha' \leftarrow \frac{1}{2}\alpha'$
    end
    return $t$+1
Function LabelFiniteNumberOfSteps($m \in \mathbb{N}$)
    forall the $t \in \{0, 1, \ldots, m-1\}$, $(r, c) \in$ AP do
        $asg(t, (r, c)) \leftarrow \mathcal{A}_0(r \cdot p(t), c, \frac{\alpha}{2m|\mathsf{AP}|}, \frac{\gamma}{2m|\mathsf{AP}|}, \frac{\delta}{3})$
    end
    return $asg$
Function AddLabelsOfInvariantDistribution($m \in \mathbb{N}, asg \in \mathbb{N} \times \mathsf{AP} \rightarrow \{\mathsf{yes}, \mathsf{no}, \mathsf{unknown}\}$)
    forall the $t \in \{m, m+1, \ldots\}$, $(r, c) \in$ AP do
        if $r \cdot p^* > c$ then
            $asg(t, (r, c)) \leftarrow$ yes
        else
            $asg(t, (r, c)) \leftarrow$ no
        end
    end
    return $asg$
Function ModelCheck
    $T \leftarrow$ NumberOfSamplingSteps();
    $asg \leftarrow$ LabelFiniteNumberOfSteps($T$);
    $asg \leftarrow$ AddLabelsOfInvariantDistribution($T, asg$);
    $[\![asg]\!] \leftarrow$ the Büchi automaton that accepts exactly the set of infinite paths induced by $asg$
    if Lang$(B_\varphi) \cap$ Lang$([\![asg]\!]) = \emptyset$ then
        return no
    if Lang$(B_{\neg\varphi}) \cap$ Lang$([\![asg]\!]) = \emptyset$ then
        return yes
    return unknown
```

$q(t) \in [m]$ with $m \in \mathbb{N}$. The continuous dynamics is

$$\frac{\mathrm{d}x}{\mathrm{d}t} = (A_{q(t)} + c_{q(t)}\|x(t)\|_\infty)x(t), \tag{48}$$

where $A_i \in \mathbb{R}^{n \times n}$ is Hurwitz and $c_i > 0$ for $i \in [m]$. The discrete state jumps spontaneously with the rate λ_1 from j to $j-1$ for $j = 2, \ldots, m$ and with the rate λ_2 from j to $j+1$ for $j = 1, \ldots, m-1$. Initially, the continuous state is distributed uniformly on the hypercube $C = \{x(0) \in \mathbb{R}^n \mid \|x(t)\|_\infty \leq K\}$; and the discrete state $q(0)$ uniformly on $[m]$. The form of the dynamics (48) is chosen for a simplified demonstration to high-dimensional non-linear dynamics. Below, the dimension n can be as high as 40, while the well-known challenging Toyota powertrain model is only 10. The same procedure directly applies to dynamics with diffusion terms.

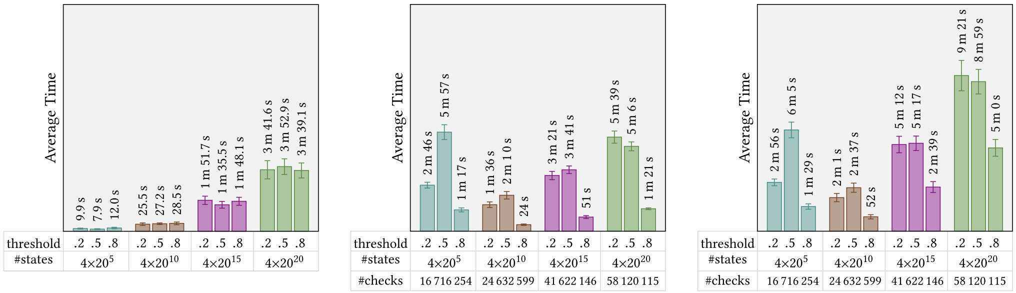
Assume that the elements of the dynamical matrices A_i are non-positive, then $x(t) \in C$ for all $t \in \mathbb{R}$. Therefore, we can partition the state space into $(2\eta)^n \times m$, each of length $1/\eta$. The hypercubes are indexed by $(i_1, \ldots, i_n, j)$ with $|i_k| \in \{-\eta, \ldots, -1, 1, \ldots, \eta\}$, $j \in [m]$, and $k \in [n]$.



(a) Time bound 1000

(b) Time bound 10000

Fig. 5. Bounded time.



(a) Time to find time-horizon

(b) Time to verify the property

(c) Total Time

Fig. 6. Unbounded time.

The transition probability rates are zero except

$$\lambda((i_1, \dots, i_n, j) \to (i_1, \dots, i_n, j-1)) = \lambda_1, \quad \lambda((i_1, \dots, i_n, j) \to (i_1, \dots, i_n, j+1)) = \lambda_2,$$

$$\lambda((i_1, \dots, i_k + 1, \dots, i_n, j) \to (i_1, \dots, i_k, \dots, i_n, j)) = c_j K \max_k \frac{|i_k|}{\eta^3} + \int_S \frac{(A_j x)_k}{\eta^2} \mathrm{d}x_1 \dots \mathrm{d}x_{k-1} \mathrm{d}x_{k+1} \dots \mathrm{d}x_n.$$

The desired property is

$$\top \mathcal{U}_{[0,T]}(w(F(t,q,x)) > p),$$

where T is a time bound (could be ∞), p is a probability threshold, and $w(\cdot)$ is the indicator function on a non-convex predicate stating exactly two elements of the continuous state are more than $\lceil K/2 \rceil$ away from the origin (formally, the predicate holds for a continuous state x iff $|\{i \in [n] \mid |x_i| \geq \lceil K/2 \rceil\}| = 2$). It asserts that before time T, a probability distribution will be reached such that the probability of a state x in that distribution satisfying the aforementioned predicate is larger than p.

We ran Algorithm 2 on multiple instances of this problem. In all of our experiments, $\lambda_1 = 0.03$, $\lambda_2 = 0.02$, $K = 1$, $\eta = 10$, and $\alpha = \beta = \delta_1 = 0.1$. We also fixed the number of discrete states (m) to be 4. The dimension of the continuous state is chosen from $\{5, 10, 15, 20, 30, 40\}$. These settings result in CTMCs with a large number of states: the smallest example has 1.28×10^7 states, and the largest example has more than 4.39×10^{52} states. In all the experiments, we set $c_1 = 0.1$, $c_2 = 0.2$, $c_3 = 0.3$, and $c_4 = 0.4$ in (48). Each instance of our simulation uses 4 Hurwitz matrices that are generated randomly beforehand. To simplify computaion, we generate the dynamic matrices A_i such that the contractive factor from (24) is < 0.99 and the reduction error from (25) is < 0.01

when each dimension is partitioned into 10 interval. The convergence to invariant distribution is also validated by numerical simulation. Finally, we used the maximum eigenvalue of the random matrices as the maximum rate of changes ($\max\{\dot{y}_i(t) \mid t \in [0, T]\}$) in our algorithm.

Our implementation is in Scala. We used the Apache Commons Mathematics Library [1] to find eigenvalues of a matrix. Our simulations are performed on Ubuntu 18.04 with i7-8700 CPU 3.2GHz and 16GB memory. We ran each test 50 times and report the average running time as well as the 95% confidence intervals. Figure 5 shows the results for the case that T is bounded ($1,000$ and $10,000$), and Figure 6 shows the results for the case that T is set to ∞. 'Threshold' is the value of p in our desired property. "#states" is the number of states in CTMC. "#checks" is the number of checkpoints the algorithm uses to discretize the time. This number does not tell how many steps the algorithm takes to simulate the system for T units of time (or until it reaches the invariant distribution). When the time is unbounded (i.e., $T = \infty$ in Figure 6), the algorithm first finds a time when the system sufficiently convergences to the invariant distribution. It is easy to see that in the invariant distribution, our example is reduced to a birth-death process, for which we can compute the invariant distribution analytically.

Figure 6(a) shows the average amount of time our algorithm spent to find a time in which the distribution is known to be invariant. Figure 6(b) shows the average amount of time the algorithm uses to verify the property after a time horizon is fixed (note that our property of interest does not hold at the invariant distribution). Figure 6(c) shows the sum of previous averages. As expected, the time consumption of our algorithm increases logarithmically with the number of states. This is because in statistical model checking, the number of required samples is independent of the number of the states, and the time to draw samples grows logarithmically with the number of the states.

7 CONCLUSION

In this work, we proposed a method of verifying temporal logic formulas on stochastic hybrid systems via model reduction in both continuous-time and discrete-time. Specifically, we reduce stochastic hybrid systems to Markov chains by partitioning the state space. We present an upper bound on the error introduced due to this reduction. In addition, we present stochastic algorithms that verify temporal logic formulas on Markov chains with arbitrarily high confidence.

REFERENCES

[1] Commons Math: The Apache Commons Mathematics Library. Retrieved June 10, 2019 from https://commons.apache.org/proper/commons-math.

[2] A. Abate, A. D'Innocenzo, and M. D. Di Benedetto. 2011. Approximate Abstractions of Stochastic Hybrid Systems. *IEEE Transactions on Automatic Control* 56, 11 (2011), 2688–2694.

[3] Alessandro Abate, Joost-Pieter Katoen, John Lygeros, and Maria Prandini. 2010. Approximate Model Checking of Stochastic Hybrid Systems. *European Journal of Control* 16, 6 (2010), 624–641.

[4] Alessandro Abate, Marta Kwiatkowska, Gethin Norman, and David Parker. 2014. Probabilistic model checking of labelled Markov processes via finite approximate bisimulations. In *Proceedings of the Horizons of the Mind. A Tribute to Prakash Panangaden*. Springer, 40–58.

[5] Alessandro Abate and Sadegh E. Z. Soudjani. 2015. Quantitative approximation of the probability distribution of a Markov process by formal abstractions. *Logical Methods in Computer Science* 11, 3 (2015), 1–29.

[6] Tuğkan Batu, Lance Fortnow, Ronitt Rubinfeld, Warren D. Smith, and Patrick White. 2013. Testing closeness of discrete distributions. *Journal of the ACM* 60, 1 (2013), 1–25.

[7] R. Alur, T. Dang, and F. Ivancic. 2003. Counter-Example Guided Predicate Abstraction of Hybrid Systems. In *Proceedings of the International Conference on Tools and Algorithms for the Construction and Analysis of Systems*. 208–223.

[8] Rajeev Alur and David L. Dill. 1994. A theory of timed automata. *Theoretical Computer Science* 126, 2 (April 1994), 183–235.

[9] Rajeev Alur, Tomás Feder, and Thomas A. Henzinger. 1996. The benefits of relaxing punctuality. *Journal of the ACM* 43, 1 (1996), 116–146.

[10] Christel Baier, Boudewijn Haverkort, Holger Hermanns, and Joost-Pieter Katoen. 2000. Model checking continuous-time Markov chains by transient analysis. In *Proceedings of the International Conference on Computer Aided Verification*. Springer, 358–372.

[11] Tuğkan Batu, Lance Fortnow, Ronitt Rubinfeld, Warren D. Smith, and Patrick White. 2013. Testing closeness of discrete distributions. *Journal of the ACM* 60, 1 (Feb. 2013), Article 4, 25 pages.

[12] C. L. Beck, S. Lall, T. Liang, and M. West. 2009. Model reduction, optimal prediction, and the Mori-Zwanzig representation of Markov chains. In *Proceedings of the 48th IEEE Conference on Decision and Control held jointly with 2009 28th Chinese Control Conference*. 3282–3287.

[13] Christos G. Cassandras and John Lygeros. 2018. *Stochastic Hybrid Systems*. CRC Press.

[14] R. Chadha and M. Viswanathan. 2010. A Counterexample Guided Abstraction-Refinement Framework for Markov Decision Processes. *ACM Transactions on Computational Logic* 12, 1 (2010), 1:1–1:49.

[15] Alexandre J. Chorin, Ole H. Hald, and Raz Kupferman. 2000. Optimal prediction and the Mori-Zwanzig representation of irreversible processes. *Proceedings of the National Academy of Sciences* 97, 7 (2000), 2968–2973.

[16] Y. S. Chow and Herbert Robbins. 1965. On the asymptotic theory of fixed-width sequential confidence intervals for the mean. *The Annals of Mathematical Statistics* 36, 2 (1965), 457–462.

[17] E. M. Clarke, A. Fehnker, Z. Han, B. Krogh, J. Ouaknine, O. Stursberg, and M. Theobald. 2003. Abstraction and Counterexample-Guided Refinement in Model Checking of Hybrid Systems. *International Journal of Foundations of Computer Science* 14, 4 (2003), 583–604.

[18] Edmund M. Clarke, Thomas A. Henzinger, Helmut Veith, and Roderick Bloem. 2018. *Handbook of Model Checking*. Springer.

[19] Dennis Dams and Orna Grumberg. 2018. Abstraction and abstraction refinement. In *Proceedings of the Handbook of Model Checking*. Springer, 385–419.

[20] Alexandre David, Dehui Du, Kim G. Larsen, Axel Legay, Marius Mikučionis, Danny Bøgsted Poulsen, and Sean Sedwards. 2012. Statistical model checking for stochastic hybrid systems. arXiv:1208.3856. Retrieved from https://arxiv.org/abs/1208.3856.

[21] M. Dellnitz and O. Junge. 1999. On the approximation of complicated dynamical behavior. *SIAM Journal on Numerical Analysis* 36, 2 (1999), 491–515.

[22] Josée Desharnais, Abbas Edalat, and Prakash Panangaden. 2002. Bisimulation for labelled Markov processes. *Information and Computation* 179, 2 (2002), 163–193.

[23] Geir E. Dullerud and Fernando Paganini. 2013. *A Course in Robust Control Theory: A Convex Approach*. Springer Science & Business Media.

[24] A. Duret-Lutz. 2011. LTL translation improvements in spot. In *Proceedings of the 5th International Conference on Verification and Evaluation of Computer and Communication Systems*. British Computer Society, Swinton, 72–83.

[25] Alexandre Duret-Lutz and Denis Poitrenaud. 2004. SPOT: An extensible model checking library using transition-based generalized büchi automata. In *Proceedings of the IEEE Computer Society's 12th Annual International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunications Systems*. IEEE Computer Society, 76–83.

[26] Sadegh Esmaeil, Zadeh Soudjani, and Alessandro Abate. 2013. Adaptive and sequential gridding procedures for the abstraction and verification of stochastic processes. *SIAM Journal on Applied Dynamical Systems* 12, 2 (2013), 921–956.

[27] Martin Fränzle, Ernst Moritz Hahn, Holger Hermanns, Nicolás Wolovick, and Lijun Zhang. 2011. Measurability and safety verification for stochastic hybrid systems. In *Proceedings of the 14th International Conference on Hybrid Systems: Computation and Control*. 43–52.

[28] Paul Gastin and Denis Oddoux. 2001. Fast LTL to BüChi automata translation. In *Proceedings of the 13th International Conference on Computer Aided Verification* . Springer-Verlag, London, 53–65.

[29] Antoine Girard, Giordano Pola, and Paulo Tabuada. 2010. Approximately bisimilar symbolic models for incrementally stable switched systems. *IEEE Transactions on Automatic Control* 55, 1 (2010), 116–126.

[30] Benjamin M. Gyori, Bing Liu, Soumya Paul, R. Ramanathan, and P.S. Thiagarajan. 2015. Approximate probabilistic verification of hybrid systems. In *Proceedings of the Hybrid Systems Biology*. Springer, 96–116.

[31] Sofie Haesaert, Sadegh Esmaeil Zadeh Soudjani, and Alessandro Abate. 2017. Verification of general Markov decision processes by approximate similarity relations and policy refinement. *SIAM Journal on Control and Optimization* 55, 4 (2017), 2333–2367.

[32] Floyd B. Hanson. 2007. *Applied Stochastic Processes and Control for Jump-Diffusions: Modeling, Analysis and Computation*. Society for Industrial and Applied Mathematics (2007), 29.

[33] Hans Hansson and Bengt Jonsson. 1994. A logic for reasoning about time and reliability. *Formal Aspects of Computing* 6, 5 (1994), 512–535.

[34] Joao Pedro Hespanha and Abhyudai Singh. 2005. Stochastic models for chemically reacting systems using polynomial stochastic hybrid systems. *International Journal of Robust and Nonlinear Control: IFAC-Affiliated Journal* 15, 15 (2005), 669–689.

[35] Jianghai Hu, John Lygeros, and Shankar Sastry. 2000. Towards a theory of stochastic hybrid systems. In *Proceedings of the International Workshop on Hybrid Systems: Computation and Control*. Springer, 160–173.

[36] Jianghai Hu, Wei-Chung Wu, and Shankar Sastry. 2004. Modeling subtilin production in bacillus subtilis using stochastic hybrid systems. In *Proceedings of the International Workshop on Hybrid Systems: Computation and Control*. Springer, 417–431.

[37] Sumit K. Jha, Edmund M. Clarke, Christopher J. Langmead, Axel Legay, André Platzer, and Paolo Zuliani. 2009. A bayesian approach to model checking biological systems. In *Proceedings of the International Conference on Computational Methods in Systems Biology*. Springer, 218–234.

[38] Xiaoqing Jin, Jyotirmoy V. Deshmukh, James Kapinski, Koichi Ueda, and Ken Butts. 2014. Benchmarks for model transformations and conformance checking. In *Proceedings of the 1st International Workshop on Applied Verification for Continuous and Hybrid Systems*.

[39] A. A. Julius and G. J. Pappas. 2009. Approximations of Stochastic Hybrid Systems. *IEEE Transactions on Automatic Control* 54, 6 (2009), 1193–1203.

[40] Ioannis Karatzas and Steven Shreve. 2012. *Brownian Motion and Stochastic Calculus*. Vol. 113. Springer Science & Business Media.

[41] M. Kloetzer and C. Belta. 2008. A fully automated framework for control of linear systems from temporal logic specifications. *IEEE Transactions on Automatic Control* 53, 1 (Feb. 2008), 287–297.

[42] Xenofon D. Koutsoukos and Derek Riley. 2008. Computational methods for verification of stochastic hybrid systems. *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans* 38, 2 (2008), 385–396.

[43] Youngmin Kwon and Gul Agha. 2004. Linear Inequality LTL (iLTL): A model checker for discrete time markov chains. In *Proceedings of the Formal Methods and Software Engineering*. Jim Davies, Wolfram Schulte, and Mike Barnett (Eds.). Lecture Notes in Computer Science, Vol. 3308. Springer, Berlin, 194–208.

[44] Youngmin Kwon and Gul Agha. 2011. Verifying the evolution of probability distributions governed by a DTMC. *IEEE Transactions on Software Engineering* 37, 1 (2011), 126–141.

[45] Abolfazl Lavaei, Sadegh Soudjani, Alessandro Abate, and Majid Zamani. 2021. Automated verification and synthesis of stochastic hybrid systems: A survey. https://arxiv.org/abs/2101.07491. Preprint.

[46] Edward Ashford Lee and Sanjit A. Seshia. 2017. *Introduction to Embedded Systems: A Cyber-Physical Systems Approach*. Mit Press.

[47] Axel Legay, Benoît Delahaye, and Saddek Bensalem. 2010. Statistical model checking: An overview. In *Proceedings of the International Conference on Runtime Verification*. Springer, 122–135.

[48] Axel Legay and Mahesh Viswanathan. 2015. *Statistical Model Checking: Challenges and Perspectives*. Springer

[49] Bing Liu, Andrei Hagiescu, Sucheendra K. Palaniappan, Bipasa Chattopadhyay, Zheng Cui, Weng-Fai Wong, and P. S. Thiagarajan. 2012. Approximate probabilistic analysis of biopathway dynamics. *Bioinformatics* 28, 11 (2012), 1508–1516.

[50] Bing Liu, David Hsu, and P. S. Thiagarajan. 2011. Probabilistic approximations of ODEs based bio-pathway dynamics. *Theoretical Computer Science* 412, 21 (2011), 2188–2206.

[51] Jun Liu, Necmiye Ozay, Ufuk Topcu, and Richard M. Murray. 2013. Synthesis of reactive switching protocols from temporal logic specifications. *IEEE Transactions on Automatic Control* 58, 7 (2013), 1771–1785.

[52] Amir Pnueli. 1977. The temporal logic of programs. In *Proceedings of the 18th Annual Symposium on Foundations of Computer Science*. IEEE, 46–57.

[53] Giordano Pola, Antoine Girard, and Paulo Tabuada. 2008. Approximately bisimilar symbolic models for nonlinear control systems. *Automatica* 44, 10 (2008), 2508–2516.

[54] Daniel Revuz and Marc Yor. 2013. *Continuous Martingales and Brownian Motion*. Vol. 293. Springer Science & Business Media.

[55] Christian Robert and George Casella. 2013. *Monte Carlo Statistical Methods*. Springer Science & Business Media.

[56] N. Roohi, P. Prabhakar, and M. Viswanathan. 2017. HARE: A Hybrid Abstraction Refinement Engine for verifying nonlinear hybrid automata. In *Proceedings of the International Conference on Tools and Algorithms for the Construction and Analysis of Systems*. 573–588. Springer, Berlin.

[57] Nima Roohi and Mahesh Viswanathan. 2018. Revisiting MITL to fix decision procedures. In *Proceedings of the International Conference on Verification, Model Checking, and Abstract Interpretation*. Springer, 474–494.

[58] Nima Roohi, Yu Wang, Matthew West, Geir E. Dullerud, and Mahesh Viswanathan. 2017. Statistical verification of the Toyota powertrain control verification benchmark. In *Proceedings of the 20th International Conference on Hybrid Systems: Computation and Control*. 65–70.

[59] Koushik Sen, Mahesh Viswanathan, and Gul Agha. 2005. On statistical model checking of stochastic systems. In *Proceedings of the Computer Aided Verification*. Kousha Etessami and Sriram K. Rajamani (Eds.). Number 3576 in Lecture Notes in Computer Science. Springer, Berlin, 266–280.

[60] Abhyudai Singh and Joao P. Hespanha. 2010. Stochastic hybrid systems for studying biochemical processes. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences* 368, 1930 (2010), 4995–5011.

[61] A. P. Sistla and E. M. Clarke. 1985. The complexity of propositional linear temporal logics. *Journal of the ACM* 32, 3 (1985), 733–749. DOI: https://doi.org/10.1145/3828.3837

[62] Sadegh Esmaeil, Zadeh Soudjani, Rupak Majumdar, and Tigran Nagapetyan. 2017. Multilevel monte carlo method for statistical model checking of hybrid systems. In *Proceedings of the International Conference on Quantitative Evaluation of Systems*. Springer, 351–367.

[63] Martin Střelec, Karel Macek, and Alessandro Abate. 2012. Modeling and simulation of a microgrid as a stochastic hybrid system. In *Proceedings of the 2012 3rd IEEE PES Innovative Smart Grid Technologies Europe*. IEEE, 1–9.

[64] Anantharaman Subbaraman and Andrew R. Teel. 2017. Robust global recurrence for a class of stochastic hybrid systems. *Nonlinear Analysis: Hybrid Systems* 25 (2017), 283–297.

[65] Sean Summers and John Lygeros. 2010. Verification of discrete time stochastic hybrid systems: A stochastic reach-avoid decision problem. *Automatica* 46, 12 (2010), 1951–1961.

[66] P. Tabuada and G. J. Pappas. 2006. Linear time logic control of discrete-time linear systems. *IEEE Transactions on Automatic Control* 51, 12 (2006), 1862–1877.

[67] Andrew R. Teel. 2017. *Recent Developments in Stability Theory for Stochastic Hybrid Inclusions*. Springer International Publishing, Cham, 329–354.

[68] Andrew R. Teel and Joao P. Hespanha. 2015. Stochastic hybrid systems: A modeling and stability theory tutorial. In *Proceedings of the 2015 IEEE 54th Annual Conference on Decision and Control*. IEEE, 3116–3136.

[69] Andrew R. Teel, Anantharaman Subbaraman, and Antonino Sferlazza. 2014. Stability analysis for stochastic hybrid systems: A survey. *Automatica* 50, 10 (2014), 2435–2456.

[70] Ilya Tkachev and Alessandro Abate. 2013. Formula-free finite abstractions for linear temporal verification of stochastic hybrid systems. In *Proceedings of the 16th International Conference on Hybrid Systems: Computation and Control*. ACM, 283–292.

[71] Ilya Tkachev, Alexandru Mereacre, Joost-Pieter Katoen, and Alessandro Abate. 2013. Quantitative automata-based controller synthesis for non-autonomous stochastic hybrid systems. In *Proceedings of the 16th International Conference on Hybrid Systems: Computation and Control*. ACM, 293–302.

[72] A. Wald. 1945. Sequential tests of statistical hypotheses. *The Annals of Mathematical Statistics* 16, 2 (1945), 117–186.

[73] Yu Wang, Nima Roohi, Matthew West, Mahesh Viswanathan, and Geir E. Dullerud. 2015. A Mori-Zwanzig and MITL based approach to statistical verification of continuous-time dynamical systems. *IFAC-PapersOnLine* 48, 27 (2015), 267–273.

[74] Yu Wang, Nima Roohi, Matthew West, Mahesh Viswanathan, and Geir E. Dullerud. 2015. Statistical verification of dynamical systems using set oriented methods. In *Proceedings of the 18th International Conference on Hybrid Systems: Computation and Control*. ACM, 169–178.

[75] Yu Wang, Nima Roohi, Matthew West, Mahesh Viswanathan, and Geir E. Dullerud. 2016. Verifying Continuous-time Stochastic Hybrid Systems via Mori-Zwanzig model reduction. In *Proceedings of the 2016 IEEE 55th Conference on Decision and Control*. IEEE, 3012–3017.

[76] Tichakorn Wongpiromsarn, Ufuk Topcu, and Richard M. Murray. 2010. Receding horizon control for temporal logic specifications. In *Proceedings of the 13th ACM International Conference on Hybrid Systems: Computation and Control* . ACM, New York, NY, 101–110.

[77] Håkan L. S. Younes. 2006. Error control for probabilistic model checking. In *Proceedings of the 7th International Conference on Verification, Model Checking, and Abstract Interpretation*. 142–156.

[78] Håkan L. S. Younes and Reid G. Simmons. 2006. Statistical probabilistic model checking with a focus on time-bounded properties. *Information and Computation* 204, 9 (2006), 1368–1409.

[79] Majid Zamani, Peyman Mohajerin Esfahani, Rupak Majumdar, Alessandro Abate, and John Lygeros. 2014. Symbolic control of stochastic systems via approximately bisimilar finite abstractions. *IEEE Transactions on Automatic Control* 59, 12 (2014), 3135–3150.

[80] Majid Zamani, Giordano Pola, Manuel Mazo, and Paulo Tabuada. 2012. Symbolic models for nonlinear control systems without stability assumptions. *IEEE Transactions on Automatic Control* 57, 7 (2012), 1804–1809.

[81] Paolo Zuliani, Christel Baier, and Edmund M. Clarke. 2012. Rare-event verification for stochastic hybrid systems. In *Proceedings of the 15th ACM International Conference on Hybrid Systems: Computation and Control*. 217–226.

[82] Paolo Zuliani, André Platzer, and Edmund M. Clarke. 2010. Bayesian statistical model checking with application to simulink/stateflow verification. In *Proceedings of the 13th ACM International Conference on Hybrid systems: Computation and Control*. 243–252.

Received September 2020; revised June 2021; accepted August 2021