Efficiency through Uncertainty: Scalable Formal Synthesis for Stochastic Hybrid Systems

Nathalie Cauchi*†
nathalie.cauchi@cs.ox.ac.uk
Department of Computer Science,
University of Oxford
Oxford, UK

Luca Laurenti*†
luca.laurenti@cs.ox.ac.uk
Department of Computer Science,
University of Oxford
Oxford, UK

Morteza Lahijanian*†
morteza.lahijanian@colorado.edu
Aerospace Engineering Sciences,
University of Colorado Boulder
Boulder, CO, USA

Alessandro Abate†
alessandro.abate@cs.ox.ac.uk
Department of Computer Science,
University of Oxford
Oxford, UK

Marta Kwiatkowska†
marta.kwiatkowska@cs.ox.ac.uk
Department of Computer Science,
University of Oxford
Oxford, UK

Luca Cardelli†
luca.cardelli@cs.ox.ac.uk
Department of Computer Science,
University of Oxford
Microsoft Research, Cambridge
UK

ABSTRACT

This work targets the development of an efficient abstraction method for formal analysis and control synthesis of discrete-time *stochastic hybrid systems* (SHS) with linear dynamics. The focus is on temporal logic specifications over both finite- and infinite-time horizons. The framework constructs a finite abstraction as a class of uncertain Markov models known as *interval Markov decision process* (IMDP). Then, a strategy that maximizes the satisfaction probability of the given specification is synthesized over the IMDP and mapped to the underlying SHS. In contrast to existing formal approaches, which are by and large limited to finite-time properties and rely on conservative over-approximations, we show that the exact abstraction error can be computed as a solution of convex optimization problems and can be embedded into the IMDP abstraction. This is later used in the synthesis step over both bounded- and unbounded-time properties, mitigating the known state-space explosion problem. Our experimental validation of the new approach compared to existing abstraction-based approaches shows: (i) significant (orders of magnitude) reduction of the abstraction error; (ii) marked speed-ups; and (iii) boosted scalability, allowing in particular to verify models with more than 10 continuous variables.

KEYWORDS

formal methods, verification, synthesis, model checking, hybrid systems, stochastic processes, interval Markov decision processes

ACM Reference Format:
Nathalie Cauchi, Luca Laurenti, Morteza Lahijanian, Alessandro Abate, Marta Kwiatkowska, and Luca Cardelli. 2019. Efficiency through Uncertainty: Scalable Formal Synthesis for Stochastic Hybrid Systems. In *22nd ACM International Conference on Hybrid Systems: Computation and Control (HSCC '19), April 16–18, 2019, Montreal, QC, Canada.* ACM, New York, NY, USA, 12 pages. https://doi.org/10.1145/3302504.3311805

1 INTRODUCTION

Stochastic hybrid systems (SHS) are general and expressive models for the quantitative description of complex dynamical and control systems, such as cyber-physical systems. SHS have been used for modeling and analysis in diverse domains, ranging from avionics [5] to chemical reaction networks [6] and manufacturing systems [7]. Many of these applications are *safety-critical*; as a consequence, a theoretical framework providing formal guarantees for analysis and control of SHS is of major importance.

Formal verification and synthesis for stochastic processes and SHS have been the focus of many recent studies [8, 17, 18, 21, 24]. These methods can provide formal guarantees on the probabilistic satisfaction of quantitative specifications, such as those expressed in *linear temporal logic* (LTL). An approach to formal verification, which is particularly relevant for discrete-time models, hinges on the abstraction of continuous-space stochastic models into discrete-space Markov process [8, 16, 17]. This leads to discrepancies between the abstract and original models, which can be captured through error guarantees. The main issue with this approach is its lack of scalability to complex models, which is related to the known state-space explosion problem. This issue is aggravated by the conservative nature of the error bounds; thus, to guarantee a given verification error, a very fine abstraction is generally required, leading to state-space explosion.

This paper introduces a theoretical and computational synthesis framework for discrete-time SHS that is both formal and scalable.

*Nathalie Cauchi has performed all the implementations, and Luca Laurenti and Morteza Lahijanian have equally contributed to the theoretical work.

†This work was supported in part by EPSRC Mobile Autonomy Program Grant EP/M019918/1, Royal Society grant RP120138, Malta's ENDEAVOUR scholarship scheme and the Turing Institute, London, UK.

HSCC '19, April 16–18, 2019, Montreal, QC, Canada

ACM ISBN 978-1-4503-6282-5/19/04…$15.00
https://doi.org/10.1145/3302504.3311805

We zoom in on SHS that take the shape of switching diffusions [23], which are linear in the continuous dynamics and where the control action resides in a mode switch. We focus on two fragments of LTL to encode properties for the SHS, namely *co-safe* LTL (CSLTL) [15], which allows the expression of unbounded and complex reachability properties, and *bounded* LTL (BLTL) [14], which enables the expression of bounded-time and safety properties. The framework consists of two stages (abstraction and control synthesis) and puts forward key novel contributions. In the first step, (i) we introduce a novel space discretization technique that is dynamics-dependent, and (ii) we derive an analytical form for tight (exact) error bounds between the abstraction and the original model, (iii) which is reduced to the solution of a set of convex optimization problems leading to fast computations. The error is formally embedded as uncertain transition probabilities in the abstract model. In the second stage, (iv) a strategy (control policy) is computed by considering only feasible transition probability distributions over the abstract model, preventing the explosion of the error term. Finally, this strategy is soundly refined to a switching strategy for the underlying SHS with guarantees on the computed probability bounds. We provide (v) an illustration of the efficacy of the framework via three case studies, including a comparison with the state of the art. In conclusion, this work provides a new computational abstraction framework for discrete-time SHS that is both formal and markedly more scalable than state-of-the-art techniques and tools.

2 PROBLEM FORMULATION

We consider a SHS and a property of interest given as a temporal logic statement. We are interested in computing a switching strategy that optimizes the probability of achieving the property. Below, we formally introduce the model, property, and problem.

2.1 Stochastic Hybrid Systems

We consider a class of discrete-time SHS with linear continuous dynamics and no resets of the continuous components.

Definition 1 (SHS). *A (discrete-time) linear stochastic hybrid system $\mathcal{H}$ is a tuple $\mathcal{H} = (A, F, G, \Upsilon, L)$, where*

- $A = \{a_1, \ldots, a_{|A|}\}$ *is a finite set of discrete modes, each of which containing a continuous domain $\mathbb{R}^m$, defining the hybrid state space $S = A \times \mathbb{R}^m$,*
- $F = \{F(a) \in \mathbb{R}^{m\times m} \mid a \in A\}$ *is a collection of drift terms,*
- $G = \{G(a) \in \mathbb{R}^{m\times r} \mid a \in A\}$ *is a collection of diffusion terms,*
- $\Upsilon = \{p_1, \ldots, p_n\}$ *is a set of atomic propositions,*
- $L : S \to 2^{\Upsilon}$ *is a labeling function that assigns to each hybrid state possibly several elements of Υ.*

A pair $s = (a, x) \in S$, where $a \in A$ and $x \in \mathbb{R}^m$, denotes a hybrid state of $\mathcal{H}$, and the evolution of $\mathcal{H}$ for $k \in \mathbb{Z}_{\geq 0}$ is a stochastic process $\mathbf{s}(k) = (\mathbf{a}(k), \mathbf{x}(k))$ with values in S. The term $\mathbf{x}$ represents the evolution of the continuous component of $\mathcal{H}$ according to the stochastic difference equation

$$\begin{aligned} \mathbf{x}(k+1) &= F(a)\mathbf{x}(k) + G(a)w, \\ a \in A, \quad w &\sim \mathcal{N}(0, Cov_w), \end{aligned} \tag{1}$$

where $w \in \mathbb{R}^r$ is a Gaussian noise with zero mean and covariance matrix $Cov_w \in \mathbb{R}^{r\times r}$. The signal $\mathbf{a}$ describes the evolution of the discrete modes over time.

For $\kappa \in \mathbb{Z}_{\geq 0} \cup \{\infty\}$, we call $Paths^{\kappa}_{\mathcal{H}} : \{0, 1, \ldots, \kappa\} \to S$ the set of sample paths of $\mathbf{s}$ of length κ. The set of all sample paths with finite and infinite lengths are denoted by $Paths^{\text{fin}}_{\mathcal{H}}$ and $Paths_{\mathcal{H}}$. We denote by $\omega_{\mathcal{H}}$, $\omega^k_{\mathcal{H}}$, and $\omega_{\mathcal{H}}(i)$ a sample path, a sample path of length k, and the $(i+1)$-th state on the path $\omega_{\mathcal{H}}$ of $\mathcal{H}$, respectively.

Definition 2 (Switching strategy). *A* switching strategy *for $\mathcal{H}$ is a function $\sigma_{\mathcal{H}} : Paths^{\text{fin}}_{\mathcal{H}} \to A$ that assigns a discrete mode $a \in A$ to a finite path $\omega_{\mathcal{H}}$ of the process $\mathbf{s}$. The set of all switching strategies is denoted by $\Sigma_{\mathcal{H}}$.*

Given a switching strategy $\sigma_{\mathcal{H}}$, the evolution of $\mathbf{s}(k)$ for $k < \kappa$, is defined on the probability space $(S^{\kappa+1}, \mathcal{B}(S^{\kappa+1}), P)$, where $\mathcal{B}(S^{\kappa+1})$ is the product sigma-algebra on the product space $S^{\kappa+1}$, and P is a probability measure. We call T the *transition kernel* such that for any measurable set $B \subseteq \mathbb{R}^m$, $x \in \mathbb{R}^m$, and $a \in A$,

$$T(B \mid x, a) = \int_B \mathcal{N}(t \mid F(a)x, G(a)^T Cov_w G(a))\, dt, \tag{2}$$

where $\mathcal{N}(t \mid F(a)x, G(a)^T Cov_w G(a))$ is a normal distribution with mean $F(a)x$ and variance $G(a)^T Cov_w G(a)$. Then, it holds that P is uniquely defined by T, and for $k < \infty$,

$$T(B \mid x_k, a_k) = P(\mathbf{x}(k+1) \in B \mid \mathbf{x}(k) = x_k, \mathbf{a}(k) = a_k).$$

We note that, for $\kappa = \infty$, P is still uniquely defined by T by the *Ionescu-Tulcea extension theorem* [2].

We are interested in the properties of $\mathcal{H}$ in set $(A \times X) \subset S$, where $X \subset \mathbb{R}^m$ is a continuous compact set. Specifically, we analyze the behavior of $\mathcal{H}$ with respect to a set of closed regions of interest $R = \{r_1, \ldots, r_n\}$, where $r_i \subseteq X$. To this end, we associate to each region r_i the atomic proposition (label) p_i, i.e., $p_i \in L(s = (a, x)) \Leftrightarrow x \in r_i$. Further, we define the (observation) *trace* of path $\omega^k_{\mathcal{H}} = s_0 s_1 \ldots s_k$ to be

$$\xi = \xi_0 \xi_1 \ldots \xi_k,$$

where $\xi_i = L(s_i) \in 2^{\Upsilon}$ for all $i \leq k$. For a path $\omega_{\mathcal{H}} \in Paths_{\mathcal{H}}$ with infinite length, we obtain an infinite-length trace.

2.2 Temporal Logic Specifications

We employ *co-safe linear temporal logic* (CSLTL) [15] and *bounded linear temporal logic* (BLTL) [14] to write the properties of $\mathcal{H}$. We use CSLTL to encode complex reachability properties with no time bounds, and BLTL to specify bounded-time properties.

Definition 3 (CSLTL syntax). *A CSLTL formula φ over a set of atomic propositions Υ is inductively defined as follows:*

$$\varphi := p \mid \neg p \mid \varphi \vee \varphi \mid \varphi \wedge \varphi \mid \mathcal{X}\varphi \mid \varphi\, \mathcal{U} \varphi \mid \mathcal{F}\varphi,$$

where $p \in \Upsilon$, $\neg$ (negation), $\vee$ (disjunction), and $\wedge$ (conjunction) are Boolean operators, and $\mathcal{X}$ ("next"), $\mathcal{U}$ ("until"), and $\mathcal{F}$ ("eventually") are temporal operators.

Definition 4 (BLTL syntax). *A BLTL formula φ over a set of atomic propositions Υ is inductively defined as following:*

$$\varphi := p \mid \neg\varphi \mid \varphi \vee \varphi \mid \mathcal{X}\varphi \mid \varphi\, \mathcal{U}^{\leq k} \varphi \mid \mathcal{F}^{\leq k}\varphi \mid \mathcal{G}^{\leq k}\varphi,$$

where $p \in \Upsilon$ is an atomic proposition, $\neg$ (negation) and $\vee$ (disjunction) are Boolean operators, $\mathcal{X}$ ("next"), $\mathcal{U}^{\leq k}$ ("bounded until"), $\mathcal{F}^{\leq k}$ ("bounded eventually"), and $\mathcal{G}^{\leq k}$ ("bounded always") are temporal operators.

Definition 5 (Semantics). *The semantics of csLTL and bLTL path formulas are defined over infinite traces over 2^{Υ}. Let $\xi = \{\xi_i\}_{i=0}^{\infty}$ with $\xi_i \in 2^{\Upsilon}$ be an infinite trace and $\xi^i = \xi_i\xi_{i+1}\ldots$ be the i-th suffix. Notation $\xi \models \varphi$ indicates that ξ satisfies formula φ and is recursively defined as following:*

- $\xi \models p$ *if* $p \in \xi_0$;
- $\xi \models \neg\varphi$ *if* $\xi \not\models \varphi$;
- $\xi \models \varphi_1 \vee \varphi_2$ *if* $\xi \models \varphi_1$ *or* $\xi \models \varphi_2$;
- $\xi \models \varphi_1 \wedge \varphi_2$ *if* $\xi \models \varphi_1$ *and* $\xi \models \varphi_2$;
- $\xi \models \mathcal{X}\varphi$ *if* $\xi^1 \models \varphi$;
- $\xi \models \varphi_1\mathcal{U}\varphi_2$ *if* $\exists k \geq 0$, $\xi^k \models \varphi_2$, *and* $\forall i \in [0, k)$, $\xi^i \models \varphi_1$;
- $\xi \models \mathcal{F}\varphi$ *if* $\exists k \geq 0$, $\xi^k \models \varphi$;
- $\xi \models \varphi_1\mathcal{U}^{\leq k}\varphi_2$ *if* $\exists j \leq k$, $\xi^j \models \varphi_2$, *and* $\forall i \in [0, j)$, $\xi^i \models \varphi_1$;
- $\xi \models \mathcal{F}^{\leq k}\varphi$ *if* $\exists j \leq k$, $\xi^j \models \varphi$;
- $\xi \models \mathcal{G}^{\leq k}\varphi$ *if* $\forall j \leq k$, $\xi^j \models \varphi$.

A trace ξ satisfies a csLTL or bLTL formula φ iff there exists a "good" finite prefix $\underline{\xi}$ of ξ such that the concatenation $\underline{\xi}\overline{\xi}$ satisfies φ for every suffix $\overline{\xi}$ [14, 15]. Therefore, even though the semantics of csLTL and bLTL are defined over infinite traces, we can restrict the analysis to the set of their good prefixes, which consists of finite traces.

2.3 Problem Statement

We say that a finite path $\omega_{\mathcal{H}}$ of $\mathcal{H}$, initialized at state $s_0 \in S$, satisfies a formula φ if the path remains in the compact set X and its corresponding finite trace $\xi \models \varphi$. Under a switching strategy $\sigma_{\mathcal{H}}$, the probability that the sHs satisfies φ is given by:

$$P(\varphi \mid s_0, X, \sigma_{\mathcal{H}}) = P\big(\omega_{\mathcal{H}} \in Paths_{\mathcal{H}}^{\text{fin},\sigma_{\mathcal{H}}} \mid \omega_{\mathcal{H}}(0) = s_0, \omega_{\mathcal{H}}(k) \in (A \times X)\ \forall k \in [0, |\omega_{\mathcal{H}}|\,],\ \xi \models \varphi\big), \quad (3)$$

where $Paths_{\mathcal{H}}^{\text{fin},\sigma_{\mathcal{H}}}$ denotes the set of all finite paths under strategy $\sigma_{\mathcal{H}}$, and ξ is the observation trace of $\omega_{\mathcal{H}}$. In this work, we are interested in synthesizing a switching strategy that maximizes the probability of satisfying property φ.

Problem 1 (Strategy synthesis). *Given the sHs $\mathcal{H}$ in Def. 1, a continuous compact set X, and a property expressed as a csLTL or bLTL formula φ, find a switching strategy $\sigma^*_{\mathcal{H}}$ that maximizes the probability of satisfying φ*

$$\sigma^*_{\mathcal{H}} = \arg\max_{\sigma_{\mathcal{H}} \in \Sigma_{\mathcal{H}}} P(\varphi \mid s_0, X, \sigma_{\mathcal{H}})$$

for all initial states $s_0 \in A \times X$.

2.4 Overview of Proposed Approach

We solve Problem 1 with a discrete abstraction that is both formal and computationally tractable. We construct a finite model in the form of an uncertain Markov process that captures all possible behaviors of the sHs $\mathcal{H}$. This construction involves a discretization of the continuous set X and hence of R. We quantify the error of this approximation and represent it in the abstract Markov model as uncertainty. We then synthesize an optimal strategy on this model that (i) optimizes the probability of satisfying φ, (ii) is robust against the uncertainty and thus (iii) can be mapped (refined) onto the concrete model $\mathcal{H}$. In the rest of the paper, we present this solution in detail and show all the proofs in Appendix A.

3 PRELIMINARIES

3.1 Markov Models

We utilize Markov models as abstraction structures.

Definition 6 (MDP). *A Markov decision process (MDP) is a tuple $\mathcal{M} = (Q, A, P, \Upsilon, L)$, where:*

- *Q is a finite set of states,*
- *A is a finite set of actions,*
- *$P : Q \times A \times Q \to [0, 1]$ is a transition probability function.*
- *Υ is a finite set of atomic propositions;*
- *$L : Q \to 2^{\Upsilon}$ is a labeling function assigning to each state possibly several elements of Υ.*

The set of actions available at $q \in Q$ is denoted by $A(q)$. The function P has the property that $\sum_{q' \in Q} P(q, a, q') = 1$ for all pairs (q, a), where $q \in Q$ and $a \in A(q)$.

A path ω through an MDP is a sequence of states $\omega = q_0 \xrightarrow{a_0} q_1 \xrightarrow{a_1} q_2 \xrightarrow{a_2} \ldots$ such that $a_i \in A(q_i)$ and $P(q_i, a_i, q_{i+1}) > 0$ for all $i \in \mathbb{N}$. We denote the last state of a finite path ω^{fin} by $last(\omega^{\text{fin}})$ and the set of all finite and infinite paths by $Paths^{\text{fin}}$ and $Paths$, respectively.

Definition 7 (Strategy). *A strategy σ of an MDP model $\mathcal{M}$ is a function $\sigma : Paths^{\text{fin}} \to A$ that maps a finite path ω^{fin} of $\mathcal{M}$ onto an action in A. If a strategy depends only on $last(\omega^{\text{fin}})$, it is called a memoryless or stationary strategy. The set of all strategies is denoted by Σ.*[1]

Given a strategy σ, a probability measure *Prob* over the set of all paths (under σ) *Paths* is induced on the resulting Markov chain [3].

A generalized class of MDPs that allows a range of transition probabilities between states is known as *bounded-pa-rameter* [10] or *interval* MDP (IMDP) [12].

Definition 8 (IMDP). *An interval Markov decision process (IMDP) is a tuple $\mathcal{I} = (Q, A, \check{P}, \hat{P}, \Upsilon, L)$, where Q, A, Υ, and L are as in Def. 6, and*

- *$\check{P} : Q \times A \times Q \to [0, 1]$ is a function, where $\check{P}(q, a, q')$ defines the lower bound of the transition probability from state q to state q' under action $a \in A(q)$,*
- *$\hat{P} : Q \times A \times Q \to [0, 1]$ is a function, where $\hat{P}(q, a, q')$ defines the upper bound of the transition probability from state q to state q' under action $a \in A(q)$.*

For all $q, q' \in Q$ and $a \in A(q)$, it holds that $\check{P}(q, a, q') \leq \hat{P}(q, a, q')$ and

$$\sum_{q' \in Q} \check{P}(q, a, q') \leq 1 \leq \sum_{q' \in Q} \hat{P}(q, a, q').$$

[1] We focus on deterministic strategies as they are sufficient for optimality of csLTL and bLTL properties [1, 17, 19].

Let $\mathcal{D}(Q)$ denote the set of discrete probability distributions over Q. Given $q \in Q$ and $a \in A(q)$, we call $\gamma_q^a \in \mathcal{D}(Q)$ a *feasible distribution* reachable from q by a if

$$\check{P}(q,a,q') \leq \gamma_q^a(q') \leq \hat{P}(q,a,q')$$

for each state $q' \in Q$. We denote the set of all feasible distributions for state q and action a by Γ_q^a.

In IMDPs, the notions of paths and strategies are extended from those of MDPs in a straightforward manner. A distinctive concept instead is that of *adversary*, which is a mechanism that selects feasible distributions from interval sets.[2]

Definition 9 (Adversary). *Given an IMDP $\mathcal{I}$, an adversary is a function $\pi : Paths^{\text{fin}} \times A \to \mathcal{D}(Q)$ that, for each finite path $\omega^{\text{fin}} \in Paths^{\text{fin}}$ and action $a \in A(last(\omega^{\text{fin}}))$, assigns a feasible distribution $\pi(\omega^{\text{fin}}, a) \in \Gamma^a_{last(\omega^{\text{fin}})}$.*

Given a finite path ω^{fin}, a strategy σ, and an adversary π, the semantics of a path of the IMDP is as follows. At state $q = last(\omega^{\text{fin}})$, first an action $a \in A(q)$ is chosen by strategy σ. Then, the adversary π resolves the uncertainties and chooses one feasible distribution $\gamma_q^a \in \Gamma_q^a$. Finally, the next state q' is chosen according to the distribution γ_q^a, and the path ω^{fin} is extended by q'.

Given a strategy σ and an adversary π, a probability measure *Prob* over the set of all finite paths *Paths* (under σ and π) is induced by the resulting Markov chain [17].

3.2 Polytopes and their Post Images

We use (convex) polytopes as means of discretization in our abstraction. Let $m \in \mathbb{N}$ and consider the m-dimensional Euclidean space $\mathbb{R}^m$. A full dimensional (convex) *polytope* P is defined as the convex hull of at least $m+1$ affinely independent points in $\mathbb{R}^m$ [11]. The *set of vertices* of P is the set of points $v_1^{\text{P}}, \ldots, v_{n_{\text{P}}}^{\text{P}} \in \mathbb{R}^m$, $n_{\text{P}} \geq m+1$, whose convex hull gives P and with the property that, for any $i = 1, \ldots, n_{\text{P}}$, point v_i^{P} is not in the convex hull of the remaining points $v_1^{\text{P}}, \ldots, v_{i-1}^{\text{P}}, v_{i+1}^{\text{P}}, \ldots, v_{n_{\text{P}}}^{\text{P}}$. A polytope is completely described by its set of vertices,

$$\text{P} = conv(v_1^{\text{P}}, \ldots, v_{n_{\text{P}}}^{\text{P}}), \tag{4}$$

where *conv* denotes the convex hull. Alternatively, P can be described as the bounded intersection of at least $m+1$ closed half spaces. In other words, there exists a $k \geq m+1$, $h_i \in \mathbb{R}^m$, and $l_i \in \mathbb{R}$, $i = 1, \ldots, k$ such that

$$\text{P} = \{x \in \mathbb{R}^m \mid h_i^T x \leq l_i,\ i = 1, \ldots, k\}. \tag{5}$$

The above definition can be written as the matrix inequality $Hx \leq L$, where $H \in \mathbb{R}^{k \times m}$ and $L \in \mathbb{R}^k$.

Given a matrix $\mathcal{T} \in \mathbb{R}^{m \times m}$, the post image of polytope P by $\mathcal{T}$ is defined as [17]:

$$Post(\text{P}, \mathcal{T}) = \left\{\mathcal{T}x \mid x \in \text{P}\right\}.$$

This post image is a polytope itself under the linear transformation $\mathcal{T}$ and can be computed as:

$$Post(\text{P}, \mathcal{T}) = conv\big(\{\mathcal{T}v_i^{\text{P}} \mid 1 \leq i \leq n_{\text{P}}\}\big).$$

[2] In the verification literature for MDPs, the notions of strategy, policy, and adversary are often used interchangeably. The semantics of adversary over IMDPs is however distinguished.

4 SHS ABSTRACTION AS AN IMDP

As the first step to approach Problem 1, we abstract the SHS $\mathcal{H}$ to an IMDP $\mathcal{I} = (Q, A, \check{P}, \hat{P}, \bar{\Upsilon}, L)$. Below we overview the construction of the abstraction, and in Sec. 5, we detail the computations involved.

IMDP States. We perform a discretization of the hybrid state space $A \times X$. For each discrete mode $a \in A$, we partition the corresponding set of interest X into a set of cells (regions) that are non-overlapping, except for trivial sets of measure zero (their boundaries). We assume that each region is a bounded polytope. We denote by $Q^a = \{q_1^a, \ldots, q_{|Q^a|}^a\}$ the resulting set of regions in mode a. To each cell q_i^a, we associate a state of the IMDP $\mathcal{I}$. We overload the notation by using q_i^a for both a region in X, and a state of $\mathcal{I}$, i.e., $q_i^a \in Q$. Therefore, the set $(A \times X) \subset S$ can be represented by $\bar{Q} = \bigcup_{a \in A} Q^a$. The set of IMDP states is $Q = \bar{Q} \cup \{q_{\text{u}}\}$ with q_{u} representing $S \setminus (A \times X)$, namely the complement of $A \times X$.

IMDP Actions and Transition Probabilities. We define the set of actions of $\mathcal{I}$ to be the set of modes A of $\mathcal{H}$, and allow all actions to be available in each state of $\mathcal{I}$, i.e., $A(q) = A$ for all $q \in Q$. We define the one-step transition probability from a continuous state $x \in X$ to region $q \in \bar{Q}$ under action (mode) $a \in A$ to be defined by the transition kernel $T(q \mid x, a)$ in (2). The caveat is that the states of $\mathcal{I}$ correspond to regions in $\mathcal{H}$, and there are uncountably many possible (continuous) initial states (here x) in each region, resulting in a range of feasible transition probabilities to the region q. Therefore, the transition probability from one region to another can be characterized by a range given by the min and max of (2) over all the possible points x in the starting region. Thus, we can now bound the feasible transition probabilities from state $q_i \in \bar{Q}$ to state $q_j \in \bar{Q}$ from below by

$$\gamma_{q_i}^a(q_j) \geq \min_{x \in q_i} T(q_j \mid x, a), \tag{6}$$

and from above by

$$\gamma_{q_i}^a(q_j) \leq \max_{x \in q_i} T(q_j \mid x, a). \tag{7}$$

Thus, for $q_i, q_j \in \bar{Q}$, we can define the extrema $\check{P}$ and $\hat{P}$ of the transition probability of $\mathcal{I}$ according to these bounds.

Similarly, we define the bounds of the feasible transition probabilities to states outside X as

$$\gamma_{q_i}^a(q_u) \geq 1 - \max_{x \in q_i} T(X \mid x, a), \tag{8}$$

$$\gamma_{q_i}^a(q_u) \leq 1 - \min_{x \in q_i} T(X \mid x, a), \tag{9}$$

and consequently set the bounds in $\mathcal{I}$ to be

$$\check{P}(q_i, a, q_u) = 1 - \max_{x \in q_i} T(X \mid x, a), \tag{10}$$

$$\hat{P}(q_i, a, q_u) = 1 - \min_{x \in q_i} T(X \mid x, a), \tag{11}$$

for all $a \in A$ and $q_i \in \bar{Q}$. Finally, since we are not interested in the behavior of $\mathcal{H}$ outside of $A \times X$, we render the state q_{u} of $\mathcal{I}$ absorbing, i.e., $\check{P}(q_u, a, q_u) = \hat{P}(q_u, a, q_u) = 1$, $\forall a \in A$.

IMDP Atomic Propositions & Labels. In order to ensure a correct abstraction of $\mathcal{H}$ by $\mathcal{I}$ with respect to the labels of $\mathcal{H}$ and the set $R = \{r_1, \ldots, r_n\}$, even for discretizations of $A \times X$ that do not respect the regions in R, we represent (possibly conservatively)

each r_i as well as its complement relative to X through the labeling of the states of $\mathcal{I}$. Let

$$r_{n+i} = X \setminus r_i$$

be the complement region of r_i with respect to X. We associate to each r_{n+i} a new atomic proposition p_{n+i} for $1 \le i \le n$. Intuitively, p_{n+i} represents $\neg p_i$ with respect to X. We define the set of atomic propositions for $\mathcal{I}$ to be

$$\bar{\Upsilon} = \Upsilon \cup \{p_{n+1}, \ldots, p_{2n}\}. \tag{12}$$

Then, we design $L : Q \to 2^{\bar{\Upsilon}}$ of $\mathcal{I}$ such that

$$p_i \in L(q) \quad \Leftrightarrow \quad q \subseteq r_i, \tag{13}$$

for all $q \in \bar{Q}$ and $0 \le i \le 2n$, and $L(q_u) = \emptyset$.

With this modeling, we capture (possibly conservatively) all the property regions of $\mathcal{H}$ by the state labels of $\mathcal{I}$. Then, a formula φ over Υ of $\mathcal{H}$ can be easily translated to a formula $\bar{\varphi}$ on $\bar{\Upsilon}$ of $\mathcal{I}$ by rewriting φ into its negation normal form and replacing $\neg p_i$ with p_{n+i}. We note that all csLTL and bLTL formulas can be written in negation normal form without any loss in their expressive power [13].

Remark 1. *The extension of the atomic propositions in* (12) *is not necessary if the discretization of $A \times X$ respects all the regions in R, i.e., $\exists Q_r \subseteq Q$ s.t. $\cup_{q \in Q_r} q = r$ for all $r \in R$.*

5 COMPUTATION OF THE IMDP

In this section, we introduce an efficient and scalable method for space discretization and computation for

$$\min_{x \in q_i} T(q_j \mid x, a), \qquad \max_{x \in q_i} T(q_j \mid x, a). \tag{14}$$

To this end, we first define a *hyper-rectangle* and *proper tranformation function* as follows.

Definition 10 (Hyper-rectangle). *A hyper-rectangle in $\mathbb{R}^m$ is an m-dimensional rectangle defined by the intervals*

$$[v_l^{(1)}, v_u^{(1)}] \times [v_l^{(2)}, v_u^{(2)}] \times \cdots \times [v_l^{(m)}, v_u^{(m)}], \tag{15}$$

where vectors $v_l, v_u \in \mathbb{R}^m$ capture the lower and upper values of the vertices of the rectangle in each dimension, and $v^{(i)}$ denotes the i-th component of vector v.

Definition 11 (Proper transformation). *For a polytope $q \subset \mathbb{R}^m$, the transformation function $\mathcal{T} \in \mathbb{R}^{m \times m}$ is proper if $Post(q, \mathcal{T})$ is a hyper-rectangle.*

We also note that process $\mathbf{x}$ in mode a is Gaussian with one-step covariance matrix

$$Cov_{\mathbf{x}}(a) = G(a)^T Cov_w G(a). \tag{16}$$

Then, we can characterize $T(q \mid x, a)$ analytically as follows.

Proposition 1. *For process $\mathbf{x}$ in mode $a \in A$, let $\mathcal{T}_a = \Lambda_a^{-\frac{1}{2}} V_a^T$ be a transformation function (matrix), where $\Lambda_a = V_a^T Cov_{\mathbf{x}}(a) V_a$ is a diagonal matrix whose entries are eigenvalues of $Cov_{\mathbf{x}}(a)$ and V_a is the corresponding orthonormal (eigenvector) matrix. For a polytopic region $q \subset \mathbb{R}^m$, if $\mathcal{T}_a$ is proper, then it holds that*

$$T(q \mid x, a) = \frac{1}{2^m} \prod_{i=1}^{m} \left(\mathrm{erf}\left(\frac{y^{(i)} - v_l^{(i)}}{\sqrt{2}}\right) - \mathrm{erf}\left(\frac{y^{(i)} - v_u^{(i)}}{\sqrt{2}}\right) \right), \tag{17}$$

where $\mathrm{erf}(\cdot)$ is the error function, and $y^{(i)}$ is the i-th component of vector $y = \mathcal{T}_a F(a) x$, and $v_l^{(i)}, v_u^{(i)}$ are as in (15).

A direct consequence of Proposition 1 is that the optimizations in (14) can be performed on (17) through a proper transformation, as stated by the following corollary.

Corollary 1. *For polytopic regions $q_i, q_j \subset \mathbb{R}^m$ and process $\mathbf{x}$ in mode a, assume $\mathcal{T}_a$ is a proper transformation function with respect to q_j, and define $q_i' = Post(q_i, F(a))$ and*

$$f(y) = \frac{1}{2^m} \prod_{i=1}^{m} \left(\mathrm{erf}\left(\frac{y^{(i)} - v_l^{(i)}}{\sqrt{2}}\right) - \mathrm{erf}\left(\frac{y^{(i)} - v_u^{(i)}}{\sqrt{2}}\right) \right), \tag{18}$$

where v_l and v_u are as in (15). *Then, it holds that*

$$\min_{x \in q_i} T(q_j \mid x, a) = \min_{y \in Post(q_i', \mathcal{T}_a)} f(y),$$
$$\max_{x \in q_i} T(q_j \mid x, a) = \max_{y \in Post(q_i', \mathcal{T}_a)} f(y).$$

The above proposition and corollary show that, for a particular proper transformation function $\mathcal{T}_a$, an analytical form can be obtained for the discrete kernel of the imdp. This is an important observation because it enables efficient computation for the min and max values of the kernel. Therefore, we use a space discretization to satisfy the condition in Proposition 1 as described below.

5.1 Space Discretization

For each mode $a \in A$, we define the linear transformation function (matrix) of

$$\mathcal{T}_a = \Lambda_a^{-\frac{1}{2}} V_a^T, \tag{19}$$

where $\Lambda_a = V_a^T Cov_{\mathbf{x}}(a) V_a$ is a diagonal matrix whose entries are the eigenvalues of $Cov_{\mathbf{x}}(a)$, and V_a is the corresponding orthonormal (eigenvector) matrix. The discretization of the continuous set X in mode a is achieved by using a grid in the transformed space by $\mathcal{T}_a$. That is, we first transform X by $\mathcal{T}_a$, and then discretize it using a grid. This method of discretization guarantees that, for each $q^a \in Q^a$, $Post(q^a, \mathcal{T}_a)$ is a hyper-rectangle, i.e., $\mathcal{T}_a$ is proper. Hence, we can use the result of Proposition 1 and Corollary 1 for the computation of the values in (14).

Remark 2. *For an arbitrary geometry of X, it may not be possible to obtain a discretization such that $\bigcup_{q^a \in Q^a} q^a = X$. Nevertheless, by using a discretization that under-approximates X, i.e., $\bigcup_{q^a \in Q^a} q^a \subseteq X$, in each mode a, we can compute a lower bound on the probability of satisfaction of a given property φ. For a better approximation, the grid can be non-uniform, allowing in particular for smaller cells near the boundary of X, as in [8].*

5.2 Transition Probability Bounds

We distinguish between transitions from $q \in \bar{Q}$ to the states in $\bar{Q}$ and to the state q_u.

5.2.1 Transitions to $q \in \bar{Q}$. We present two approaches to solving the values for (14). The first approach is based on *Karush-Kuhn-Tucker* (kkt) conditions [4], which shed light into the optimization problem and lays down the conditions on where to look for the optimal points, giving geometric intuition. This method boils down

to solving systems of non-linear equations, which turns out to be efficient and exact for low-dimensional systems. In the second approach, we show that the problem reduces to a convex optimization problem, allowing the adoption of existing optimization tools and hence making the approach suitable for high-dimensional systems.

KKT Optimization Approach: In the next theorem, we use the result of Corollary 1 and the KKT conditions [4] to compute the exact values for (14).

THEOREM 1. *For polytopic regions $q_i, q_j \subset \mathbb{R}^m$ and proper transformation matrix $\mathcal{T}_a$ with respect to q_j, let*

$$Post(q_i', \mathcal{T}_a) = \{y \in \mathbb{R}^m \mid Hy \leq b\},$$

where $q_i' = Post(q_i, F(a))$, $H \in \mathbb{R}^{k \times m}$, $b \in \mathbb{R}^m$, and $k \geq m+1$, and introduce the following conditions:

- **Condition 1**: *y is at the center of $Post(q_j, \mathcal{T}_a)$, i.e.,*
$$y = \left(\frac{v_u^{(1)} + v_l^{(1)}}{2}, \ldots, \frac{v_u^{(m)} + v_l^{(m)}}{2}\right).$$
- **Condition 2**: *y is a vertex of $Post(q_i', \mathcal{T}_a)$.*
- **Condition 3**: *y is on the boundary of $Post(q_i', \mathcal{T}_a)$, where $r \geq 1$ of the k half-spaces that define $Post(q_i', \mathcal{T}_a)$ intersect, and*
$$\nabla f(y) = \bar{H}^T \mu,$$
for vector $\mu = (\mu_1, \ldots, \mu_r)$ of non-negative constants, and submatrix $\bar{H} \in \mathbb{R}^{r \times m}$ that contains only the rows of H that correspond to the r-intersecting half-spaces at y.
- **Condition 4**: *y is as in Condition 3, and*
$$\nabla f(y) = -\bar{H}^T \mu,$$
for vector $\mu = (\mu_1, \ldots, \mu_r)$ of non-negative constants, and $\bar{H}$ is defined as in Condition 3.

Then, it follows that the point $y \in Post(q_i', \mathcal{T}_a)$ that satisfies Condition 1 necessarily maximizes $f(y)$. If Condition 1 cannot be satisfied, then the maximum is necessarily given by one of the points that satisfy Condition 2 or 3. Furthermore, the point $y \in Post(q_i', \mathcal{T}_a)$ that minimizes $f(y)$ necessarily satisfies Condition 2 or 4.

Theorem 1 identifies the arguments (points $y \in Post(q_i', \mathcal{T}_a)$) that give rise to the optimal values of T in (14). Then, the actual optimal values of T can be computed by (18) as guaranteed by Corollary 1. Thus, from Theorem 1, an algorithm can be constructed to generate a set of finite candidate points based on Conditions 1-4 and to obtain the exact values of (14) by plugging those points into (18).

In short, Condition 1 maximizes the unconstrained problem and gives rise to the global maximum. Hence, if the center of q_j is contained in $Post(q_i', \mathcal{T}_a)$, no further check is required for maximum. If not, the maximum is given by a point on the boundary of $Post(q_i', \mathcal{T}_a)$. It is either a vertex (Condition 2) or a boundary point that satisfies Condition 3. The minimum is always given by a boundary point, which can be either a vertex or a boundary point that satisfies Condition 4. Note that Conditions 3 and 4 are similar and both state that the optimal value of T is given by a point where the gradient of T becomes linearly dependent on the vectors that are defined by the intersecting half-spaces of $Post(q_i', \mathcal{T}_a)$ at that point. Each of these two conditions defines a system of m equations and $r < m$ variables, which may have a solution only if some of the equations are linear combinations of the others.

The above algorithm computes the exact values for the transition probability bounds. It is computationally efficient for small dimensional systems, e.g., $m < 4$. For large m, however, the efficiency drops because the number of boundary constraints that need to be checked and solved for in Conditions 3 and 4 increases, in the worst case, exponentially with m. Below, we propose an equivalent but more efficient method to compute min and max of T for large dimensional systems, e.g., $m \geq 4$.

Convex Optimization Approach: In order to show how upper and lower bounds of $f(y)$ can be efficiently computed using convex optimization tools, we need to introduce the definition of *concave* and *log-concave* functions.

DEFINITION 12 (CONCAVE FUNCTION). *A function $\partial : \mathbb{R}^m \to \mathbb{R}$ is said to be concave if and only if for $y_1, y_2 \in \mathbb{R}^m$, $\lambda \in [0,1]$*

$$\partial(\lambda y_1 + (1-\lambda) y_2) \geq \lambda \partial(y_1) + (1-\lambda)\partial(y_2).$$

DEFINITION 13 (LOG-CONCAVE FUNCTION). *A function $\partial : \mathbb{R}^m \to \mathbb{R}$ is said to be log-concave if and only if $log \partial(\partial)$ is a concave function. That is, for $y_1, y_2 \in \mathbb{R}^m$, $\lambda \in [0,1]$*

$$\partial(\lambda y_1 + (1-\lambda) y_2) \geq \partial(y_1)^{\lambda} \partial(y_2)^{(1-\lambda)}.$$

In the following proposition, we show that $f(y)$, as defined in Corollary 1, is log-concave. This enables efficient computation of the upper and lower bounds of $f(y)$ through standard convex optimization techniques such as gradient descent or semidefinite programming [**?**]. Hence, we can make use of readily available software tools, e.g., NLopt [**?**], which have been highly optimized in terms of efficiency and scalability.

PROPOSITION 2. *$f(y)$, as defined in Corollary 1, is a log-concave function.*

5.2.2 Transitions to sink state q_u. Here, we focus on the transition probabilities to state q_u in (10) and (11). To this end, we need to compute

$$\max_{x \in q_i} T(X \mid x, a), \quad \min_{x \in q_i} T(X \mid x, a). \tag{20}$$

We can efficiently compute bounds for these quantities by using the results obtained above. The following proposition shows this efficient method of computation.

PROPOSITION 3. *Let $\check{Q}^a$ and $\hat{Q}^a$ be two sets of polytopic regions in mode a such that*

$$\bigcup_{q \in \check{Q}^a} q \subseteq X \subseteq \bigcup_{q \in \hat{Q}^a} q,$$

and $\mathcal{T}_a$ be a proper transformation function for every $q \in \check{Q}^a \cup \hat{Q}^a$, and call

$$f(y, q) = \frac{1}{2^m} \prod_{i=1}^{m} \left(\text{erf}\left(\frac{y^{(i)} - v_{l,q}^{(i)}}{\sqrt{2}}\right) - \text{erf}\left(\frac{y^{(i)} - v_{u,q}^{(i)}}{\sqrt{2}}\right) \right), \tag{21}$$

where $v_{l,q}$ and $v_{u,q}$ are as in (15) for q. Then, it holds that

$$\max_{x \in q_i} T(X \mid x, a) \leq \max_{y \in Post(q_i', \mathcal{T}_a)} \sum_{q \in \hat{Q}^a} f(y, q), \tag{22}$$

$$\min_{x \in q_i} T(X \mid x, a) \geq \min_{y \in Post(q_i', \mathcal{T}_a)} \sum_{q \in \hat{Q}^a} f(y, q), \tag{23}$$

where $q_i' = Post(q_i, F(a))$.

Intuitively, Proposition 3 states that, with a particular choice of discretization, i.e., a grid in the transformed space, the transition probability to X is equal to the sum of the transition probabilities to the discrete regions, where each discrete transition kernel is given by the close-form function $f(y,q)$ in (21). If X cannot be precisely discretized with a grid (in the transformed space), then the upper and lower bounds of the transition probabilities are given by the over- and under-approximating grids ($\hat{Q}^a$ and $\check{Q}^a$), respectively.

Remark 3. *For the computation of the values in* (22) *and* (23)*, Proposition 2 can be applied, making both methods of KKT and convex optimization applicable.*

6 STRATEGY SYNTHESIS AS A GAME

Recall that our objective is, given the compact set X and a BLTL or CSLTL formula φ, to compute a strategy for $\mathcal{H}$ that maximizes the probability of satisfying φ without exiting X. The IMDP abstraction $\mathcal{I}$, as constructed above, captures (possibly conservatively) the behavior of the SHS $\mathcal{H}$ with respect to the regions of interest R within X, and the probabilities of exiting X are encompassed via the state q_{u}. Since state q_{u} is absorbing, the paths of $\mathcal{I}$ are not allowed to exit and re-enter X; as such, the analysis on $\mathcal{I}$ narrows the focus to dynamics within set X, as desired. Therefore, we can focus on finding a strategy for $\mathcal{I}$ that is robust against all the uncertainties (errors) introduced by the discretization of $A \times X$ and which maximizes φ.

The uncertainties in $\mathcal{I}$ can be viewed as the nondeterministic choice of a feasible transition probability from one IMDP state to another under a given action. Therefore, we interpret a synthesis task over the IMDP as a 2-player stochastic game, where Player 1 chooses an action $a \in A$ at state $q \in Q$, and Player 2 chooses a feasible transition probability distribution $\gamma_q^a \in \Gamma_q^a$. Towards robust analysis, we set up this game as adversarial: the objectives of Players 1 and 2 are to maximize and minimize the probability of satisfying φ, respectively. Hence, the goal becomes to synthesize a strategy for Player 1 that is robust against all adversarial choices of Player 2 and maximizes the probability of achieving φ.

In order to compute this strategy, we first translate φ over Υ into its equivalent formula $\bar{\varphi}$ over $\bar{\Upsilon}$. Then, we construct a *deterministic finite automaton* (DFA) $\mathcal{A}_{\bar{\varphi}}$ that precisely accepts all the good prefixes that satisfy $\bar{\varphi}$ [15].

Definition 14 (DFA). *A DFA constructed from a CSLTL or BLTL formula $\bar{\varphi}$ is a tuple $\mathcal{A}_{\bar{\varphi}} = (Z, 2^{\bar{\Upsilon}}, \tau, z_0, Z_{\mathrm{ac}})$, where Z is a finite set of states, $2^{\bar{\Upsilon}}$ is the set of input alphabets, $\tau : Z \times 2^{\bar{\Upsilon}} \to Z$ is the transition function, $z_0 \in Z$ is the initial state, and $Z_{\mathrm{ac}} \subseteq Z$ is the set of accepting states.*

A *finite run* of $\mathcal{A}_{\bar{\varphi}}$ on a trace $\xi = \xi_1 \cdots \xi_n$, where $\xi_i \in 2^{\bar{\Upsilon}}$, is a sequence of states $\mu = z_0 z_1 \ldots z_n$ with $z_i = \tau(z_{i-1}, \xi_i)$ for $i = 1, \ldots, n$. Run μ is called *accepting* if $\mu_n \in Z_{\mathrm{ac}}$. Trace $\xi \models \bar{\varphi}$ iff its corresponding run μ in $\mathcal{A}_{\bar{\varphi}}$ is accepting.

Next, we construct the product IMDP $\mathcal{I}_{\bar{\varphi}} = \mathcal{I} \times \mathcal{A}_{\bar{\varphi}}$, which is a tuple $\mathcal{I}_{\bar{\varphi}} = (Q_{\bar{\varphi}}, A_{\bar{\varphi}}, \check{P}_{\bar{\varphi}}, \hat{P}_{\bar{\varphi}}, Q_{\bar{\varphi}\mathrm{ac}})$, where

$$Q_{\bar{\varphi}} = Q \times Z, \quad A_{\bar{\varphi}} = A, \quad Q_{\bar{\varphi}\mathrm{ac}} = Q \times Z_{\mathrm{ac}},$$

$$\check{P}_{\bar{\varphi}}((q,z),a,(q',z')) = \begin{cases} \check{P}(q,a,q') & \text{if } z' = \tau(z, L(q')) \\ 0 & \text{otherwise,} \end{cases}$$

$$\hat{P}_{\bar{\varphi}}((q,z),a,(q',z')) = \begin{cases} \hat{P}(q,a,q') & \text{if } z' = \tau(z, L(q')) \\ 0 & \text{otherwise,} \end{cases}$$

for all $q, q' \in Q$, $a \in A$, $z \in Z$ and $Q_{\bar{\varphi}\mathrm{ac}}$ is the set of accepting states with respect to the product IMDP. Intuitively, $\mathcal{I}_{\bar{\varphi}}$ contains both $\mathcal{I}$ and $\mathcal{A}_{\bar{\varphi}}$ and hence can identify all the paths of $\mathcal{I}$ that satisfy $\bar{\varphi}$, i.e., the satisfying paths terminate in $Q_{\bar{\varphi}\mathrm{ac}}$ since their corresponding $\mathcal{A}_{\bar{\varphi}}$ runs are accepting. Therefore, the synthesis problem reduces to computing a robust strategy on $\mathcal{I}_{\bar{\varphi}}$ that maximizes the probability of reaching $Q_{\bar{\varphi}\mathrm{ac}}$. This problem is equivalent to solving the *maximal reachability probability problem* [17, 22?] as explained below.

Given a strategy σ on an IMDP, the probability of reaching a terminal state from each state is necessarily a range for all the available adversarial choices of Player 2. Let $\check{p}^{\sigma}(q)$ and $\hat{p}^{\sigma}(q)$ denote lower and upper bounds for the probability of reaching a state in $Q_{\bar{\varphi}\mathrm{ac}}$ starting from $q \in Q_{\bar{\varphi}}$ under σ. Derived from the Bellman equation, we can compute the optimal lower bound by recursive evaluations of

$$\check{p}^{\sigma^*}(q) = \begin{cases} 1 & \text{if } q \in Q_{\bar{\varphi}\mathrm{ac}} \\ \max\limits_{a \in A(q)} \min\limits_{\gamma_q^a \in \Gamma_q^a} \sum\limits_{q' \in Q_{\bar{\varphi}}} \gamma_q^a(q') \check{p}^{\sigma^*}(q') & \text{otherwise,} \end{cases} \tag{24}$$

for all $q \in Q_{\bar{\varphi}}$. Each iteration of this Bellman equation involves a minimization over the adversarial choices, which can be computed through an ordering of the states of $\mathcal{I}_{\bar{\varphi}}$ [10, 17], and a maximization over the actions. This Bellman equation is guaranteed to converge in finite time [17, 22] and results in the lower-bound probability $\check{p}^{\sigma^*}(q)$ for each $q \in Q_{\bar{\varphi}}$ and in a stationary (memoryless in the product) strategy σ^*. The upper bounds are similarly given by recursive evaluations of

$$\hat{p}^{\sigma^*}(q) = \begin{cases} 1 & \text{if } q \in Q_{\bar{\varphi}\mathrm{ac}} \\ \max\limits_{\gamma_q^{\sigma^*} \in \Gamma_q^{\sigma^*}} \sum\limits_{q' \in Q_{\bar{\varphi}}} \gamma_q^{\sigma^*}(q') \hat{p}^{\sigma^*}(q') & \text{otherwise,} \end{cases} \tag{25}$$

which is also guaranteed to converge in finite time.

The optimal strategy σ^* on $\mathcal{I}_{\bar{\varphi}}$ can be mapped onto the states and actions of the abstraction IMDP $\mathcal{I}$, resulting in a (history-dependent) strategy. By construction, then the optimal lower and upper probability bounds of satisfying φ from the states of $\mathcal{I}$ are:

$$\check{p}_{\varphi}^{\sigma^*}(q) = \check{p}^{\sigma^*}((q,z_0)), \quad \hat{p}_{\varphi}^{\sigma^*}(q) = \hat{p}^{\sigma^*}((q,z_0)), \tag{26}$$

for all $q \in Q$ of $\mathcal{I}$.

The complexity of the above strategy synthesis algorithm is polynomial in the size of the IMDP $\mathcal{I}_{\bar{\varphi}}$ [17, 22] and exponential in the size of the formula φ (in the worst case) [15]. Note that the size of φ used to express the properties of SHS is typically small.

7 CORRECTNESS

We show that the strategy σ^* computed over $\mathcal{I}$ can be refined over (mapped onto) $\mathcal{H}$ and the lower probability bound $\check{p}_{\varphi}^{\sigma^*}$ on $\mathcal{I}$ always holds for the hybrid system $\mathcal{H}$. The upper bound $\hat{p}_{\varphi}^{\sigma^*}$ also holds for $\mathcal{H}$ if the discretization respects the regions in R. In the case that the discretization is not R-respecting, a modified upper bound

that holds for $\mathcal{H}$ can be computed with a small additional step as detailed below.

Let $J : S \to Q$ be a function that maps the hybrid states $s \in S$ to their corresponding discrete regions (states of $\mathcal{I}$), i.e, $J(s) = q \in Q$ if $s \in q$. With a slight abuse of notations, we also use J to denote the mapping from the finite paths of $\mathcal{H}$ to their corresponding paths of $\mathcal{I}$, i.e.,

$$\omega^k_{\mathcal{H}} = s_0 s_1 \dots s_k \quad \Rightarrow \quad J(\omega^k_{\mathcal{H}}) = J(s_0)J(s_1)\dots J(s_k).$$

Then, the IMDP strategy σ^* correctly maps to a switching strategy $\sigma^*_{\mathcal{H}}$ for $\mathcal{H}$ via

$$\sigma^*_{\mathcal{H}}(\omega^k_{\mathcal{H}}) = \sigma^*(J(\omega^k_{\mathcal{H}})). \tag{27}$$

The following theorem shows that for a given φ, the probability bounds $\check{p}^{\sigma^*}_{\varphi}$ and $\hat{p}^{\sigma^*}_{\varphi}$ are guaranteed to hold for the process $\mathbf{s}$ under $\sigma^*_{\mathcal{H}}$ as constructed above.

THEOREM 2. *Given a SHS $\mathcal{H}$, a continuous set X, and a CSLTL or BLTL formula φ, let $\mathcal{I}$ be the IMDP abstraction of $\mathcal{H}$ as described in Section 4 through a discretization that respects the regions of interest in R. Further, let σ^* be the strategy on $\mathcal{I}$ computed by* (24) *and* (25) *with probability bounds $\check{p}^{\sigma^*}_{\varphi}$ and $\hat{p}^{\sigma^*}_{\varphi}$ in* (26)*. Refine σ^* into a switching strategy $\sigma^*_{\mathcal{H}}$ as in* (27)*. Then, for any initial hybrid state $s_0 \in S$, where $s_0 \in q_0 \in Q$, it holds that*

$$P(\varphi \mid s_0, X, \sigma^*_{\mathcal{H}}) \in \left[\check{p}^{\sigma^*}_{\varphi}(q_0), \hat{p}^{\sigma^*}_{\varphi}(q_0)\right]. \tag{28}$$

Note that an assumption in Theorem 2 is that the discretization Q respects the regions in R. If this assumption is violated, then the lower bound $\check{p}^{\sigma^*}_{\varphi}$ still holds, unlike the upper bound $\hat{p}^{\sigma^*}_{\varphi}$. That is because we design the labeling function L of $\mathcal{I}$ to under-approximate the regions of interest $r \in R$, making the upper bound $\hat{p}^{\sigma^*}_{\varphi}$ valid with respect to the under-approximate representation of R by L but possibly under-approximated with respect to the actual R. To compute an upper bound that accounts for this, we need to design a new labeling function that over-approximates the labels of each region, as follows. Let $L' : Q \to \bar{\Upsilon}$ be this labeling function with

$$p_i \in L'(q) \quad \Leftrightarrow \quad \exists (a, x) \in q \text{ s.t. } x \in r_i, \tag{29}$$

where $p_i \in \bar{\Upsilon}$ is the associated proposition to $r_i \in R$. Then, we can compute the over-approximated upper bound $\hat{p}'^{\sigma^*}_{\varphi}$ via (25) on the product IMDP $\mathcal{I}'_{\bar{\varphi}}$ constructed using L'.

LEMMA 1. *If abstraction $\mathcal{I}$ is constructed through a discretization that does not respect the regions in R, then*

$$P(\varphi \mid s_0, X, \sigma^*_{\mathcal{H}}) \in \left[\check{p}^{\sigma^*}_{\varphi}(q_0), \hat{p}'^{\sigma^*}_{\varphi}(q_0)\right], \tag{30}$$

where $\hat{p}'^{\sigma^*}_{\varphi}$ is computed via (25) *using the labels in* (29)*.*

Theorem 2 and Lemma 1 guarantee that the satisfaction probability of φ for the process $\mathbf{s}$, solution of the SHS $\mathcal{H}$, is contained in the probability interval computed on the abstraction $\mathcal{I}$. The size of this interval depends on the difference of the one-step transition probability bounds of $\check{P}$ and $\hat{P}$ as well as the embedded approximations in the labeling functions L and L' in $\mathcal{I}$, which can be viewed as the error induced by space discretization of $\mathcal{H}$ cast into the abstraction $\mathcal{I}$. This error can be tuned by the size of the discretization: in particular, in the limit of an infinitely fine grid, the error of the abstraction goes to zero, and the IMDP abstraction is refined into an MDP, namely for all $q, q' \in Q$ and $a \in A(q)$, $\check{P}(q, a, q') \to P(q, a, q') \leftarrow \hat{P}(q, a, q')$.

REMARK 4. *In practice, the interest in synthesis problems is typically on deriving lower bounds for the probability, whereas the upper bound computation is useful for error analysis.*

REMARK 5. *With a simple modification, the proposed framework can be used for verification of SHS $\mathcal{H}$ against property φ: (i) compute the lower-bound probability by replacing $\max_{a \in A(q)}$ with $\min_{a \in A(q)}$ in* (25) *on abstraction $\mathcal{I}$ with labeling function L, and (ii) compute the upper-bound probability by replacing $\min_{\gamma^a_q \in \Gamma^a_q}$ with $\max_{\gamma^a_q \in \Gamma^a_q}$ in* (25) *on abstraction $\mathcal{I}$ with labeling function L'.*

8 EXPERIMENTAL RESULTS

We implement the abstraction and synthesis algorithms and test their performance on three case studies. We first present a two dimensional stochastic process with a single mode and perform a comparison against the algorithms and tool FAUST^2 [9] in Case Study 1. Next, we consider a two dimensional, two-mode model and show the synthesis over unbounded-time properties in Case Study 2. Last, we analyze the scalability of the proposed techniques over increasing continuous dimension of the SHS in Case Study 3.

The implementation of the abstraction algorithm is in MATLAB and C++: more precisely, the approach based on KKT method is in MATLAB (proof of concept), and the convex optimization method with *gradient decent* (GD) is in C++. The IMDP synthesis algorithm is also implemented in C++. The experiments are run on an Intel Core i7-8550U CPU at 1.80GHz × 8 machine with 8 GB of RAM.

8.1 Case Study 1 - Formal Verification

We consider a stochastic process with dynamics in (1) and a single discrete mode ($A = \{a_1\}$), where

$$F(a_1) = \begin{pmatrix} 0.85 & 0 \\ 0 & 0.90 \end{pmatrix}, \quad G(a_1) = \begin{pmatrix} 0.15 & 0 \\ 0 & 0.05 \end{pmatrix},$$

with $X = [-1, 1] \times [-1, 1]$ and safety property

$$\varphi_1 = \mathcal{G}^{\leq k} X.$$

We compare the verification results of the above model using our method against those of the state-of-the-art tool FAUST^2 [9]. Namely, we compare probability of satisfaction of φ_1, computation times, and errors for a range of values for time horizon k and grid sizes. To obtain the IMDP abstraction of our method, we used a uniform grid discretization per Sec. 5. Tool FAUST^2 abstracts the model into an MDP and treats the error as a separate parameter. The grid generated in FAUST^2 is based on computation of the global Lipschitz constant via integrals [9]. We define the error of the IMDP method to be $\varepsilon_q = \hat{p}^*_{\varphi}(q) - \check{p}^*_{\varphi}(q)$ for each state, and the global error to be $\varepsilon_{max} = \max_{q \in Q} \varepsilon_q$. Similarly, for FAUST^2 the resulting error corresponds to the maximum error over all the states. The FAUST^2 tool is written in MATLAB and run over this platform, however additionally for fair comparison we have re-implemented the abstraction based on FAUST^2 in the C++ language (cf. corresponding lines in Table 1).

The results are shown in Table 1 for $k = 2$ and various grid sizes. We saturate conservative errors output by FAUST^2 that are greater than 1 to this value. For the particular grid $|Q| = 3722$, the lower bound probabilities of satisfying φ_1 are shown in Fig. 3 within Appendix B. As evident in Table 1, our approach greatly

outperforms the state of the art. With respect to the error generated for the same grid size, our method has significantly (an order of magnitude) smaller error than FAUST^2. Our IMDP method also requires lower computation times. We also note that, as guaranteed by the theory (Theorem 1 and Proposition 2), both KKT and GD approaches compute the same error.

Tool Method	Impl. Platform	$\lvert\bar{Q}\rvert$ (states)	Time taken (secs)	Error $\varepsilon_{\max}$
IMDP (KKT)	MATLAB	361	19.789	0.211
IMDP (GD)	C++	361	29.003	0.211
FAUST^2	MATLAB	361	108.265	1.000
FAUST^2	C++	361	136.71	1.000
IMDP (KKT)	MATLAB	625	145.563	0.163
IMDP (GD)	C++	625	117.741	0.163
FAUST^2	MATLAB	625	285.795	1.000
FAUST^2	C++	625	302.900	1.000
IMDP (KKT)	MATLAB	1444	4464.783	0.109
IMDP (GD)	C++	1444	510.920	0.109
FAUST^2	MATLAB	1444	1445.441	1.000
FAUST^2	C++	1444	1201.950	1.000
IMDP (KKT)	MATLAB	2601	28127.256	0.082
IMDP (GD)	C++	2601	2939.050	0.082
FAUST^2	MATLAB	2601	5274.578	0.995
FAUST^2	C++	2601	3305.490	0.995
IMDP (KKT)	MATLAB	3721	Time out[3]	-
IMDP (GD)	C++	3721	3973.28	0.068
FAUST^2	MATLAB	3721	11285.313	0.832
FAUST^2	C++	3721	7537.750	0.832

Table 1: Comparison of verification results of our IMDP algorithms against FAUST^2 for φ_1 with $k = 2$.

In Fig. 1, we show the error of each method as a function of the time horizon k in φ_1. From these figures it is evident that our approach again greatly outperforms FAUST^2. That is because our method embeds the error in the abstraction and performs computations according to feasible transition probabilities, which prevents the error from exploding over time, whereas the error of FAUST^2 keeps increasing monotonically with the time horizon. An interesting aspect in Fig. 1a is that the error of our method goes to zero as k increases. That is because the system under consideration is an unbounded Gaussian process, and despite its stable dynamics, the probability of it remaining within the bounded set X approaches zero as time grows larger. This is meaningfully captured by the upper and lower probability bounds of our method. On the other hand, FAUST^2 is not able to capture this behavior and its error explodes.

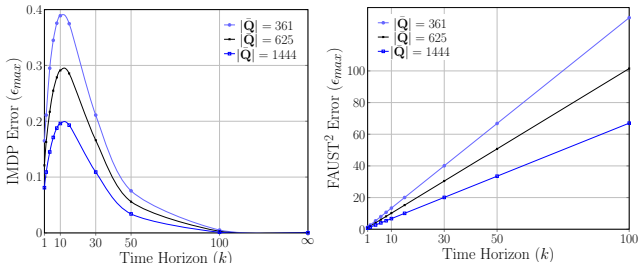
8.2 Case Study 2 - Strategy synthesis

We consider a 2-dimensional SHS with two modes $A = \{a_1, a_2\}$:

$$F(a_1) = \begin{pmatrix} 0.1 & 0.9 \\ 0.8 & 0.2 \end{pmatrix}, \quad G(a_1) = \begin{pmatrix} 0.3 & 0.1 \\ 0.1 & 0.2 \end{pmatrix},$$

$$F(a_2) = \begin{pmatrix} 0.8 & 0.2 \\ 0.1 & 0.9 \end{pmatrix}, \quad G(a_2) = \begin{pmatrix} 0.2 & 0 \\ 0 & 0.1 \end{pmatrix}.$$

Note that $F(a_1)$ and $F(a_2)$ are not asymptotically stable, as they both have one eigenvalue equal to 1. We are interested in synthesizing a



(a) IMDP **(b) FAUST^2**

Figure 1: Maximum error incurred in satisfying φ_1 as a function of time horizon k.

switching strategy that maximizes the probability of satisfying

$$\varphi_2 = \neg red \; \mathcal{U} \; \partial reen.$$

within the set $X = [-2, 2] \times [-2, 2]$. The regions associated with the labels *red* and *green* are depicted in Fig. 2a.

Note that φ_2 has an unbounded time horizon, hence, FAUST^2 cannot be applied. We make use of an adaptive grid, inspired by [8], such that the resulting cells have maximum and minimum sizes in the original space of $\Delta x_{\max} = 0.13$ and $\Delta x_{\min} = 0.05$, respectively. Our adaptive-grid algorithm first over-approximates $Post(X, \mathcal{T}_{a_i})$ for $i \in \{1, 2\}$ by using a uniform grid with the allowed maximum-sized cells. It refines the cells that belong to the green and red regions in the original space, up to the resolution of the minimum-sized cells. Fig. 2c and 2d show the discretization of modes a_1 and a_2, respectively. The generated IMDP has $|Q| = 3612$ states with $|Q^{a_1}| = 1862$ and $|Q^{a_2}| = 1750$. Note that in mode a_1 the cells associated with the label $\neg red$ under-approximate $X \setminus red$, i.e., red is over-approximated, and the regions associated with the label *green* under-approximate the green region. This is due to the transformation function $\mathcal{T}_{a_1}$ that includes a rotation in addition to a translation, which does not respect the regions of interest in R.

We run the synthesis algorithm to obtain the robust strategy $\sigma^*_{\varphi_2}$ with the corresponding lower probability bounds. For each state, the lower probability bounds are depicted in Fig. 2c and 2d. The total time to compute the abstraction and to generate $\sigma^*_{\varphi_2}$ is 5434 seconds. Fig. 2a shows the simulation of two trajectories using $\sigma^*_{\varphi_2}$ with a starting point of $(2, -0.5)$ within mode a_1 and $(-2, 2)$ within mode a_2 respectively. In both instances, the property φ_2 is satisfied.

We also analyze the errors of our method for φ_2 as a function of time horizon for various grid sizes. Fig. 2b shows the results. It can be seen that, for a fixed k, ε_{max} decreases monotonically with the number of states (similar to Fig. 1a in Case Study 1), and ε_{max} converges to a steady-state value for each grid size as the time horizon increases.

8.3 Case Study 3 - Scaling in continuous dimension

We consider a stochastic process with $A = \{a_1\}$ (single mode) and dynamics characterised by $F(a_1) = -0.95\mathbf{I}_d$ and $G(a_1) = 0.1\mathbf{I}_d$, where d corresponds to the continuous dimension of the stochastic process (number of continuous variables) and $X = [-1, 1]^d$. We are interested in checking the specification

$$\varphi_3 = \mathcal{G}^{\leq 50} X$$

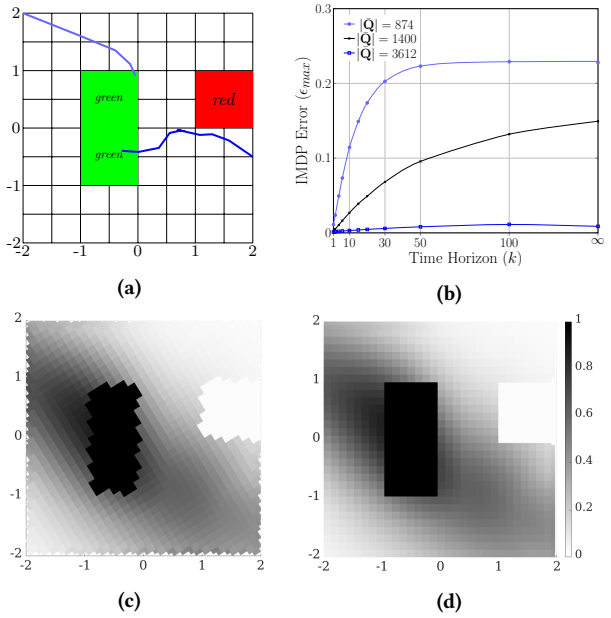


Figure 2: Synthesis results for φ_2 with (a) original set X with simulated trajectories under $\sigma^*_{\varphi_2}$, (b) maximum error incurred in satisfying φ_2 as function of time horizon k, and lower bound probabilities of satisfying φ_2 for modes (c) a_1 and (d) a_2.

as the continuous dimension d of the model varies. We use a uniform grid characterized by parameter $\Delta x = 1$ per side. We compute the corresponding lower- and upper-bound probabilities of satisfying φ_3 and list the number of states required for each dimension together with the associated ε_{max} in Table 2. The method generates abstract models with manageable state spaces, and displays scalability with respect to the continuous dimension d of the SHS to models with more than ten variables, which is a marked improvement over state-of-the-art tools [9].

Dimensions (d)	$\lvert\bar{Q}\rvert$ (states)	Time taken (secs)	Error (ε_{max})
2	4	0.014	0.030
3	14	0.088	0.003
4	30	0.345	0.004
5	62	1.576	0.003
6	125	6.150	0.004
7	254	23.333	0.003
8	510	88.726	0.003
9	1022	367.133	0.003
10	2046	1787.250	0.003
11	8190	25500.000	0.003

Table 2: Verification results of our IMDP approach for φ_3.

9 CONCLUSIONS

This work has presented a theoretical and computational technique for analysis and synthesis of discrete-time stochastic hybrid systems. A suitable choice of the abstraction framework results in exact error bounds, leading to precise and compact abstractions for the synthesis tasks. The experimental results illustrate that the proposed framework greatly outperforms the state of the art time-wise and that is more scalable, thus mitigating the state-space explosion problem. Whilst the framework is tailored to BLTL and CSLTL properties, it can be extended to verification and synthesis for more complex and even multi-objective [12] properties.

REFERENCES

[1] Alessandro Abate, Maria Prandini, John Lygeros, and Shankar Sastry. 2008. Probabilistic reachability and safety for controlled discrete time stochastic hybrid systems. *Automatica* 44, 11 (2008), 2724–2734.

[2] Alessandro Abate, Frank Redig, and Ilya Tkachev. 2014. On the effect of perturbation of conditional probabilities in total variation. *Statistics & Probability Letters* 88 (2014), 1–8.

[3] Christel Baier, Joost-Pieter Katoen, et al. 2008. *Principles of model checking*. Vol. 26202649. MIT press Cambridge.

[4] Dimitri P Bertsekas. 2014. *Constrained optimization and Lagrange multiplier methods*. Academic press.

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

[6] Luca Cardelli, Marta Kwiatkowska, and Luca Laurenti. 2016. A Stochastic Hybrid Approximation for Chemical Kinetics Based on the Linear Noise Approximation. In *Int. Conf. on Computational Methods in Systems Biology*. Springer, 147–167.

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

[8] 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.

[9] Sadegh Esmaeil Zadeh Soudjani, Caspar Gevaerts, and Alessandro Abate. 2015. FAUST^2: Formal Abstractions of Uncountable-STate STochastic Processes.. In *TACAS*, Vol. 15. 272–286.

[10] Robert Givan, Sonia Leach, and Thomas Dean. 2000. Bounded-parameter Markov decision processes. *Artificial Intelligence* 122, 1-2 (2000), 71–109.

[11] Branko Grünbaum, Victor Klee, Micha A Perles, and Geoffrey Colin Shephard. 1967. *Convex polytopes*. Vol. 16. Springer.

[12] Ernst Moritz Hahn, Vahid Hashemi, Holger Hermanns, Morteza Lahijanian, and Andrea Turrini. 2017. Multi-objective Robust Strategy Synthesis for Interval Markov Decision Processes. In *Int. Conf. on Quantitative Evaluation of SysTems (QEST)*. Springer, Berlin, Germany, 207–223.

[13] Mohammad Hekmatnejad and Georgios Fainekos. 2018. Optimal Multi-Valued LTL Planning for Systems with Access Right Levels. In *2018 Annual American Control Conf. (ACC)*. IEEE, 2363–2370.

[14] 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 *CMSB*. Springer, 218–234.

[15] Orna Kupferman and Moshe Y. Vardi. 2001. Model Checking of Safety Properties. *Formal Methods in System Design* 19 (2001), 291–314. Issue 3.

[16] Morteza Lahijanian, Sean B Andersson, and Calin Belta. 2012. Approximate Markovian abstractions for linear stochastic systems. In *IEEE Conf. on Decision and Control (CDC)*,. IEEE, 5966–5971.

[17] Morteza Lahijanian, Sean B Andersson, and Calin Belta. 2015. Formal verification and synthesis for discrete-time stochastic systems. *IEEE Trans. Automat. Control* 60, 8 (2015), 2031–2045.

[18] Luca Laurenti, Alessandro Abate, Luca Bortolussi, Luca Cardelli, Milan Ceska, and Marta Kwiatkowska. 2017. Reachability Computation for Switching Diffusions: Finite Abstractions with Certifiable and Tuneable Precision. In *Proceedings of the 20th Int. Conf. on Hybrid Systems: Computation and Control*. ACM, 55–64.

[19] Ryan Luna, Morteza Lahijanian, Mark Moll, and Lydia E. Kavraki. 2014. Asymptotically Optimal Stochastic Motion Planning with Temporal Goals. In *Int'l Workshop on the Algorithmic Foundations of Robotics (WAFR)*. Istanbul, Turkey, 335–352.

[20] András Prékopa. 1971. Logarithmic concave measures with application to stochastic programming. *Acta Scientiarum Mathematicarum* 32 (1971), 301–316.

[21] Abraham P Vinod, Baisravan Homchaudhuri, and Meeko MK Oishi. 2017. Forward stochastic reachability analysis for uncontrolled linear systems using Fourier transforms. In *Proceedings of the 20th Int. Conf. on Hybrid Systems: Computation and Control*. ACM, 35–44.

[22] Di Wu and Xenofon Koutsoukos. 2008. Reachability analysis of uncertain systems using bounded-parameter Markov decision processes. *Artificial Intelligence* 172, 8-9 (2008), 945–954.

[23] George Yin and Chao Zhu. 2010. *Hybrid switching diffusions: properties and applications*. Vol. 63. Springer New York.

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

A PROOFS

A.1 Proof of Proposition 1

Proof. For a fixed $a \in A$, recall that

$$T(q \mid x, a) = \int_q \mathcal{N}(t \mid F(a)x, Cov_{\mathbf{x}}(a))\, dt,$$

where $Cov_{\mathbf{x}}(a) = G^T(a) Cov_w G(a)$. By applying a whitening through the transformation matrix $\mathcal{T}_a = \Lambda_a^{-\frac{1}{2}} V_a^T$, we obtain that $\mathcal{T}_a Cov_{\mathbf{x}}(a) \mathcal{T}_a^T = \mathbf{I}$, where $\mathbf{I}$ is the identity matrix. Thus, by working in the transformed space induced by $\mathcal{T}_a$, we obtain

$$T(q \mid x, a) = \int_{Post(q, \mathcal{T}_a)} \mathcal{N}\big(t \mid \mathcal{T}_a F(a)x, \mathbf{I}\big)\, dt.$$

Under the assumption that $Post(q, \mathcal{T}_a)$ is a hyper-rectangle, the above multidimensional integral can be separated and expressed as a product of m integrals of uni-dimensional normal distributions:

$$\begin{aligned} T(q \mid x, a) &= \int_{Post(q, \mathcal{T}_a)} \mathcal{N}\big(t \mid \mathcal{T}_a F(a)x, \mathbf{I}\big)\, dt \\ &= \int_{v_l^{(1)}}^{v_u^{(1)}} \cdots \int_{v_l^{(m)}}^{v_u^{(m)}} \mathcal{N}(t_1 \mid y^{(1)}, 1) \cdots \mathcal{N}(t_m \mid \\ &\qquad\qquad y^{(m)}, 1)\, dt_1 \cdots dt_m \\ &= \prod_{i=1}^{m} \int_{v_l^{(i)}}^{v_u^{(i)}} \mathcal{N}\big(t_i \mid y^{(i)}, 1\big)\, dt_i \\ &= \prod_{i=1}^{m} \frac{1}{2}\Big(\mathrm{erf}(\frac{y^{(i)} - v_l^{(i)}}{\sqrt{2}}) - \mathrm{erf}(\frac{y^{(i)} - v_u^{(i)}}{\sqrt{2}})\Big), \end{aligned}$$

where $y = \mathcal{T}_a F(a)x$. □

A.2 Proof of Theorem 1

Proof. We first consider the maximum case and then discuss the minimum case. The KKT conditions guarantee that if $y \in Post(q_i', \mathcal{T}_a)$ is a local maximum for f, then there must exist a vector of constants $\mu = (\mu_1, \ldots, \mu_k)$ such that $\nabla f(y) = H^T \mu$, $\mu_i \geq 0$ for all $i \in \{1, \ldots, k\}$, and $\mu_i(\sum_{j=1}^{m} H^{(i,j)} y^{(j)} - b_i) = 0$, where $H^{(i,j)}$ is the component in the i-th row and j-th column of matrix H. Note that we have a constant $\mu_i, i \in \{1, \ldots, k\}$, for each of the half-paces defining $Post(q_i', \mathcal{T}_a)$. Thus, there are three possible cases:

Case 1: x^* is not in the boundary of $Post(q_i', \mathcal{T}_a)$. In this case the KKT conditions imply that y is a maximum only if $\nabla f(y) = 0$. For a normal distribution with identity covariance, this point is exactly $y = (\frac{v_u^{(i)} + v_l^{(1)}}{2}, \ldots, \frac{v_u^{(m)} + v_l^{(m)}}{2})$. If $y \in Post(q_i', \mathcal{T}_a)$, then this is the global maximum, because it is the global maximum of the unconstrained problem.

Case 2: x^* is a vertex of $Post(q_i', \mathcal{T}_a)$. We call a vertex an intersection of m half-spaces. As a consequence, we have that the KKT conditions are satisfied in y, vertex of $Post(q_i', \mathcal{T}_a)$, if and only if $\nabla f(y) = \bar{H}^T \mu$, where $\bar{H}$ is the submatrix that contains only the m rows of H representing the half-spaces interesting at y, and vector μ contains only the m corresponding constants. Thus, we have a system of m equations and m variables that has solution for $\mu_i \in \mathbb{R}$. However, since the set of vertices is finite, it is generally faster to just include all the vertices as possible candidate solutions instead of solving the system of equations.

Case 3: y is in the boundary of $Post(q_i', \mathcal{T}_a)$, but is not a vertex. In this case only $r < m$ of the half-spaces in H intersect at y. Thus, if y is a maximum then $\nabla f(y) = \bar{H}^T \mu$, where $\bar{H}$ is the submatrix of H containing the $r < m$ half-spaces intersecting at y, and μ contains only the r corresponding constants. Note that this is a system with more equations than variables. Therefore, only when some of constraints become linearly dependent, there may be a solution for $y \in Post(q_i', \mathcal{T}_a)$, if at all.

The minimum case is identical except that condition $\nabla f(y) = H^T \mu$ is replaced with $\nabla f(y) = -H^T \mu$. □

A.3 Proof of Proposition 2

Proof. By Definition we have

$$f(y) = \prod_{i=1}^{m} \bar{f}(y^{(i)} \mid v_l^{(i)}, v_u^{(i)}),$$

where

$$\bar{f}(y^{(i)} \mid v_l^{(i)}, v_u^{(i)}) = \frac{1}{2}\big(\mathrm{erf}(\frac{y^{(i)} - v_l^{(i)}}{\sqrt{2}}) - \mathrm{erf}(\frac{y^{(i)} - v_u^{(i)}}{\sqrt{2}})\big)$$

with $v_u^{(i)} > v_l^{(i)}$. Now, since a product of log-concave functions is a log-concave function itself, to show that $f(y)$ is log-concave, it is enough to show that $\bar{f}(y^{(i)} \mid v_l^{(i)}, v_u^{(i)})$ is log-concave for $i \in \{1, \ldots, m\}$. In order to do that we first need to observe that

$$\bar{f}(y^{(i)} \mid v_l^{(i)}, v_u^{(i)}) = \int_{y^{(i)} - v_u^{(i)}}^{y^{(i)} - v_l^{(i)}} \mathcal{N}(t \mid 0, 1) dt.$$

That is, $\bar{f}$ induces a standard Gaussian probability measure $\bar{P}$. We denote with $\bar{P}([y^{(i)} - v_u^{(i)}, y^{(i)} - v_l^{(i)}])$ the resulting probability for convex Borel set $[y^{(i)} - v_u^{(i)}, y^{(i)} - v_l^{(i)}]$. By rearranging terms, for $\lambda \in [0, 1], y_1, y_2 \in \mathbb{R}$, we finally obtain

$$\begin{aligned} &\bar{f}(\lambda y_1 + (1-\lambda) y_2 \mid v_l^{(i)}, v_u^{(i)}) = \\ &\bar{P}(\lambda[y_1 - v_u^{(i)}, y_1 - v_l^{(i)}] + (1-\lambda)[y_2 - v_u^{(i)}, y_2 - v_l^{(i)}]) \geq \\ &\bar{P}([y_1 - v_u^{(i)}, y_1 - v_l^{(i)}])^{\lambda} \bar{P}[y_2 - v_u^{(i)}, y_2 - v_l^{(i)}])^{1-\lambda} = \\ &\bar{f}(y_1 \mid v_l^{(i)}, v_u^{(i)})^{\lambda} \bar{f}(y_2 \mid v_l^{(i)}, v_u^{(i)}))^{(1-\lambda)}, \end{aligned}$$

where the above inequality is due to Theorem 2 in [20]. □

A.4 Proof of Proposition 3

Proof. For the upper bound, we have that for $q_i \in Q_{\text{safe}}$ and $a \in A$,

$$\begin{aligned} \max_{x \in q_i} T(X \mid x, a) &\leq \max_{x \in q_i} \int_X \mathcal{N}(z \mid F(a)x, Cov_{\mathbf{x}}(a)\, dz \\ &= \max_{y \in Post(q_i', \mathcal{T}_a)} \int_{Post(X, \mathcal{T}_a)} \mathcal{N}(z \mid y, \mathbf{I})\, dz \\ &\leq \max_{y \in Post(q_i', \mathcal{T}_a)} \sum_{q \in \bar{Q}^a} \int_{Post(q, \mathcal{T}_a)} \mathcal{N}(z \mid y, \mathbf{I})\, dz \\ &= \max_{y \in Post(q_i', \mathcal{T}_a)} \sum_{q \in \bar{Q}^a} f(y, q). \end{aligned}$$

For the lower bound, similarly to the upper bound, we have that

$$\min_{x\in q_i} T(X \mid x,a) \geq \min_{y\in Post(q_i',\mathcal{T}_a)} \sum_{q\in Q^a} f(y,q).$$

□

A.5 Proof of Theorem 2

For each φ, let $\mathcal{A}_{\bar{\varphi}} = (Z, 2^{\bar{\Upsilon}}, \tau, z_0, Z_{\text{ac}})$ be the DFA correspondent to φ with initial state z_0. Then, $P(\varphi \mid x, X, \sigma^*_{\mathcal{H}})$ can be computed on the product stochastic hybrid system $\mathcal{H}_\varphi = \mathcal{H} \times \mathcal{A}_{\bar{\varphi}} = (A \times Z, F_\varphi, G_\varphi, \Upsilon, L_\varphi)$, where $L_\varphi(x,(a,z)) = L((a,x))$, $F_\varphi(a,z) = F(a)$ and $G_\varphi(a,z) = G(a)$. We define the set of accepting states of $\mathcal{H}_\varphi$ as $X_{ac} = X \times A \times Z_{\text{ac}}$. It is possible to show that $P(\varphi \mid x, X, \sigma^*_{\mathcal{H}})$ can be computed as the solution of the following Bellman equation

$$V(z_0,x,X,\sigma^*_{\mathcal{H}}) = \begin{cases} 1 & \text{if } (x,\sigma^*_{\mathcal{H}}(x),z_0) \in X_{ac} \\ 0 & \text{if } x \notin X \\ \int_X f(x'|x,\sigma^*_{\mathcal{H}}(x_0)) V(\tau(z_0, L(x,\sigma^*_{\mathcal{H}}(x)), x', X, \sigma^*_{\mathcal{H}}) dx & \end{cases} \tag{31}$$

where $f(x'|x,\sigma^*_{\mathcal{H}}(x_0))$ the density function of transition kernel T and, with an abuse of notation, we call $\sigma^*_{\mathcal{H}}(x_0)$ the action resulting from the application of the (stationary) strategy $\sigma^*_{\mathcal{H}}$ in x_0. For $q \in Q$ call

$$\check{V}^{\sigma^*_{\mathcal{H}}}(z,q,X,\sigma^*_{\mathcal{H}}) = \min_{x\in q} V(z,x,X,\sigma^*_{\mathcal{H}}).$$

Then, it follows that

$$\check{V}^{\sigma^*_{\mathcal{H}}}(z_0,q,X,\sigma^*_{\mathcal{H}}) = \begin{cases} 1 & \text{if there exists } x \in q \text{ s.t. } (x,\sigma^*_{\mathcal{H}}(x),z_0) \in X_{ac} \\ 0 & \text{if } x \notin X \\ \min_{x\in q} \int_X f(x'|x,\sigma^*_{\mathcal{H}}(x)) V(\tau(z_0, L(x,\sigma^*_{\mathcal{H}}(x)), x', X, \sigma^*_{\mathcal{H}}) dx' & \end{cases}$$

Then, because for each $x_1, x_2 \in q$ it holds that $\sigma^*_{\mathcal{H}}(x_1) = \sigma^*_{\mathcal{H}}(x_2)$ and Q_φ is a discretization of X that respects the propositional regions, we obtain

$$\check{V}^{\sigma^*_{\mathcal{H}}}(z_0,q,X,\sigma^*_{\mathcal{H}}) \leq \begin{cases} 1 & \text{if there exists } x \in q \text{ s.t. } (x,\sigma^*_{\mathcal{H}}(x),z_0) \in X_{ac} \\ 0 & \text{if } x \notin X \\ \min_{x\in q} \sum_{q\in Q_\varphi} T(q|x,\sigma^*_{\mathcal{H}}(x)) \check{V}^{\sigma^*_{\mathcal{H}}}(\tau(z_0, L(x,\sigma^*_{\mathcal{H}}(x)), x', X, \sigma^*_{\mathcal{H}}) & \end{cases}$$

The latter expression is exactly (24) for a fixed strategy $\sigma^*_{\mathcal{H}}$. Similar approach can be used to prove that the solution of (31) is upper bounded by (25).

A.6 Proof of Lemma 1

Q_φ is a discretization of X that does not respect the propositional regions R, and the labeling function L of $\mathcal{I}$ introduces an under approximation of those regions. Similar to the proof of Theorem 2, a product SHS $\mathcal{H}_\varphi$ can be constructed. By replacing the discretization Q_φ in the Bellman equation and noting that L under-approximates R, it holds that $\check{V}^{\sigma^*_{\mathcal{H}}}(z_0,q,X,\sigma^*_{\mathcal{H}})$ is an under-approximation of $P(\varphi \mid s_0, X, \sigma^*_{\mathcal{H}})$.

For the upper bound, note that the labeling function L' over-approximates the labels of each region. With the same derivation as above but using L' instead of L, it follows that

$$\hat{V}^{\sigma^*_{\mathcal{H}}}(z,q,X,\sigma^*_{\mathcal{H}}) \geq P(\varphi \mid s_0, X, \sigma^*_{\mathcal{H}}),$$

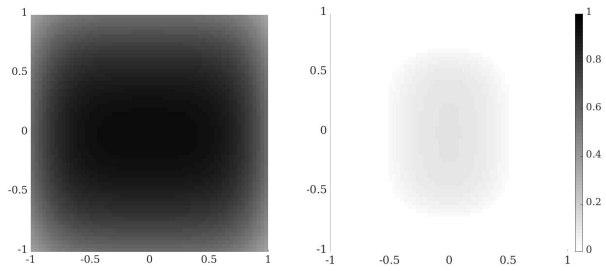
where

$$\hat{V}^{\sigma^*_{\mathcal{H}}}(z,q,X,\sigma^*_{\mathcal{H}}) = \max_{x\in q} V(z,x,X,\sigma^*_{\mathcal{H}}),$$

and $V(z,x,X,\sigma^*_{\mathcal{H}})$ is defined in (31).

B CASE STUDY 1

We present the lower bound probabilities of satisfying φ_1 using both IMDP and FAUST^2 based abstractions, for the particular grid $|Q| = 3722$ in Fig. 3. This further highlights that our approach greatly outperforms the state of the art with respect to probability of satisfaction for the same size of the grid.



(a) IMDP (b) FAUST^2

Figure 3: Lower bound probabilities of satisfying φ_1 with $|\bar{Q}| = 3721$ and $k = 2$.