Data-Driven Verification of Stochastic Linear Systems with Signal Temporal Logic Constraints

Ali Salamati[a], Sadegh Soudjani[b], Majid Zamani[a,c]

[a]Department of Computer Science, Ludwig-Maximilians-Universität München, Germany, (e-mail: ali.salamati@lmu.de)
[b]School of Computing, Newcastle University, United Kingdom, (e-mail: sadegh.soudjani@newcastle.ac.uk)
[c]Department of Computer Science, University of Colorado Boulder, the USA, (e-mail: majid.zamani@colorado.edu)

Abstract

Cyber-physical systems usually have complex dynamics and are required to fulfill complex tasks. In recent years, formal methods from Computer Science have been used by control theorists for both describing the required tasks and ensuring that they are fulfilled by the systems. The crucial drawback of formal methods is that a complete model of the system often needs to be available. The main goal of this paper is to study formal verification of linear time-invariant systems with respect to a fragment of temporal logic specifications when only a partial knowledge of the model is available, i.e., a parameterized model of the system is known but the exact values of the parameters are unknown. We provide a probabilistic measure for the satisfaction of the specification by trajectories of the system under the influence of uncertainty. We assume these specifications are expressed as signal temporal logic formulae and provide an approach that relies on gathering input-output data from the system and employs Bayesian inference on the collected data to associate a notion of confidence to the satisfaction of the specification. The main novelty of our approach is to combine both data-driven and model-based techniques in order to have a two-layer probabilistic reasoning over the behavior of the system. The inner layer is with respect to the uncertainties in dynamics and observed data while the outer layer is with respect to the distribution over the parameter space. The latter is updated using Bayesian inference on the collected data. The proposed approach is demonstrated in two case studies.

Keywords: Bayesian Inference, Data-Driven Methods, Verification, Synthesis, Signal Temporal Logic, Parameterized Models.

1. Introduction

Nowadays, data-driven methods are being used extensively in many engineering applications. However, they suffer from several limitations in terms of accuracy and confidence. Due to the complexity of safety-critical cyber-physical systems (CPS), e.g., self-driving cars and traffic networks, there is a huge demand towards formal guarantees for the correctness of existing data-driven methods [2, 11]. On the other hand, formal methods can provide such guarantees when a model of the system is available. However, the main challenge which most model-based techniques face is the lack of a precise model of the system. This motivates the need for combining data-driven methods with formal techniques that will lead to more efficient formal method algorithms [1].

Formal methods have been vastly used in the realm of Computer Science to provide correctness guarantees on the expected behavior of a program. Most of these formal techniques have been developed for finite-state models [4, 5]. In order to fully utilize the advantages of formal techniques in real physical applications, one needs to first construct a sufficiently precise model of the system. Usually, it is hard to model a system accurately. Besides, the dynamics of a system may vary in the course of time. In such cases, statistical model checking can be beneficial if all states of the system can be measured [10, 33, 34]. However, statistical model checking usually needs a large number of experiments and is not able to handle synthesis problems directly [34].

In this work, we aim at putting together Bayesian inference and formal verification technique and subsequently provide a probabilistic confidence on satisfying a desired specification by trajectories of a stochastic system. We study formal verification of linear time-invariant (LTI) systems with respect to a fragment of temporal logic specifications when only a partial knowledge of the model is available, i.e., a parameterized model of the system is known but the exact values of the parameters are unknown. We provide a probabilistic measure for the satisfaction of the temporal logic specification by trajectories of the system under the influence of uncertainty. We assume these specifications are expressed by signal temporal logic (STL) formulae [26] and provide an approach that relies on collecting input-output data from the system. We employ Bayesian inference to associate a notion of confidence to the satisfaction of the specification. Our main objective is

[*]This work was supported in part by the H2020 ERC Starting Grant AutoCPS (grant agreement No. 804639) and by the EPSRC-funded CodeCPS project (EP/V043676/1).

to combine both data-driven and model-based techniques for stochastic LTI systems in order to verify the system against STL specifications.

Our approach considers probability thresholds as the lower bounds for the satisfaction of STL specifications by the stochastic trajectories of the system. We under-approximate the feasible parameter sets of the probabilistic constraints by transforming them into algebraic inequalities. Then, confidence values are computed using the obtained feasible sets and distributions of parameters which are updated based on collected data from the systems. We also propose relaxation of the algebraic inequalities in order to reduce the conservativeness of under-approximations.

Related work. A comparison between statistical model checking and probabilistic numerical model checking methods is provided in [38]. A multi-level statistical model checking approach is proposed in [36] for hybrid systems. A novel method is introduced in [29] for learning control Lyapunov-like functions in order to synthesize controllers for nonlinear dynamical systems for stability, safety, and reachability specifications. A data-driven approach was developed in [31] for control of piecewise affine systems with additive disturbances against STL specifications. In [3], concepts from formal modeling and machine learning are exploited to develop methodologies that can identify temporal logic formulae that discriminate different stochastic processes based on observations. In [9], authors propose an approach to approximate the posterior distributions of unknown parameters for nonlinear deterministic systems.

Properties expressed as STL formuale are introduced and used in the literature including the works in [28] and [12]. A new definition for probabilistic STL formulae is introduced in [30] that assigns probabilities to the atomic propositions and then combines them through Boolean operators. A robust treatment of uncertainties under STL constraints is performed in [14] in the framework of model predictive control. An under-approximation of constraints described as probabilistic STL formulae is proposed in [13] and applied to design control strategies for the Barcelona wastewater system [15].

In recent years, researchers also investigated data-driven techniques for formal policy synthesis of dynamical systems due to their applicability to high dimensional spaces. A data-driven approach is proposed in [35] for synthesis of safe digital controllers for sampled-data stochastic nonlinear systems. The learning approach proposed in [8] finds Lyapunov functions for dynamical systems ensuring their stability. The work in [20] applies model-free reinforcement learning for policy synthesis of finite-state models. This method is extended in [25] for continuous-space dynamical systems and finite-horizon specifications under continuity assumptions on the dynamics of the system. The authors in [21] propose a reinforcement learning for the synthesis of continuous-state dynamical systems but the convergence is only demonstrated empirically. The recent approach in [24] applies reinforcement learning for satisfying linear temporal logic (LTL) specifications with convergence guarantees and without requiring any continuity assumption on the system dynamics.

A data-driven and model-based formal verification approach for partially unknown LTI systems is recently developed in [18], [17]. In these works, authors proposed a new method based on Bayesian inference and reachability analysis to provide a confidence based on which a physical system affected by noisy measurements verifies a given bounded-time LTL specification. In [19], a method based on Bayesian inference and model checking is developed for Markov decision processes. The recent results in [32] extend those of [18] and [17] to verification of stochastic LTI systems under specifications expressed as STL formulae. In this work, we extended the results in [32] to verification of fully parameterized LTI systems affected by both process and measurement noises. Furthermore, a more efficient method is proposed in order to under-approximate the feasible region of a special category of stochastic dynamical systems affected by bounded support noise.

Outline of the paper. The structure of the paper is as follows. Section 2 gives definitions, assumptions, and the problem statement. Bayesian inference is introduced in Section 3 for systems affected by both measurement and process noises. Section 4 demonstrates a technique in order to under-approximate the feasible domain of probabilistic STL constraints. Section 5 shows how to compute the feasible set of parameters for stochastic LTI systems. Section 6 gives an approximation of the feasible set as a linear program by substituting a bounded support distribution of the noise for the unbounded Gaussian one. The proposed approach is illustrated on two case studies in Section 7. Finally, we conclude the paper in Section 8. Due to lack of space, only the intuitions behind the proofs of statements are provided.

2. Preliminaries and Problem Formulation

In this section, we give the system definition and the problem statement.

2.1. Parametric LTI Systems

Consider the set of parameterized stochastic linear time-invariant (LTI) models $\Omega := \{\mathrm{M}(\theta) \mid \theta \in \Theta\}$ such that

$$\mathrm{M}(\theta) := \begin{cases} x(t+1) = A(\theta)x(t) + B(\theta)u(t) + Gw(t), \\ \hat{y}(t) = C(\theta)x(t) + D(\theta)u(t), \end{cases} \tag{1}$$

where $x(t) \in \mathbb{R}^n$ is the state, $\hat{y}(t) \in \mathbb{R}^m$ is the output, $u(t) \in \mathcal{U} \subset \mathbb{R}^r$ is the input, and $\theta \in \Theta \subset \mathbb{R}^p$ is the parameter of the model $\mathrm{M}(\theta)$. Here, $\mathcal{U}$ is the set of valid inputs and is assumed to be bounded. The process noise $w\colon \mathbb{R}_{\geq 0} \to \mathbb{R}^n$ is selected to be a zero-mean Gaussian distribution, which has a covariance matrix Σ_w.

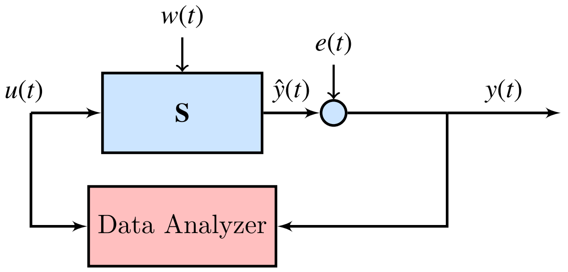


Figure 1: Data collection from the system **S**.

Assumption 1. we assume that our target model S is picked from the class of stochastic dynamical systems and its behavior can be characterized by the model $\mathrm{M}(\theta_{\mathtt{true}})$ for some true parameter $\theta_{\mathtt{true}} \in \Theta$. This true parameter is unknown in general. Further, we assume having access to the output of system S, that is,

$$y(t) = \hat{y}(t) + e(t), \tag{2}$$

in which $e\colon \mathbb{R}_{\geq 0} \to \mathbb{R}^m$ represents the measurement noise with a zero-mean Gaussian distribution and a covariance matrix Σ_e. Both process and measurement noises are assumed to be uncorrelated to the input.

Consider a specification ψ defined over trajectories of the system **S**. We assume ψ belongs to the class of STL specifications which will be defined formally in Subsection 4.1. We denote the satisfaction relation by $\mathbf{S} \models \psi$ which is true when the trajectory of the system **S** satisfies ψ. We plan to provide a confidence value for the satisfaction of ψ by trajectories of **S**. Our approach relies on collecting data from the system and using Bayesian inference to provide the confidence value.

2.2. Data Collection

The process of data collection is depicted in Fig. 1. Let us denote the set of data collected from the system by $\mathcal{D} = \{\tilde{u}_{\mathtt{exp}}(t), \tilde{y}_{\mathtt{exp}}(t)\}_{t=0}^{\mathrm{N}_{\mathtt{exp}}}$, in which $\tilde{u}_{\mathtt{exp}}(t)$ and $\tilde{y}_{\mathtt{exp}}(t)$ are input-output pairs within the time horizon $\{0, \ldots, \mathrm{N}_{\mathtt{exp}}\}$. In general, it is assumed that we can excite the system with any desirable input signal but within the acceptable range of inputs.

Assumption 2. Process noise $\{w(t),\, t = 0, 1, 2, \ldots\}$ and measurement noise $\{e(t),\, t = 0, 1, 2, \ldots\}$ are independent and identically distributed over time, and are independent from each other. In addition, the initial state $x(0)$ is known, and the input $u(t)$ is deterministic.

The assumption on the initial state $x(0)$ can be generalized by allowing it to have a Gaussian distribution independent of $w(\cdot)$ and $e(\cdot)$. Our approach is still applicable to this more general case.

2.3. Stochastic Bayesian Confidence

When the model $\mathrm{M}(\theta)$ is deterministic, the satisfaction relation $\mathrm{M}(\theta) \models \psi$ is a binary relation over the parameter space Θ. This is due to having a unique state trajectory for a given input trajectory. If Ω is the set of parameterized deterministic models, we can define the satisfaction function for the deterministic system as $g_\psi : \Theta \to \{0, 1\}$ in which $g_\psi(\theta) \equiv (\mathrm{M}(\theta) \models \psi)$. This function can only take values that are zero or one. If the system is affected by the process noise, satisfaction relation becomes a random variable over {0,1}. We are interested in computing the probability with which the satisfaction relation holds. In this case, we define a threshold on the satisfaction probability of ψ as

$$\mathbb{P}(\mathrm{M}(\theta) \models \psi) \geq 1 - \delta, \tag{3}$$

where $\delta \in (0, 1)$. Now we can assign a satisfaction function f_ψ^δ to the above chance constraint which is again a binary function on the parameter space Θ.

Definition 1. Consider $\Omega = \{\mathrm{M}(\theta) \mid \theta \in \Theta\}$ with $\mathrm{M}(\theta)$ defined as in (1), and the specification ψ. The satisfaction function $f_\psi^\delta : \Theta \to \{0, 1\}$ is defined as

$$f_\psi^\delta(\theta) = \begin{cases} 1 & \text{if } \mathbb{P}\,(\mathrm{M}(\theta) \models \psi) \geq 1 - \delta, \\ 0 & \text{otherwise,} \end{cases} \tag{4}$$

for any $\delta \in (0, 1)$.

The set of parameters for which $f_\psi^\delta(\theta) = 1$ is called the feasible set of parameters which can be represented as

$$\Theta_\psi := \{\theta \in \Theta \mid f_\psi^\delta(\theta) = 1\}. \tag{5}$$

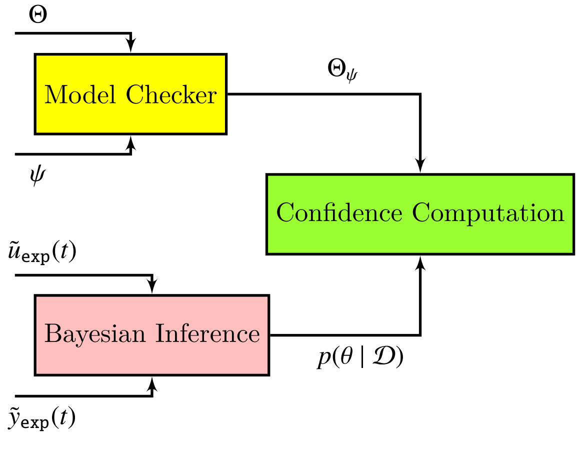
Let us denote by $\mathbb{P}(.)$ and $p(.)$ the probability of an event and the probability density function of a random variable, respectively. We define a probabilistic confidence on satisfaction of the specification using Bayesian inference as follows.

Definition 2. Given a specification ψ and a set of data $\mathcal{D}$, the confidence on satisfaction of ψ by trajectories of the system is

$$\mathbb{P}(\mathbf{S} \models \psi \mid \mathcal{D}) := \int_\Theta f_\psi^\delta(\theta)\; p(\theta \mid \mathcal{D})\, d\theta, \tag{6}$$

where $p(\cdot \mid \mathcal{D})$ is the posteriori distribution on the parameter space conditioned on the input-output data set, and $f_\psi^\delta(\theta)$ is the satisfaction function defined in (4).

Assume that we have a prior knowledge of parameterized models for **S** in the form of some distribution over Θ. This prior knowledge can be used in order to improve the posterior distribution function over Θ after collecting data from the system.



Θ
Model Checker
Θ_ψ
ψ
Confidence Computation
$\tilde{u}_{\mathtt{exp}}(t)$
Bayesian Inference
$p(\theta \mid \mathcal{D})$
$\tilde{y}_{\mathtt{exp}}(t)$

Figure 2: An overview of our proposed approach.

2.4. Problem Statement

Note that the satisfaction function in (4), the feasible set in (5), and the confidence in (6) all depend on the input trajectory of the system. If we require the inequality in (4) to hold for all possible input trajectories, these quantities become independent of the input trajectory. This is indeed a verification problem stated next.

Problem 1 (Verification). Given a parameterized LTI system in (1) together with the noisy output in (2), data set $\mathcal{D}$, and specification ψ, we aim at computing the confidence (6) when $f_\psi^\delta(\theta) = 1$ or equivalently when

$$\mathbb{P}\left(\mathrm{M}(\theta) \models \psi\right) \geq 1 - \delta \quad \forall u(t) \in \mathcal{U}, \forall t \geq 0. \tag{7}$$

A schematic of our proposed approach, which allows us to incorporate any prior information regarding appropriate parameters θ in order to achieve a more precise confidence, is depicted in Fig. 2.

3. Bayesian inference

We use Bayesian inference in order to provide confidences of satisfaction for the given specifications for parametric LTI systems. Given a prior density function over the set of parameters, denoted by $p(\theta)$ and an input-output data set $\mathcal{D}$, a posterior distribution $p(\theta \mid \mathcal{D})$ can be inferred for θ by

$$p(\theta \mid \mathcal{D}) = \frac{p(\mathcal{D} \mid \theta)\ p(\theta)}{\int_\Theta p(\mathcal{D} \mid \theta)\ p(\theta) d\theta}, \tag{8}$$

where $p(\mathcal{D} \mid \theta)$ is the likelihood distribution function. For the dataset $\mathcal{D} = \{\tilde{u}_{\mathsf{exp}}(t), \tilde{y}_{\mathsf{exp}}(t)\}_{t=0}^{\mathrm{N}_{\mathsf{exp}}}$, the likelihood distribution is the joint distribution of all measured outputs in the form of

$$p(\tilde{y}_{\mathsf{exp}}(0), \tilde{y}_{\mathsf{exp}}(1), \ldots, \tilde{y}_{\mathsf{exp}}(\mathrm{N}_{\mathsf{exp}}) \mid \theta). \tag{9}$$

Proposition 1. Consider the LTI model (1)-(2). The joint distribution $p(\mathcal{D} \mid \theta)$ is multi-variate Gaussian with mean

$$\bar{\mathbf{y}}(\theta) = [\bar{y}(0); \cdots ; \bar{y}(\mathrm{N}_{\mathsf{exp}})], \tag{10}$$

and covariance matrix $\Sigma_{\tilde{\mathbf{y}}}(\theta)$, where

$$\begin{aligned}\bar{y}(t) &:= C(\theta)A(\theta)^t x(0) + D(\theta)u(t)\\ &\qquad + \sum_{i=0}^{t-1} C(\theta)A(\theta)^i B(\theta)u(t-i-1),\\ \Sigma_{\tilde{\mathbf{y}}}(\theta) &:= \mathcal{M}(\theta)\ \Sigma_W\ \mathcal{M}(\theta)^T + \Sigma_E,\end{aligned}$$

where $\Sigma_W := \mathrm{diag}(\Sigma_w, \ldots, \Sigma_w)$ and $\Sigma_E := \mathrm{diag}(\Sigma_e, \ldots, \Sigma_e)$ are block diagonal with respectively $\mathrm{N}_{\mathsf{exp}}$ and $(\mathrm{N}_{\mathsf{exp}} + 1)$ blocks. Matrix $\mathcal{M}(\theta) \in \mathbb{R}^{(m\mathrm{N}_{\mathsf{exp}}+m)\times(n\mathrm{N}_{\mathsf{exp}})}$ is represented as:

$$\mathcal{M}(\theta) = \begin{bmatrix} 0 & 0 & 0 & \cdots & 0\\ C(\theta)G & 0 & 0 & \cdots & 0\\ C(\theta)A(\theta)G & C(\theta)G & 0 & \cdots & 0\\ C(\theta)A(\theta)^2G & C(\theta)A(\theta)G & C(\theta)G & \cdots & 0\\ \vdots & \vdots & \vdots & \vdots & \vdots\\ C(\theta)A(\theta)^{\mathrm{N}_{\mathsf{exp}}-1}G & C(\theta)A(\theta)^{\mathrm{N}_{\mathsf{exp}}-2}G & \cdots & \cdots & C(\theta)G \end{bmatrix}.$$

Based on the above Proposition, the joint Gaussian distribution for measured outputs can be characterized as

$$p(\tilde{y}_{\mathsf{exp}}(0), \tilde{y}_{\mathsf{exp}}(1), \ldots, \tilde{y}_{\mathsf{exp}}(\mathrm{N}_{\mathsf{exp}}) \mid \theta) = \frac{1}{|\Sigma_{\tilde{\mathbf{y}}}(\theta)|^{\frac{1}{2}}(2\pi)^{\frac{m\mathrm{N}_{\mathsf{exp}}}{2}}} \exp\left\{-\frac{1}{2}(\tilde{\mathbf{y}} - \bar{\mathbf{y}}(\theta))^T\ \Sigma_{\tilde{\mathbf{y}}}(\theta)^{-1}(\tilde{\mathbf{y}} - \bar{\mathbf{y}}(\theta))\right\}, \tag{11}$$

where, $\tilde{\mathbf{y}} = [\tilde{y}_{\mathsf{exp}}(0); \tilde{y}_{\mathsf{exp}}(1); \cdots ; \tilde{y}_{\mathsf{exp}}(\mathrm{N}_{\mathsf{exp}})]$ is the vector of noisy measured outputs and $\bar{\mathbf{y}}(\theta)$ is defined in (10). $|\Sigma_{\tilde{\mathbf{y}}}(\theta)|$ is determinant of the covariance matrix. The density function (11) can be used to update the posterior distribution using (8).

4. STL and Under-Approximation

4.1. Signal Temporal Logic (STL)

One of the main advantages of STL specifications is their capability in quantifying temporal specifications for trajectories of physical systems. We denote an infinite state trajectory of the system in (1) by $\xi = x(0), x(1), x(2), \ldots$ where $x(t)$ is the state of the system at time $t \in \mathbb{N}_0 := \{0, 1, 2, \ldots\}$. Below, we define syntax and semantics of STL specifications using the standard notation employed in [2, 26].

Syntax: Signal temporal logic (STL) formulae are defined recursively using the following syntax:

$$\psi ::= \mathsf{T} \mid \mu \mid \neg\psi_1 \mid \psi_1 \wedge \psi_2 \mid \psi_1\ \mathsf{U}_{[a,b]}\ \psi_2, \tag{12}$$

where the separator sign | indicates that any specification ψ in this logic can take one of the given five forms, separated by | in (12), and is constructed by combining specifications ψ_1, ψ_2 from this logic. T is the true predicate, and $\mu : \mathbb{R}^n \to \{\mathsf{T}, \mathsf{F}\}$ is a predicate such that its truth value is determined by the sign of a function of the state x, i.e., $\mu(x) = \mathsf{T}$ if and only if $\alpha(x) \geq 0$ with $\alpha : \mathbb{R}^n \to \mathbb{R}$ being an affine function of the state and is associated with μ. Notations $\neg$ and $\wedge$ denote negation and conjunction of formulas. Notation $\mathsf{U}_{[a,b]}$ denotes the until operator where $a, b \in \mathbb{R}_{\geq 0}$ and $a \leq b$.

Semantics: The satisfaction of an STL formula ψ by a trajectory ξ at time t is denoted by $(\xi, t) \models \psi$ which is defined recursively as follows:

$$\begin{aligned}&(\xi, t) \models \mu \Leftrightarrow \mu(\xi, t) = \mathsf{T}\\ &(\xi, t) \models \neg\mu \Leftrightarrow \neg((\xi, t) \models \mu)\\ &(\xi, t) \models \psi \wedge \phi \Leftrightarrow (\xi, t) \models \psi \wedge (\xi, t) \models \phi\\ &(\xi, t) \models \psi\ \mathsf{U}_{[a,b]}\ \phi \Leftrightarrow \exists t' \in [t + a, t + b] \text{ s.t. } (\xi, t') \models \phi\\ &\wedge \forall t'' \in [t, t'],\ (\xi, t'') \models \psi.\end{aligned}$$

A trajectory ξ satisfies a specification ψ, denoted by $\xi \models \psi$, if $(\xi, 0) \models \psi$. We also write $\mathbf{S} \models \psi$ to indicate that $\xi \models \psi$ with ξ being the trajectory of the system $\mathbf{S}$ started from the initial condition $x(0)$.

Furthermore, other standard operators can be expressed using the above defined ones. For disjunction, we can write

$\psi \vee \phi := \neg(\neg\psi \wedge \neg\phi)$ and the eventually operator can be defined as $\Diamond_{[a,b]}\psi := \top \mathsf{U}_{[a,b]}\, \psi$. Finally, the always operator is defined as $\Box_{[a,b]}\psi := \neg\Diamond_{[a,b]}\neg\psi$. The horizon of an STL formula, denoted by $len(\psi)$, is the maximum over all upper bounds of intervals on the temporal operators. Intuitively, $len(\psi)$, is the horizon in which satisfaction of $(\xi, t) \models \psi$ should be studied. Let us now denote a finite trajectory by $\xi(t : N) := x(t), x(t{+}1), ..., x(t{+}N)$. For checking $(\xi, t) \models \psi$, it is sufficient to consider a finite trajectory $\xi(t : N)$ with $N = len(\psi)$.

4.2. Under-approximation of STL Constraints

The stochastic satisfaction function defined in (4) requires the exact feasible set of the chance constraint in (3). This feasible set does not have a closed form in general. Previous works tried to find under-approximations of the feasible set. We leverage the proposed procedure in [13] to get an under-approximation of the feasible set. This procedure transforms the chance constraints on the STL specification into similar constraints on the predicates of the specification using the structure of the STL formula. We discuss this procedure in this subsection and show how this under-approximation can be improved in Subsection 4.3.

The next lemma, borrowed from [13], shows how one can transform the chance constraints on the satisfaction of STL formulae into similar constraints on the predicates of formulae. Since STL formulae are defined on trajectories of the system, we write $\xi(t : N) \models \psi$ instead of $\mathrm{M}(\theta) \models \psi$ to indicate satisfaction of ψ by trajectories starting at time t.

Lemma 1. For any STL formula ψ and a value $\delta \in (0, 1)$, probability constraints of the forms $\mathbb{P}(\xi(t : N) \models \psi) \geq 1 - \delta$ and $\mathbb{P}(\xi(t : N) \models \psi) \leq 1 - \delta$ can be transformed into similar constraints on the predicates of ψ based on the structure of ψ.

In the following, we discuss how this transformation is performed.
Case I Negation $\psi = \neg\psi_1$

$$\mathbb{P}(\xi(t : N) \models \psi) \geq \delta \Leftrightarrow \tag{13}$$
$$\mathbb{P}(\xi(t : N) \models \psi_1) \leq 1 - \delta.$$

Case II Conjunction $\psi = \psi_1 \wedge \psi_2$

$$\mathbb{P}(\xi(t : N) \models \psi) \geq \delta \Leftarrow \tag{14}$$
$$\mathbb{P}(\xi(t : N) \not\models \psi_i) \leq \frac{1-\delta}{2},\ i = 1, 2.$$

Case III $\psi = \psi_1 \,\mathsf{U}_{[a,b]}\, \psi_2$

$$\mathbb{P}(\xi(t : N) \models \psi) \geq \delta \Leftarrow \tag{15}$$
$$\mathbb{P}(\Lambda_j) \geq \frac{\delta}{(b - a + 1)},\ j = 1, \ldots, N,$$

in which the events Λ_j are defined as

$$\Lambda_j := \bigwedge_{k=t}^{t+a-1} (\xi(k : N) \models \psi_1)$$
$$\bigwedge_{k=a+t}^{j-1} (\xi(k : N) \models (\psi_1 \wedge \neg\psi_2)) \wedge (\xi(j : N) \models \psi_2). \tag{16}$$

These transformations are based on multiple application of Boole's inequality [11]. Required transformations for the complements of Cases II and III can be derived similarly.

Lemma 1 enables us to write down probabilistic inequalities on the satisfaction of atomic predicates and use them as under-approximations of the original probabilistic STL constraints. These probabilistic inequalities can be equivalently written as algebraic inequalities given that we know the statistical properties of the state trajectories.

In the case of LTI systems under Assumption 2, $x(t)$ is also Gaussian with known mean and covariance. Let us consider predicate $\mu(x) = \{\alpha(x) \geq 0\}$ with $\alpha(x) := \tilde{\theta}_0 + \tilde{\theta}^T x$, for some $\tilde{\theta} \in \mathbb{R}^n$ and $\tilde{\theta}_0 \in \mathbb{R}$. One can write $\mathbb{E}[\alpha(x)] = \tilde{\theta}_0 + \tilde{\theta}^T\, \mathbb{E}[x]$ and $\mathrm{Var}[\alpha(x)] = \tilde{\theta}^T \mathrm{Cov}(x)\tilde{\theta}$. Therefore,

$$\mathbb{P}(\alpha(x) \geq 0) \geq 1 - \delta \;\Leftrightarrow\; \mathbb{P}(\alpha(x) < 0) \leq \delta$$
$$\Leftrightarrow \mathbb{E}[\alpha(x)] + \mathrm{Var}[\alpha(x)]\mathbf{erf}^{-1}(\delta) \geq 0, \tag{17}$$

where $\mathbf{erf}^{-1}(\cdot)$ is the error inverse function defined with $\mathbf{erf}(x) = \frac{1}{\sqrt{\pi}} \int_{-x}^{x} \exp(-t^2)dt$ where $\exp(\cdot)$ denotes the natural exponential function. In the following proposition, we show that the algebraic inequalities of the form (17) are linear with respect to the input.

Proposition 2. Chance constraint $\mathbb{P}(\alpha(x(t)) \geq 0) \geq 1 - \delta$, where $\alpha(x) = \tilde{\theta}_0 {+} \tilde{\theta}^T x$ and $x(t)$ being the state of the stochastic system (1) at time t, can be written as the following constraint that is affine with respect to the input:

$$\sum_{i=0}^{t-1} \tilde{\theta}^T A(\theta)^i B(\theta)\; u(t - i - 1)$$
$$+ \tilde{\theta}_0 + \tilde{\theta}^T A(\theta)^t x(0) + \tilde{\theta}^T \Gamma(\theta, \delta)\tilde{\theta} \geq 0, \tag{18}$$

where

$$\Gamma(\theta, \delta) := \mathbf{erf}^{-1}(\delta) \sum_{i=0}^{t-1} A(\theta)^i G\, \Sigma_w\, G^T (A(\theta)^T)^i. \tag{19}$$

Note that in general $\Gamma(\theta, \delta)$ and the left-hand side of (18) are nonlinear functions of θ. They become polynomial functions of θ if $A(\theta)$ and $B(\theta)$ depend on θ linearly.

4.3. A Less Conservative Approximation

The proposed procedure in Lemma 1 for transforming the chance constraints into similar inequalities on atomic predicates can be very conservative. This is due to the fact that constraints of type $\mathbb{P}(A_1 \cup A_2) \leq \delta$ are conservatively replaced by inequalities $\mathbb{P}(A_i) \leq \delta/2$, $i = 1, 2$. This replacement puts a uniform upper bound on the probability of events A_i and does not create any room for the intersection of these events. In this subsection, we increase the flexibility in the under-approximation and enlarge the feasible set of the probabilistic STL constraint through intermediate weights.

This new under-approximation procedure results in new constraints with a larger number of variables. It is based on the structure of the STL formula similar to the discussion in the previous subsection and has the following three cases:
Case I: Disjunction

$$\begin{aligned}&\mathbb{P}(\xi(t:N) \models (\psi_1 \vee \cdots \vee \psi_\iota \vee \cdots \vee \psi_N)) \geq \delta\\ \Longleftarrow\ &\mathbb{P}(\xi(t:N) \models \psi_\iota) \geq \alpha_\iota\, \delta,\ \iota \in \{1,\dots,N\},\\ &0 \leq \alpha_\iota \leq 1,\ \alpha_1 + \cdots + \alpha_N = 1.\end{aligned} \tag{20}$$

Case II: Conjunction

$$\begin{aligned}&\mathbb{P}(\xi(t:N) \models (\psi_1 \wedge \cdots \wedge \psi_\iota \wedge \cdots \wedge \psi_N)) \geq \delta\\ \Longleftarrow\ &\mathbb{P}(\xi(t:N) \not\models \psi_\iota) \leq \beta_\iota(1-\delta),\ \iota \in \{1,\dots,N\},\\ &0 \leq \beta_\iota \leq 1,\ \beta_1 + \cdots + \beta_N = 1.\end{aligned} \tag{21}$$

Case III: Until

$$\begin{aligned}&\mathbb{P}(\xi(t:N) \models \psi_1 \mathsf{U}_{[a,b]}\, \psi_2) \geq \delta\\ \Longleftarrow\ &\mathbb{P}(\Lambda_j) \geq \gamma_\iota \frac{\delta}{(b-a+1)}, \iota \in \mathbb{N},\\ &0 \leq \gamma_\iota \leq 1,\ \gamma_1 + \cdots + \gamma_N = 1,\end{aligned} \tag{22}$$

in which, Λ_j is defined as in (16).

In relations (20)-(22), α_ι, β_ι, and γ_ι are intermediate weights that regulate the effect of each probabilistic predicate and contributes to a bigger feasible set. If any knowledge about the likelihood of the satisfaction of subformulas in the main formula is available, it can be exploited to select proper values for these parameters to get a less conservative result.

5. Verification of Probabilistic STL Constraints

5.1. Feasible Set Computation

After transforming the probabilistic STL constraints into the algebraic inequalities, as described in Section 4, these inequalities are in the form of (18) which are linear with respect to the input trajectory and must hold for the whole input range. We use robust linear programming to solve those inequalities. Here, the primary robust linear programming problem is converted to another dual linear programming without a universal quantifier over the input based on Farkas' lemma [16]. Assume the set of valid inputs $\boldsymbol{\mathcal{U}}$ is a bounded polytope characterized by the linear inequalities $Du \leq d$ for some matrix D and vector d with appropriate dimensions. Define the set of valid input trajectories within horizon $\{0,\dots,(t-1)\}$ with $\boldsymbol{\mathcal{U}} := \{\mathbf{Du} \leq \mathbf{d}\}$, where $\mathbf{u} = [u(0); u(1); \dots; u(t-1)]$, $\mathbf{d} = [d; d; \dots; d]$, and $\mathbf{D} = \mathrm{diag}(D,\dots,D)$.

In the next theorem, we show that the feasible set of the probabilistic predicates at each time step can be characterized by a set of constraints at that time step. The proof of this theorem leverages the dual linear programming in its symmetric form, which requires all variables to be non-negative. Therefore, we extract a lower bound $\mathbf{u}_l$ for the input trajectories and shift the input variables to make them non-negative. This lower bound $\mathbf{u}_l$ is readily computable knowing the bounded polytope containing all the input values.

Theorem 1. Assume that the set of valid input trajectories $\boldsymbol{\mathcal{U}}$ is a bounded polytope of the form $\mathbf{Du} \leq \mathbf{d}$ such that $\mathbf{u} \geq \mathbf{u}_l$. The inequality (18) holds for all $\mathbf{u} \in \boldsymbol{\mathcal{U}}$ if the following set of inequalities is feasible over $\mathbf{z}$,

$$\begin{cases}(\mathbf{d} - \mathbf{D}\mathbf{u}_l)^T\mathbf{z} \leq b(\theta,\delta) + \mathbf{f}(\theta)\mathbf{u}_l,\\ -\mathbf{D}^T\mathbf{z} \leq \mathbf{f}(\theta)^T, \quad \mathbf{z} \geq 0,\end{cases} \tag{23}$$

where

$$\begin{aligned}b(\theta,\delta) &= \tilde{\theta}_0 + \tilde{\theta}^T A(\theta)^t x(0) + \tilde{\theta}^T \Gamma(\theta,\delta)\tilde{\theta}, \qquad (24)\\ \mathbf{f}(\theta) &= \tilde{\theta}^T[B(\theta), A(\theta)B(\theta), A(\theta)^2B(\theta),\dots,A(\theta)^{t-1}B(\theta)],\end{aligned}$$

with $\Gamma(\theta,\delta)$ defined in (19).

Solving constraints (23) simultaneously for all predicates of the STL specification gives the feasible set of parameters θ for the stochastic system S in (1). However, the main challenge of using inequalities of the form (23) as under-approximation of the feasible set is that these inequalities are still nonlinear with respect to θ. In the following subsection we propose two numerical techniques to address this challenge.

5.2. Confidence Computation Techniques

Monte Carlo Method. Considering that the constraints (23) are in general nonlinear with respect to θ, computation of integral in (6) can be done efficiently using Monte Carlo integration. The idea is to choose N random points θ_i uniformly from the bounded region of the parameters and use those values that satisfy all the constraints in (23) associated with the predicates of the STL specification in order to compute the integral in (6). The confidence value Q_{N} computed using Monte Carlo integration is a random variable defined as

$$Q_{\mathrm{N}} := \frac{V}{\mathrm{N}}\sum_{i=1}^{\mathrm{N}} K(\theta_i) \text{ with } K(\theta_i) := f_\psi^\delta(\theta_i)\ p(\theta_i \mid \mathcal{D}),$$

where V is the volume of the parameter space. Here, Q_{N} is an unbiased estimator of the integral. Due to the law of large numbers, Q_{N} converges to the true integral when N goes to infinity. An unbiased estimation of the variance of Q_{N} can be computed as $\mathrm{Var}[Q_{\mathrm{N}}] = \frac{V^2\sigma_{\mathrm{N}}^2}{\mathrm{N}}$ with

$$\sigma_{\mathrm{N}}^2 := \frac{1}{\mathrm{N}-1}\sum_{i=1}^{\mathrm{N}}(K(\theta_i) - \bar{K})^2 \text{ and } \bar{K} := \frac{1}{\mathrm{N}}\sum_{i=1}^{\mathrm{N}} K(\theta_i).$$

Note that the $\mathrm{Var}[Q_\mathrm{N}]$ decreases to zero asymptotically with rate 1/N when N goes to infinity and as long as the sequence $\{\sigma_1^2, \sigma_2^2, \sigma_3^2, \ldots\}$ is bounded. This result does not depend on the number of dimensions of the integral in (6), which is the advantage of Monte Carlo integration.

According to Chebyshev's inequality, one has

$$\mathbb{P}(\mathbb{E}[Q_\mathrm{N}] \in [Q_\mathrm{N} - \varepsilon, Q_\mathrm{N} + \varepsilon]) \geq 1 - \frac{\mathrm{Var}[Q_\mathrm{N}]}{\varepsilon^2}, \qquad (25)$$

for any given $\varepsilon > 0$. By choosing an appropriate number of samples N and computing Q_N, the exact value of the integral lies within the interval $[Q_\mathrm{N} - \varepsilon, Q_\mathrm{N} + \varepsilon]$ with confidence $1 - V^2\sigma_\mathrm{N}^2/\mathrm{N}\varepsilon^2$.

Computing the under-approximation of the confidence in (6) using the Monte Carlo integration requires sampling from the domain Θ and rejecting those that render (23) infeasible. It is also possible to find a sampling domain Θ' tighter than Θ by finding the extreme values of θ for which the inequalities (23) are feasible. This will improve the efficiency of the Monte Carlo integration by requiring a smaller number of samples for a given accuracy.

Piecewise Affine Approximation of the Nonlinear Constraints. Another approach for computing the confidence value in (6) is approximating the nonlinear terms $b(\theta, \delta)$ and $\mathbf{f}(\theta)$ in (24) using piecewise affine (PWA) functions. Then, linear programming can be used in order to approximate the feasible set. PWA approximations have been used recently in formal approaches in order to deal with the nonlinearity in dynamical systems [6, 31].

Assuming that $A(\theta)$ and $B(\theta)$ are twice differentiable with respect to θ, $b(\theta, \delta)$ and $\mathbf{f}(\theta)$ in (24) are also twice differentiable. We can partition their domain into polytopic regions, select a nominal value (θ_0, δ_0) in each region, and rewrite $b(\theta, \delta)$ in each region as

$$b(\theta, \delta) \in (\theta - \theta_0)^T \mathscr{M} + (\delta - \delta_0)\mathscr{N} + \epsilon\mathscr{B}, \qquad (26)$$

where

$$\mathscr{M} := \frac{\partial b(\theta, \delta)}{\partial \theta}\Big|_{(\theta_0, \delta_0)} \quad \text{and} \quad \mathscr{N} := \frac{\partial b(\theta, \delta)}{\partial \delta}\Big|_{(\theta_0, \delta_0)},$$

and ϵ is a bound where

$$\epsilon \geq \frac{1}{2}[(\theta - \theta_0)^T, (\delta - \delta_0)]\ \mathbf{H}\ [(\theta - \theta_0); (\delta - \delta_0)],$$

where $\mathbf{H}$ is the Hessian matrix of $b(\theta, \delta)$. Here, $\mathscr{B}$ denotes the unit interval $[-1, 1]$. A similar approximation holds for $\mathbf{f}(\theta)$. The region of parameters is divided into sufficiently large numbers of regions and then inequalities and equations regarding the satisfaction of STL specifications in (23) will be checked in these regions. In the next lemma, we show that the real feasible set can be constructed in the limit when the number of piecewise regions increases.

Lemma 2. The actual feasible set (23) for the STL specification in (6) can be recovered in the limit by increasing the numbers of regions in PWA approximation of the nonlinear terms in (23).

6. Bounded Support Noise

In this section, we show that if the given matrices A and B in (1) are independent of the parameters θ and are known, the probabilistic inequalities can be under-approximated by inequalities that are linear in terms of inputs. These inequalities can be solved using linear programming efficiently to compute the feasible region of parameters. The essential idea in this approach is to replace the Gaussian distributions with truncated ones while quantifying the induced error. Having a bounded support for the noise enables us to use Chernoff-Hoeffding inequality [13, 23] for the under-approximation. The Chernoff-Hoeffding inequality provides a bound on the tail probability of sum of bounded random variables that depends only on the support of these random variables regardless of the shape of their distributions. First, we formally define the support of a random variable.

Definition 3. For a given random variable ω with values in $\mathbb{R}^\mathrm{n}$ and probability distribution $\mathbb{P}$, consider the set of subsets of $\mathbb{R}^\mathrm{n}$ as

$$\mathscr{A} := \{C \subset \mathbb{R}^\mathrm{n} \mid C \text{ is closed and } \mathbb{P}(\omega \in C) = 1\}.$$

The smallest element of $\mathscr{A}$ with respect to the inclusion property is called the support of ω and is denoted by S_ω.

The next proposition provides an upper bound on the error of the probability of satisfying the specification when the noise distributions are replaced by truncated Gaussian distributions.

Proposition 3. Suppose we consider two distributions for the process noise $w(\cdot)$: one which is Gaussian distribution t_w and the other one which is truncated normal $\bar{t}_w$ with support S_w. We denote the probability measures induced on the trajectories ξ of the system $M(\theta)$ by $\mathbb{P}$ and $\mathbb{P}_\mathrm{t}$, respectively. Then we have

$$\mathbb{P}(\xi \models \psi) - \mathbb{P}_\mathrm{t}(\xi \models \psi) \leq \frac{N\alpha}{1 - \alpha}, \qquad (27)$$

for any specification ψ with horizon N. Here, α is the truncated probability $\alpha := 1 - \int_{S_w} t_w(v)dv$.

Using inequality (27), we under-approximate the chance constraint $\mathbb{P}(\xi \models \psi) \geq 1 - \delta$ with

$$\mathbb{P}_\mathrm{t}(\xi \models \psi) \geq 1 - \bar{\delta}, \quad \bar{\delta} := \delta + \frac{N\alpha}{1 - \alpha}. \qquad (28)$$

Assumption 3. For the rest of this section, we focus on under-approximating (28) when the truncated support of $w(t)$ is S_w and is contained in a hyper-rectangle $[a, b]$ (which is the Cartesian product of intervals with vectors a, b indicating the end points of the intervals). We also assume matrices A and B in (1) are non-parametric.

Next lemma, borrowed from [13], shows the relation between supports of $\alpha(x(t))$ and $w(t)$ given the predicate $\mu(x) = \{\alpha(x) \geq 0\}$ with $\alpha(x) := \tilde{\theta}_0 + \tilde{\theta}^T x$.

Lemma 3. The support of $\alpha(x(t))$ is $S_{\alpha(x(t))} := [\tilde{\theta}_0 + \tilde{a}_t + \tilde{\theta}^T \tilde{C}_t, \tilde{\theta}_0 + \tilde{b}_t + \tilde{\theta}^T \tilde{C}_t]$ where $\tilde{a}_t$ and $\tilde{b}_t$ are weighted sum of a and b obtained using interval arithmetics and $\tilde{C}_t := A^t x(0) + \sum_{i=0}^{t-1} A^i B u(t-i-1)$.

We use Chernoff-Hoeffding inequality to replace (28) with a condition on the expected value of the predicate. The following proposition, used also in [13], describes this approximation. Note that Chernoff-Hoeffding inequality requires a particular constant from the dependency graph of the random variables [23]. In such a graph, the nodes represent random variables and two nodes are connected if and only if their related random variables are dependent.

Proposition 4. The probabilistic inequality $\mathbb{P}_t(\alpha(x(t)) > 0) \geq 1 - \bar{\delta}$ can be under-approximated by the inequality

$$\mathbb{E}_t(\alpha(x(t))) \geq \sqrt{-\nu \log(\bar{\delta}) \sum_{t=1}^{N} (\tilde{b}_t - \tilde{a}_t)^2}, \qquad (29)$$

where $\nu = \mathcal{X}(w)/2$, and $\mathcal{X}(w)$ is the chromatic number of the dependency graph of the noises $w(0), \dots, w(N-1)$.

Note that the chromatic number of a graph $\hat{G}$ is the minimum number of colors needed to color vertices of $\hat{G}$ with no two adjacent vertices sharing the same color. This number is equal to 1 for a graph with no edges (e.g., when disturbances $w(i)$ are independent). The interested authors are referred to [27] and [7] for more information about chromatic number of a graph.

Proposition 5. Under Assumption 3 and using Lemma 3, we can under-approximate constraint (28) with

$$\sum_{i=0}^{t-1} \tilde{\theta}^T A^i B\, u(t-i-1) \\ + \tilde{\theta}_0 + \tilde{\theta}^T A^t x(0) \geq \Gamma(\delta, \tilde{a}, \tilde{b}), \qquad (30)$$

where

$$\Gamma(\delta, \tilde{a}, \tilde{b}) := \sqrt{-\nu \log(\bar{\delta}) \sum_{t=1}^{N} (\tilde{b}_t - \tilde{a}_t)^2}. \qquad (31)$$

Note that since $\bar{\delta} \in (0,1)$ and, hence, $\log(\delta) < 0$, the right hand side of the inequality (30) becomes a real value and one has a linear inequality in terms of input. Finally, the next theorem shows that the feasible set of the chance-constraints on the predicates can be approximated by a set of linear constraints.

Theorem 2. Assume that the set of input trajectories $\boldsymbol{\mathcal{U}}$ is a bounded polytope of the form $\mathbf{Du} \leq \mathbf{d}$, $\forall \mathbf{u} \in \boldsymbol{\mathcal{U}}$. The inequality (30) holds for all $\mathbf{u} \in \boldsymbol{\mathcal{U}}$ if the set of linear inequalities (23) is feasible over $\mathbf{z}$, where

$$b(\theta, \delta) = \tilde{\theta}_0 + \tilde{\theta}^T A^t x(0) - \Gamma(\delta, \tilde{a}, \tilde{b}), \qquad (32)$$
$$\mathbf{f}(\theta) = \tilde{\theta}^T [B, AB, A^2 B, \dots, A^{t-1} B],$$

with $\Gamma(\delta, \tilde{a}, \tilde{b})$ defined in (31).

Remark 1. In presenting our approach in this section, we assumed that parameters $\tilde{\theta}_0$ and $\tilde{\theta}$ of the predicate $\alpha(x(t))$ are known. We emphasize that our approach is still valid if $\tilde{\theta}_0$ and $\tilde{\theta}$ depend on the unknown parameters θ of the model. This case can happen when the predicate is defined on the output $\hat{y}(t)$ instead of the state $x(t)$ of the system. The experimental results in the next section demonstrate this case as well.

7. Experimental Results

7.1. Verification Case Study: Unbounded Support Noise

Consider a parameterized class of models $\mathbf{M}(\theta)$ with the state-space representation

$$x(t+1) = \begin{bmatrix} a & 0 \\ 1-a^2 & a \end{bmatrix} x(t) + \begin{bmatrix} \sqrt{1-a^2} \\ -a\sqrt{1-a^2} \end{bmatrix} u(t) + \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} w(t),$$
$$\hat{y}(t, \theta) = \theta^T x(t).$$

Each model in $\mathbf{M}(\theta)$ has a single input and a single output. The coefficient a is 0.4 and the parameter set is selected as $\theta \in \Theta = [-10, 10] \times [-10, 10]$. The system $\mathbf{S} \in \mathbf{M}(\theta)$ has the true parameter $\theta_{\mathtt{true}} = [-0.5, 1]^T$. System $\mathbf{S}$ is a member of models demonstrated by the Laguerre-basis functions [18]. This is a special case of the orthonormal basis functions and can be translated to the aforementioned parameterized state space format. The system is affected by a process noise which is a Gaussian process with covariance matrix $0.5\mathbb{I}_2$, where $\mathbb{I}_2$ is a 2×2 identity matrix. There is also an additive measurement noise with zero-mean and variance 0.5. The input range is considered to be $[-0.2, 0.2]$.

We want to verify with high probability if the output of the system $\mathbf{S}$ remains in $\ell_g = [-0.5, 0.5]$ until it reaches $\ell_y = [-0.1, 0.1]$ at some time in the interval $[2, 4]$. We denote the atomic propositions $\mu_1 = \{y \geq -0.5\}$, $\mu_2 = \{-y \geq -0.5\}$, $\mu_3 = \{y \geq -0.1\}$, $\mu_4 = \{-y \geq -0.1\}$. Our desired property can be written as

$$\mathbb{P}(\mathbf{S} \models (\mu_1 \wedge \mu_2)\, \mathsf{U}_{[2,4]}\, (\mu_3 \wedge \mu_4)) \geq 1 - \delta.$$

We select $\delta = 0.01$. The system starts at the initial condition $x(0) = 0$.

We used the procedure in Section 4 to decompose this STL specification to algebraic constraints on the atomic propositions. Equation (21) is used to improve the conservativeness of the approximation. The feasible set is approximated either using the Monte Carlo method or the piecewise affine approximation described in Section 5. The initial set of parameters can be restricted by finding the extreme values of θ over all constraints as described in Subsection 5.2 which is considered $[-3.5, 3.5]$ for this case study. We select random points which are uniformly distributed in this restricted region in order to compute the confidence value using the Monte Carlo method with the precision 0.000001 in (25). Computed feasible set using the Monte Carlo technique is demonstrated in Fig. 3 with

Figure 3: Contours of $p(\theta \mid \mathcal{D})$ for $\theta_{\texttt{true}} = [-0.5, 1]^T$ after 50 measurements over the feasible set computed by the Monte Carlo and PWA techniques which are represented by red and blue points, respectively.

red-face squares. The feasible set which is recovered with the piecewise affine technique is illustrated in Fig. 3 with blue-edge diamonds. We used linear programming in order to find the feasible set of parameters $(\theta, \mathbf{z})$ for the linearized form of (23) for all time steps. Then, this feasible set is projected into θ space using MPT3 toolbox [22]. We choose the total number of regions in the piecewise affine approximation to be 25.

As we do not have any prior knowledge about the parameters, we choose a uniform distribution $p(\theta)$ on the possible models. Based on the uniform prior, the confidence is computed using (6) as 0.0279 and 0.0258 with Monte Carlo and PWA approximations, respectively. Afterward, we designed an experiment on the system with the true parameter and an input sequence with a uniform distribution over $[-2, 2]$ and measured output for 50 consecutive time instances. Using updated $p(\theta \mid \mathcal{D})$ coming from the measurement data, confidence improved significantly into 0.9099 and 0.8962 for Monte Carlo and PWA, respectively. Contours of the posterior distribution are illustrated in Fig. 3.

We repeated the same experiment 100 times for several other true parameters $\theta_{\texttt{true}}$. For all of these instances, updated posteriori probability in (11), after 50 measurements, is used in order to compute the confidence value according to (6). Results of computing the confidence with Monte Carlo and PWA approximation are shown in Table 1. As it can be seen, for parameters that lie deep inside the feasible set, the confidence value is high with a low variance for both techniques. Meanwhile, for the points near the edges, the variance is higher and confidence value is lower. For points far enough from the feasible set, confidence tends to be very close to zero.

Table1. Means and variances of computed confidence values for 5 different true parameters after 50 measurements.

	Monte Carlo		PWA	
$\theta_{\texttt{true}}$	Mean	Variance	Mean	Variance
$[-0.5, 1]^T$	0.9587	0.0023	0.9514	0.0042
$[3, -1]^T$	0.4902	0.0061	0.5032	0.0062
$[1, 0.5]^T$	0.7932	0.0025	0.7584	0.0053
$[-2, 1.5]^T$	0.9018	0.0009	0.9156	0.0005
$[2, -1]^T$	0.0278	0.0005	0.0480	0.0006

7.2. Verification Case Study: Bounded Support Noise

In this section, we consider the multi-zone model of a building developed in [37]. The model gives the dynamic response of indoor temperatures and humidity for a building depicted in Fig. 4. The state vector is $X_{room} = [\Delta t_{a,s}, \Delta W_{a,s}, \Delta t_{riw,s}, \Delta t_{a,n}, \Delta W_{a,n}, \Delta t_{riw,n}, \Delta t_{rew,n}, \Delta t_{a,r}, \Delta W_{a,r}, \Delta t_{riw,r}]^T$, where its elements are variations in air-supply temperature, air-supply humidity, internal wall temperature (air-supply zone), work zone temperature, work zone humidity, internal wall temperature (work zone), external wall temperature (work zone), air-return temperature, air-return humidity, and internal wall temperature (air-return zone), respectively. The input vector is $[\Delta t_{a,i}, \Delta W_{a,i}, \Delta G_{a,i}, \Delta t_{a,out}, \Delta I_{sol}]$ which its elements correspond to air-supply temperature set-point, air-supply humidity set-point, air flow set-point, return temperature set-point, and solar radiant intensity, respectively.

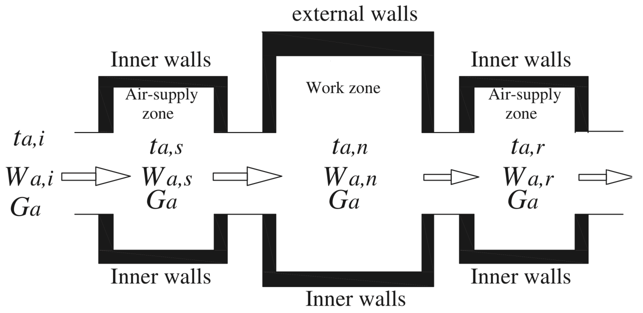


Figure 4: Schematic of the air-conditioned building [37].

All states are affected by a Gaussian process noise with variance of 0.001. We assume the input can change every 100 seconds. Then we discretize the dynamic by $\tau = 100s$. The comfort criterion is defined as a weighted combination of work zone temperature and humidity variations: $\theta_1 \Delta t_{a,n} + \theta_2 \Delta W_{a,n}$ with weights θ_1 and θ_2. This comfort criterion is the output of the system. The measurements of this output is available but affected by a Gaussian noise with variance 0.01. We consider the following STL specification:

$$\mathbb{P}\left(\bigwedge_{t=1}^{5} |\theta_1 \Delta t_{a,n}(t) + \theta_2 \Delta W_{a,n}(t)| \leq \beta\right) \geq 0.9, \tag{33}$$

with $\beta = 1$ in our numerical implementation. We assume that θ_1 and θ_2 are not known but have the true values 1 and 0.5, respectively ($\theta_{\texttt{true}} = [1, 0.5]^T$). The initial parameter space is considered to be $(\theta_1, \theta_2) \in [-2.5, 2.5]^2$. Our goal is to verify whether the above property is satisfied for all inputs $\Delta t_{a,i}$,$\Delta W_{a,i} \in [-1, 1]$. Other inputs are considered to be zero in this case study.

Figure 5: Updated posterior function after 10 measurements over the feasible polyhedron region computed using MPT3 toolbox.

We utilize the approach of Section 6 and limit the supports of process noise to the bounded interval $[-0.1, 0.1]$. This amounts to having $\alpha = 0.0155$ in Proposition 3 and replacing the above chance constraint with

$$\mathbb{P}_{\mathrm{t}}\left(\bigwedge_{t=1}^{5}|\theta_1\Delta t_{a,n}(t)+\theta_2\Delta W_{a,n}(t)|\leq\beta\right)\geq 0.9787.$$

The computed feasible region for this STL specification which is a polyhedron and computed by MPT3 toolbox [22], is demonstrated in Fig. 5 (green region). Since it is assumed that we do not have any prior knowledge about the parameters, a uniform distribution is chosen over the parameter space. The posterior distribution is illustrated in Fig. 5 after collecting 10 measurements and updating the distribution. This approach computes the feasible region and the confidence value in only 55 seconds. We have repeated this experiment 100 times and computed the confidence values using (6). The mean and variance of the confidence values are respectively 0.8607 and 0.0012.

If we directly apply the approach of Section 5 to the constraint (33) and the unbounded support noise, we have to use the methods in Subsection 5.2 in order to approximately compute the confidence value, which is computationally more expensive. We computed the confidence value using Monte Carlo integration with 6.25×10^6 samples from the parameter space, which gives the interval $[0.8505, 0.8705]$ for the confidence with probability 0.99 over the sampled parameters. This interval is close to the confidence value obtained using truncation but the computation time is 19 minutes on an iMac (3.5 GHz Intel Core i7 processor) which is much larger than 55 seconds taken based on truncation.

8. Conclusion and Future Works

In this work, we considered parametric linear time-invariant (LTI) systems. We developed a scheme for providing a confidence value for the satisfaction of STL specifications for such systems by incorporating both model-based and Bayesian inference techniques. Using our approach, one can transform the probabilistic STL specification over the states of the system into a set of algebraic inequalities. Solving these inequalities for the whole range of inputs results in the feasible set of parameters. By leveraging the collected data from the system, the probability density of the unknown parameters is updated and the confidence value is computed over the feasible domain of the parameters. While this paper is focused on verification of parametric LTI systems, in the future we plan to address synthesis problem using maximum likelihood methods to find inputs that maximize the probability of satisfying a given specification. Another interesting research direction is to study robustness of the computations with respect to uncertainties in the distribution of random variables affecting the evolution of the system.

References

[1] Alessandro Abate. Formal verification of complex systems: model-based and data-driven methods. In Proceedings of the 15th ACM-IEEE International Conference on Formal Methods and Models for System Design, pages 91–93. ACM, 2017.

[2] Christel Baier and Joost-Pieter Katoen. Principles of model checking. MIT press, 2008.

[3] Ezio Bartocci, Luca Bortolussi, and Guido Sanguinetti. Data-driven statistical learning of temporal logic properties. In International Conference on Formal Modeling and Analysis of Timed Systems, pages 23–37. Springer, 2014.

[4] Dirk Beyer, Matthias Dangl, and Philipp Wendler. A unifying view on SMT-based software verification. Journal of Automated Reasoning, 60(3):299–335, 2018.

[5] Dirk Beyer and M Erkan Keremoglu. CPAchecker: A tool for configurable software verification. In International Conference on Computer Aided Verification, pages 184–190. Springer, 2011.

[6] Sergiy Bogomolov, Christian Schilling, Ezio Bartocci, Gregory Batt, Hui Kong, and Radu Grosu. Abstraction-based parameter synthesis for multiaffine systems. In Haifa Verification Conference, pages 19–35. Springer, 2015.

[7] Olivier Bouissou, Eric Goubault, Sylvie Putot, Aleksandar Chakarov, and Sriram Sankaranarayanan. Uncertainty propagation using probabilistic affine forms and concentration of measure inequalities. In International Conference on Tools and Algorithms for the Construction and Analysis of Systems, pages 225–243. Springer, 2016.

[8] Ya-Chien Chang, Nima Roohi, and Sicun Gao. Neural Lyapunov control. In Advances in Neural Information Processing Systems, pages 3240–3249, 2019.

[9] Yi Chou and Sriram Sankaranarayanan. Bayesian parameter estimation for nonlinear dynamics using sensitivity analysis. In 28th International Joint Conference on Artificial Intelligence, pages 5708–5714. AAAI Press, 2019.

[10] Edmund M Clarke and Paolo Zuliani. Statistical model checking for cyber-physical systems. In International Symposium on Automated Technology for Verification and Analysis, pages 1–12. Springer, 2011.

[11] Patricia Derler, Edward A Lee, and Alberto Sangiovanni Vincentelli. Modeling cyber–physical systems. Proceedings of the IEEE, 100(1):13–28, 2011.

[12] Georgios E Fainekos and George J Pappas. Robustness of temporal logic specifications. In Formal Approaches to Software Testing and Runtime Verification, pages 178–192. Springer, 2006.

[13] Samira S Farahani, Rupak Majumdar, Vinayak S Prabhu, and Sadegh Soudjani. Shrinking horizon model predictive control with signal temporal logic constraints under stochastic disturbances. IEEE Transactions on Automatic Control, 2018.

[14] Samira S. Farahani, Sadegh Soudjani, Rupak Majumdar, and Carlos Ocampo-Martinez. Robust model predictive control with signal temporal logic constraints for Barcelona wastewater system. IFAC-PapersOnLine, 50(1):6594 – 6600, 2017. 20th IFAC World Congress.

[15] Samira S Farahani, Sadegh Soudjani, Rupak Majumdar, and Carlos Ocampo-Martinez. Formal controller synthesis for wastewater systems with signal temporal logic constraints: The Barcelona case study. Journal of Process Control, 69:179–191, 2018.

[16] Angelos Georghiou, Angelos Tsoukalas, and Wolfram Wiesemann. Robust dual dynamic programming. Operations Research, 2019.

[17] S Haesaert, PMJ van den Hof, and A Abate. Data-driven and model-based verification via Bayesian identification and reachability analysis. Automatica, 79:115–126, 2017.

[18] Sofie Haesaert, Paul MJ Van den Hof, and Alessandro Abate. Data-driven property verification of grey-box systems by Bayesian experiment design. In 2015 American Control Conference (ACC), pages 1800–1805. IEEE, 2015.

[19] Sofie Haesaert, Paul MJ Van den Hof, and Alessandro Abate. Automated experiment design for data-efficient verification of parametric Markov decision processes. In International Conference on Quantitative Evaluation of Systems, pages 259–274. Springer, 2017.

[20] EM Hahn, M Perez, S Schewe, F Somenzi, A Trivedi, and D Wojtczak. Omega-regular objectives in model-free reinforcement learning (2018). arXiv:1810.00950, 340.

[21] Mohammadhosein Hasanbeig, Alessandro Abate, and Daniel Kroening. Logically-constrained neural fitted Q-iteration. In Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems, pages 2012–2014. International Foundation for Autonomous Agents and Multiagent Systems, 2019.

[22] Martin Herceg, Michal Kvasnica, Colin N Jones, and Manfred Morari. Multi-Parametric Toolbox 3.0. In 2013 European Control Conference (ECC), pages 502–510. IEEE, 2013.

[23] Svante Janson. Large deviations for sums of partly dependent random variables. Random Structures & Algorithms, 24(3):234–248, 2004.

[24] M. Kazemi and Sadegh Soudjani. Formal policy synthesis for continuous-space systems via reinforcement learning. In IFM, 2020.

[25] Abolfazl Lavaei, Fabio Somenzi, Sadegh Soudjani, Ashutosh Trivedi, and Majid Zamani. Formal controller synthesis for continuous-space MDPs via model-free reinforcement learning. in ACM/IEEE 11th International Conference on Cyber-Physical Systems (ICCPS), 2020.

[26] Oded Maler and Dejan Nickovic. Monitoring temporal properties of continuous signals. In Formal Techniques, Modelling and Analysis of Timed and Fault-Tolerant Systems, pages 152–166. Springer, 2004.

[27] Sriram Pemmaraju and Steven Skiena. Computational Discrete Mathematics: Combinatorics and Graph Theory with Mathematica®. Cambridge university press, 2003.

[28] Vasumathi Raman, Alexandre Donzé, Dorsa Sadigh, Richard M Murray, and Sanjit A Seshia. Reactive synthesis from signal temporal logic specifications. In Proceedings of the 18th International Conference on Hybrid Systems: Computation and Control, pages 239–248. ACM, 2015.

[29] Hadi Ravanbakhsh and Sriram Sankaranarayanan. Learning control Lyapunov functions from counterexamples and demonstrations. Autonomous Robots, 43(2):275–307, 2019.

[30] Dorsa Sadigh and Ashish Kapoor. Safe control under uncertainty with probabilistic signal temporal logic. 2016.

[31] Sadra Sadraddini and Calin Belta. Formal guarantees in data-driven model identification and control synthesis. In Proceedings of the 21st International Conference on Hybrid Systems: Computation and Control, pages 147–156, 2018.

[32] Ali Salamti, Sadegh Soudjani, and Zamani. Data-driven verification under signal temporal logic constraints. in 21th IFAC World Congress, 2020.

[33] Koushik Sen, Mahesh Viswanathan, and Gul Agha. Statistical model checking of black-box probabilistic systems. In International Conference on Computer Aided Verification, pages 202–215. Springer, 2004.

[34] Koushik Sen, Mahesh Viswanathan, and Gul Agha. On statistical model checking of stochastic systems. In International Conference on Computer Aided Verification, pages 266–280. Springer, 2005.

[35] F. Shmarov, S. Soudjani, N. Paoletti, E. Bartocci, S. Lin, S. A. Smolka, and P. Zuliani. Automated synthesis of safe digital controllers for sampled-data stochastic nonlinear systems. IEEE Access, 8:180825–180843, 2020.

[36] Sadegh Soudjani, Rupak Majumdar, and Tigran Nagapetyan. Multilevel Monte Carlo method for statistical model checking of hybrid systems. In Quantitative Evaluation of Systems, pages 351–367, Cham, 2017. Springer International Publishing.

[37] Ye Yao, Kun Yang, Mengwei Huang, and Liangzhu Wang. A state-space model for dynamic response of indoor air temperature and humidity. Building and Environment, 64:26–37, 2013.

[38] Håkan LS Younes, Marta Kwiatkowska, Gethin Norman, and David Parker. Numerical vs. statistical probabilistic model checking. International Journal on Software Tools for Technology Transfer, 8(3):216–228, 2006.