Robust Conformal Prediction for STL Runtime Verification under Distribution Shift

Yiqi Zhao[1], Bardh Hoxha[2], Georgios Fainekos[2], Jyotirmoy V. Deshmukh[1], and Lars Lindemann[1]

[1]Thomas Lord Department of Computer Science, University of Southern California
[2]Toyota NA R&D

***Abstract*—Cyber-physical systems (CPS) designed in simulators behave differently in the real-world. Once they are deployed in the real-world, we would hence like to predict system failures during runtime. We propose robust predictive runtime verification (RPRV) algorithms under signal temporal logic (STL) tasks for general stochastic CPS. The RPRV problem faces several challenges: (1) there may not be sufficient data of the behavior of the deployed CPS, (2) predictive models are based on a distribution over system trajectories encountered during the design phase, i.e., there may be a distribution shift during deployment. To address these challenges, we assume to know an upper bound on the statistical distance (in terms of an f-divergence) between the distributions at deployment and design time, and we utilize techniques based on *robust conformal prediction*. Motivated by our results in [1], we construct an accurate and an interpretable RPRV algorithm. We use a trajectory prediction model to estimate the system behavior at runtime and robust conformal prediction to obtain probabilistic guarantees by accounting for distribution shifts. We precisely quantify the relationship between calibration data, desired confidence, and permissible distribution shift. To the best of our knowledge, these are the first statistically valid algorithms under distribution shift in this setting. We empirically validate our algorithms on a Franka manipulator within the NVIDIA Isaac sim environment.**

***Index Terms*—Predictive runtime verification, stochastic system verification, signal temporal logic, conformal prediction.**

I. Introduction

Cyber-physical Systems (CPS) operate in highly unpredictable environments and often have intrinsic and extrinsic sources of stochasticity that affect their dynamic behavior. Many such CPSs can be modeled simply as a distribution $\mathcal{D}$ over the space of system trajectories. In this paper, we are interested in the predictive runtime verification of such a stochastic CPS against a specification ϕ expressed in signal temporal logic (STL). In other words, during the operation of the system we would like to compute the probability that the system will satisfy (or violate) the specification ϕ using the already observed part of the system trajectory. System verification using formal languages is of great interest for the safe design of many CPS applications, e.g., the responsibility sensitive safety model can be encoded in STL for verification of safe behaviors in autonomous vehicles [2].

The use of high-fidelity models and simulators has resulted in statistical and data-driven paradigms for the design, analysis, and test of many CPS. While previous approaches have formulated algorithms for predictive runtime monitoring of CPSs [1], [3], the guarantees that these approaches provide depend on the underlying distribution being the same at test and design time. However, when a CPS is deployed in the real-world, the underlying distribution may be different – a phenomenon called a *distribution shift*. Our focus is thus on designing *robust predictive runtime verification (RPRV) algorithms* that provide valid verification results even when the test distribution $\mathcal{D}$ is different from the training distribution $\mathcal{D}_0$.

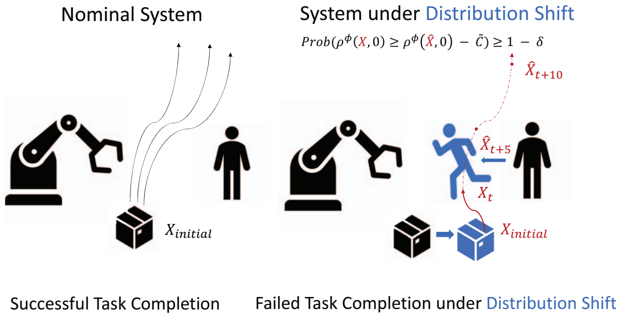

Nominal System
System under Distribution Shift
$Prob(\rho^{\phi}(X,0) \geq \rho^{\phi}(\hat{X},0) - \tilde{C}) \geq 1 - \delta$
$\hat{X}_{t+10}$
$\hat{X}_{t+5}$
X_t
$X_{initial}$
$X_{initial}$
Successful Task Completion
Failed Task Completion under Distribution Shift

Fig. 1: Left: A robotic arm manipulating a box in the presence of a human. Right: The robotic arm performing the same task when the human behavior and the package location changed. We propose predictive robust runtime verification algorithms to verify systems under such distribution shift.

Specifically, we propose RPRV algorithms that account for all test distributions $\mathcal{D}$ that are contained within a set of possible test distributions $P(\mathcal{D}_0)$. The set $P(\mathcal{D}_0)$ here denotes all distributions that are close to the training distribution $\mathcal{D}_0$ under a suitable distance measure. For instance, $P(\mathcal{D}_0)$ may denote those distributions $\mathcal{D}$ whose f-divergence with respect to $\mathcal{D}_0$ is bounded by $\epsilon > 0$, i.e., $P(\mathcal{D}_0)$ may be defined as $\{\mathcal{D} | D_f(\mathcal{D}, \mathcal{D}_0) \leq \epsilon\}$ where D_f is an f-divergence measure. RPRV algorithms are needed since test distributions are usually different from the training distribution. Examples include varying conditions in the environment that the system operates in, e.g., weather conditions or traffic, or also when $\mathcal{D}_0$ models a high-fidelity simulator. To the best of our knowledge, existing predictive runtime verification algorithms are not robust to distribution shift. Furthermore, in this paper, we deal with the challenge of increasing data requirements for larger distribution shifts. We make the following contributions.

- We present an accurate and an interpretable RPRV algorithms that i) use trajectory predictors to predict future

979-8-3503-6927-4/24/$31.00 ©2024 IEEE
DOI 10.1109/ICCPS61052.2024.00022

system behavior, and ii) leverage robust conformal prediction [4] to quantify prediction uncertainty using calibration data from $\mathcal{D}_0$. The algorithms are valid with a user-defined probability for all test distributions $\mathcal{D} \in P(\mathcal{D}_0)$.

- We analyze data requirements and the interplay between confidence and permissible distribution shift. We make various algorithmic advancements of the interpretable method, originally presented in [1], to improve accuracy and data efficiency for increasing distribution shifts.
- We empirically validate our algorithms on a stylized running example and a Franka manipulator within NVIDIA Isaac sim. We illustrate the efficacy of our robust algorithms compared to a non-robust version from [1].

A. Related Work.

(Offline) Verification of Stochastic Systems. Formal verification of general CPS models is known to be an undecidable problem [5]. To address the computational cost and inefficiency of abstraction-based verification algorithms, techniques such as statistical model checking have been used to give statistical guarantees [6]–[8]. In [9], statistical hypothesis testing is performed with a set of executions from a black-box system to verify continuous stochastic logic specifications. Statistical verification under STL specifications was first considered in [10], [11]. The works in [8], [12] consider the problem of statistical verification of learning-enabled CPS with respect to STL specifications. Recent work has also sought to combine model-based techniques and statistical, data-driven techniques [13], [14]. For instance, in [15] surrogate Gaussian process models of the system are learned and used to to obtain probabilistic guarantees on satisfying STL specifications. These works assume that the distribution from which data is sampled is fixed. The authors in [16] propose an active sampling approach using imprecise neural networks to promote distributional robustness in statistical verification of neural networks. In another direction, recent frameworks train generative models that capture distribution shifts [17], [18].

Runtime Verification of Stochastic Systems. In runtime verification (RV), we are instead interested in verifying system properties during the operation of the system solely based on the currently observed system trajectory [19]–[22]. RV techniques complement offline verification techniques, and there is a growing body of literature on RV approaches for STL specifications [23]–[26]. Predictive RV is a special class of RV where we use a system model to predict the future system behavior from the currently observed system trajectory to either check if (hidden) system states satisfy a given specification [27], [28] or if the system may violate system specifications in the future [29]–[34]. In our prior work [1], we presented two predictive RV algorithms that use conformal prediction, a statistical tool for uncertainty quantification [35], by calibrating prediction errors of a trajectory predictor to obtain valid probabilistic verification guarantees on the satisfaction of STL specifications. Similar in spirit, the authors in [3] used a technique known as conformalized quantile regression [36] to design predictive RV algorithms that also provide probabilistic verification guarantees on the satisfaction of STL specifications. While statistical guarantees are provided in [1], [3], these guarantees are not valid when the trajectory distribution of the deployed system deviates from the distribution of the design-time system. The authors in [37] take a first step in this direction with conformal prediction by proposing robust evaluators that minimize distribution shifts in high-dimensional measurements. However, to the best of our knowledge, no existing work provides statistically valid RV guarantees under distribution shift as we do in this paper.

II. Problem Formulation

To describe stochastic systems, we consider an unknown test distribution $\mathcal{D}$ over system trajectories $X := (X_0, X_1 \ldots) \sim \mathcal{D}$ where $X_\tau \in \mathbb{R}^n$ is the state of the system at time τ. We make no assumption about $\mathcal{D}$, e.g., $\mathcal{D}$ can describe Markov decision processes or hybrid stochastic systems. While the distribution $\mathcal{D}$ is completely unknown, we assume that we have access to K calibration trajectories $(X^{(1)}, \ldots, X^{(K)})$ from a training distribution $\mathcal{D}_0$ that is close to $\mathcal{D}$ (as specified later).[1]

Assumption 1. *We have access to a dataset* $S := (X^{(1)}, \ldots, X^{(K)})$ *in which each of the K trajectories* $X^{(i)} := (X_0^{(i)}, X_1^{(i)}, \ldots)$ *is independently drawn from a training distribution* $\mathcal{D}_0$*, i.e.,* $X^{(i)} \sim \mathcal{D}_0$.[2]

Assumption 1 holds in many applications. An example would be a setting in which $\mathcal{D}_0$ describes the motion of a robot within a high-fidelity simulator, while $\mathcal{D}$ describes a real robot operating in a lab environment. To measure closeness of the distributions $\mathcal{D}_0$ and $\mathcal{D}$, we use the f-divergence, a statistical distance, that quantifies the similarity between $\mathcal{D}_0$ and $\mathcal{D}$ and thereby the distribution shift. Specifically, the f-divergence $D_f(\mathcal{D}, \mathcal{D}_0)$ is defined as

$$D_f(\mathcal{D}, \mathcal{D}_0) := \int_{\mathcal{X}} f\Big(\frac{d\mathcal{D}}{d\mathcal{D}_0}\Big) d\mathcal{D}_0$$

where $\mathcal{X}$ is the support of $\mathcal{D}_0$, $\mathcal{D}$ is absolutely continuous with respect to $\mathcal{D}_0$, and $\frac{d\mathcal{D}}{d\mathcal{D}_0}$ is the Radon-Nikodym derivative of $\mathcal{D}$ with respect to $\mathcal{D}_0$. The function $f : [0, \infty) \to \mathbb{R}$ is convex and satisfies $f(1) = 0$. If we set $f(z) := \frac{1}{2}|z - 1|$, we obtain the total variation distance $TV(\mathcal{D}, \mathcal{D}_0) := \frac{1}{2}\int_x |P(x) - Q(x)| dx$ where P and Q are probability density functions to $\mathcal{D}$ and $\mathcal{D}_0$.

Assumption 2. *The test and training distributions* $\mathcal{D}$ *and* $\mathcal{D}_0$ *are such that* $D_f(\mathcal{D}, \mathcal{D}_0) \leq \epsilon$ *where* $\epsilon > 0$. *We hence assume that* $\mathcal{D} \in P(\mathcal{D}_0) := \{\mathcal{D}' \mid D_f(\mathcal{D}', \mathcal{D}_0) \leq \epsilon\}$.

We emphasize that the parameter ϵ is a measure of the permissible distribution shift in terms of the f-divergence D_f.

Challenges in Runtime Verification. Given a specification ϕ

[1] The distributions $\mathcal{D}$ and $\mathcal{D}_0$ are defined over the same probability space $(\Omega, \mathcal{F}, \mathbb{P})$ where Ω is the sample space, $\mathcal{F}$ is a σ-algebra of Ω, and $\mathbb{P} : \mathcal{F} \to [0, 1]$ is a probability measure. For simplicity, we will mostly use the notation Prob to be independent of the underlying probability space.

[2] For instance, we can obtain such i.i.d. trajectories from a simulator that we can query repeatedly with a fixed distribution over simulation parameters.

and a partial observation $(X_0, \ldots, X_t)$ from the test trajectory $X \sim \mathcal{D}$ at runtime t, we are interested in computing the probability that X satisfies ϕ. The challenges are that we only have knowledge about the training distribution $\mathcal{D}_0$ as per Assumption 1, and our knowledge about the test distribution $\mathcal{D}$ is limited that $\mathcal{D}_0$ and $\mathcal{D}$ are ϵ-close. We design RPRV algorithms that are predictive in the sense that we use predictions $\hat{X}_{\tau|t}$ of future states X_τ for $\tau > t$ and robust as we provide valid probabilistic guarantees as long as $\mathcal{D} \in P(\mathcal{D}_0)$.

A. Signal Temporal Logic

We use signal temporal logic (STL) to express system specifications and define STL over discrete-time trajectories $x := (x_0, x_1, \ldots)$, e.g., x can be a realization of the stochastic trajectory X. We note that readers with limited background in temporal logics can, if they like to, skip the following formal definitions of syntax and semantics of an STL formula ϕ and instead think of ϕ as a high-level system specification that is imposed on the system at time τ_0. We let $(x, \tau_0) \models \phi$ indicate that x satisfies ϕ and we assume that bounded trajectories x of length L^ϕ are sufficient to compute $(x, \tau_0) \models \phi$. The notation $\rho^\phi(x, \tau_0) \in \mathbb{R}$ will indicate how well ϕ is satisfied by x at time τ_0 with larger values indicating better satisfaction.

Formally, the atomic elements of STL are predicates that are functions $\pi : \mathbb{R}^n \to \{\text{True}, \text{False}\}$. The predicate π is defined via a predicate function $h : \mathbb{R}^n \to \mathbb{R}$ as $\pi(x_\tau) := \text{True}$ if $h(x_\tau) \geq 0$ and $\pi(x_\tau) := \text{False}$ otherwise. The syntax of STL is recursively defined as

$$\phi \ ::= \ \text{True} \mid \pi \mid \neg\phi' \mid \phi' \wedge \phi'' \mid \phi' U_I \phi'' \tag{1}$$

where ϕ' and ϕ'' are STL formulas. The Boolean operators $\neg$ and $\wedge$ encode negations ("not") and conjunctions ("and"), respectively. The until operator $\phi' U_I \phi''$ encodes that ϕ' has to be true from now on until ϕ'' becomes true at some future time within the time interval $I \subseteq \mathbb{R}_{\geq 0}$. We can further derive the operators for disjunction ($\phi' \vee \phi'' := \neg(\neg\phi' \wedge \neg\phi'')$), eventually ($F_I \phi := \top U_I \phi$), and always ($G_I \phi := \neg F_I \neg \phi$).

To determine if a trajectory x satisfies an STL formula ϕ that is enabled at time τ_0, we can define the semantics as a relation $\models$, i.e., $(x, \tau_0) \models \phi$ means that ϕ is satisfied. While the STL semantics are fairly standard [38], we recall them in Appendix A. Additionally, we can define robust semantics $\rho^\phi(x, \tau_0) \in \mathbb{R}$ that indicate how robustly the formula ϕ is satisfied or violated [39], [40], see Appendix A. Larger and positive values of $\rho^\phi(x, \tau_0)$ hence indicate that the specification is satisfied more robustly. Importantly, it holds that $(x, \tau_0) \models \phi$ if $\rho^\phi(x, \tau_0) > 0$ due to [40, Proposition 16]. We make the following assumption on the class of STL formulas in this paper.

Assumption 3. *We consider bounded STL formulas ϕ, i.e., all time intervals I within the formula ϕ are bounded.*

Satisfaction of bounded STL formulas can be decided by finite length trajectories [41]. The minimum length is given by the formula length L^ϕ, i.e., with knowledge of $(x_{\tau_0}, \ldots, x_{\tau_0 + L^\phi})$ we can compute $(x, \tau_0) \models \phi$, see again Appendix A for more details.

B. Robust Predictive Runtime Verification

Assume that we have observed the states $X_{\text{obs}} := (X_0, \ldots, X_t)$ at runtime t, i.e., all states up until time t are known, while future states $X_{\text{un}} := (X_{t+1}, X_{t+2}, \ldots)$ from $X = (X_{\text{obs}}, X_{\text{un}}) \sim \mathcal{D}$ are not known yet. In this paper, we are interested in solving the following problem.

Problem 1. *Let $\mathcal{D}_0$ be a training distribution, S be a calibration dataset from $\mathcal{D}_0$ that satisfies Assumption 1, $\mathcal{D}$ be a test distribution from $P(\mathcal{D}_0)$ that satisfies Assumption 2, and ϕ be an STL formula imposed at time τ_0 that satisfies Assumption 3. Given the current time t, observations X_{obs} from $X \sim \mathcal{D}$, and a failure probability $\delta \in (0, 1)$, compute a lower bound ρ^* such that $Prob(\rho^\phi(X, \tau_0) \geq \rho^*) \geq 1 - \delta$.*

Once we have computed the lower bound ρ^*, we remark that $\text{Prob}\big((X, \tau_0) \models \phi\big) \geq 1 - \delta$ if $\rho^* > 0$, see [40, Proposition 16]. We next introduce a running example that we use throughout this paper to explain and illustrate our algorithms.

Example 1. *We consider the F-16 aircraft from [42] with a hybrid controller modelled by 16 states. We only consider the height h (given in ft) as the state to verify the specification $\phi := G_{[0,105]} h \geq 60$ and to find ρ^* from Problem 1 for $\delta := 0.2$. To construct a simple academic example, we collect a single trajectory x_c from the simulator and then add independent noise to x_c at each time, i.e., we let $\mathcal{N}(x_c(t), 3^2)$ and $\mathcal{N}(x_c(t), 3.5^2)$ describe $\mathcal{D}_0$ and $\mathcal{D}$, respectively. We assume that we have $K := 2000$ calibration trajectories $X^{(i)}$ with $i \in \{1, \ldots, K\}$ from $\mathcal{D}_0$ as per Assumption 1.*

III. Robust Conformal Prediction

Our solution to Problem 1 will rely on trajectory predictors that predict future system states X_{un} from observations X_{obs}. To quantify the accuracy of these predictions, we use the calibration dataset S from $\mathcal{D}_0$ along with robust conformal prediction as presented in [4] to account for the distribution shift between $\mathcal{D}_0$ and $\mathcal{D}$. Robust conformal prediction is an extension of conformal prediction which is a statistical tool for uncertainty quantification [35], [43]–[45].

Conformal Prediction (CP). Let $R^{(0)}, \ldots, R^{(K)} \sim \mathcal{R}_0$ be $K+1$ independent and identically distributed random variables following a training distribution $\mathcal{R}_0$.[3] The variable $R^{(i)}$ can be freely defined and is referred to as the nonconformity score. In regression, a common choice for $R^{(i)}$ is the prediction error $|Z^{(i)} - \mu(U^{(i)})|$ where the predictor μ attempts to predict $Z^{(i)}$ based on an input $U^{(i)}$. We note that a large nonconformity score indicates a large prediction error. Our goal is thus to obtain an upper bound for $R^{(0)}$ (our test data) from $R^{(1)}, \ldots, R^{(K)}$ (our calibration data). Formally, given a failure probability $\delta \in (0, 1)$, we want to compute a constant C (which depends on $R^{(1)}, \ldots, R^{(K)}$) such that

$$\text{Prob}(R^{(0)} \leq C) \geq 1 - \delta.$$

The probability $\text{Prob}(\cdot)$ is defined over the product measure

[3] In fact, exchangeability of $R^{(0)}, \ldots, R^{(K)}$ would be sufficient which is a weaker requirement than being independent and identically distributed.

of $R^{(0)},\ldots,R^{(K)}$. By a simple statistical argument, one can obtain C to be the $1/K$ corrected $(1-\delta)$th quantile of the empirical distribution of the values $R^{(1)},\ldots,R^{(K)}$, i.e.,

$$C := \text{Quantile}_{(1+1/K)(1-\delta)}(R^{(1)},\ldots,R^{(K)}). \tag{2}$$

Formally, for $\beta\in[0,1]$, the quantile function is defined as

$$\text{Quantile}_{\beta}(R^{(1)},\ldots,R^{(K)}) := \inf\{z\in\mathbb{R}|\text{Prob}(Z\le z)\ge\beta\}$$

where the random variable $Z := \sum_i \delta_{R^{(i)}}/K$ where $\delta_{R^{(i)}}$ is a dirac distribution centered at $R^{(i)}$. Equation (2) thus requires $0 \le (1+1/K)(1-\delta) \le 1$ and imposes the implicit lower bound $(K+1)(1-\delta)\le K$ on the number of data K. If this bound is satisfied and $R^{(1)},\ldots,R^{(K)}$ are sorted in non-decreasing order, we obtain $C := R^{(p)}$ with $p := \lceil (K+1)(1-\delta)\rceil$, i.e., C is the pth smallest nonconformity score. We remark that we trivially have $C := \infty$ if $(K+1)(1-\delta) > K$.

Robust CP. In this paper, our test data is different from the training data. We thus use a robust version of conformal prediction based on [4]. Assume that $R^{(0)},\ldots,R^{(K)}$ are again independent, but not identically distributed in the sense that $R^{(0)}\sim\mathcal{R}$ while $R^{(1)},\ldots,R^{(K)}\sim\mathcal{R}_0$ where $\mathcal{R}$ is a test distribution. Under the assumption that test and training distributions $\mathcal{R}$ and $\mathcal{R}_0$ are close, the calibration data from the training distribution can still be used to bound $R^{(0)}$.

Lemma 1 (Corollary 2.2 in [4])**.** *Let $R^{(0)},\ldots,R^{(K)}$ be independent random variables with $R^{(0)}\sim\mathcal{R}$ and $R^{(1)},\ldots,R^{(K)}\sim\mathcal{R}_0$ where the distributions $\mathcal{R}$ and $\mathcal{R}_0$ are such that $D_f(\mathcal{R},\mathcal{R}_0)\le\epsilon$. For a failure probability $\delta\in(0,1)$, it holds that*

$$\mathit{Prob}(R^{(0)}\le\tilde{C})\ge 1-\delta \tag{3}$$

where the constant $\tilde{C}$ is computed as

$$\tilde{C} := \mathit{Quantile}_{1-\tilde{\delta}}(R^{(1)},\ldots,R^{(K)}) \tag{4}$$

where the adjusted confidence level $\tilde{\delta}$ is defined as

$$\tilde{\delta} := 1-g^{-1}(1-\delta_n)$$

and obtained by solving a series of convex optimization problems as

$$\begin{aligned}
\delta_n &:= 1-g\big((1+1/K)g^{-1}(1-\delta)\big),\\
g(\beta) &:= \inf\{z\in[0,1]|\beta f(\frac{z}{\beta})+(1-\beta)f(\frac{1-z}{1-\beta})\le\epsilon\},\\
g^{-1}(\tau) &:= \sup\{\beta\in[0,1]|g(\beta)\le\tau\}.
\end{aligned}$$

We remark that equation (4) requires $0\le 1-\tilde{\delta}\le 1$ and poses restrictions on the number of data K, the failure probability δ, and the distribution shift ϵ as we elaborate on later in the paper. Note further that g and g^{-1} are both solutions to convex programs. The solution to $g^{-1}(\tau)$ can thus be computed efficiently, e.g., using line search over $\beta\in(0,1)$. In some special cases, we can even obtain closed-form solutions for g, e.g., for $f(r) := \frac{1}{2}|r-1|$, associated with the total variation distance, we obtain $g(\beta)=\max(0,\beta-\epsilon)$.

Induced Distribution Shift. Note that in runtime verification we assume an ϵ-bounded distribution shift in terms of the f-divergence on the trajectory level, as described by $\mathcal{D}$ and $\mathcal{D}_0$. In both runtime verification algorithms, it will be necessary to quantify the induced distribution shift of functions that are defined over $X\sim\mathcal{D}$ and $X_0\sim\mathcal{D}_0$. The following result follows trivially by the the data processing inequality.

Lemma 2 (Data processing inequality)**.** *Let $\mathcal{D}$ and $\mathcal{D}_0$ be distributions such that $D_f(\mathcal{D},\mathcal{D}_0)\le\epsilon$ and let $R:\mathcal{X}\to\mathbb{R}$ be a measurable function. For $X\sim\mathcal{D}$ and $X_0\sim\mathcal{D}_0$, let $\mathcal{R}$ and $\mathcal{R}_0$ denote the induced distributions of $R(X)$ and $R(X_0)$, respectively. Then, it holds that*

$$D_f(\mathcal{D},\mathcal{D}_0)\le\epsilon \quad\Rightarrow\quad D_f(\mathcal{R},\mathcal{R}_0)\le\epsilon.$$

IV. Predictive Runtime Verification under Distribution Shift

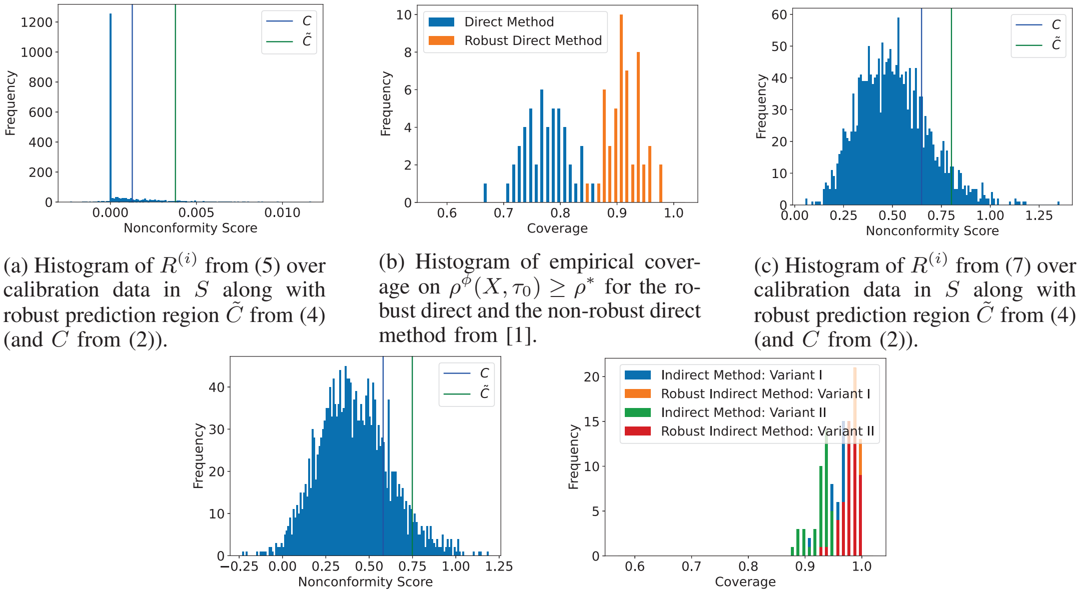
We propose two robust predictive runtime verification algorithms that account for the test distribution $\mathcal{D}$ being different from the training distribution $\mathcal{D}_0$. Our algorithms are inspired by our prior work in [1], but are able to deal with bounded distribution shifts $D_f(\mathcal{D},\mathcal{D}_0)\le\epsilon$. The first algorithm directly uses robust conformal prediction from Lemma 1 to obtain a probabilistic lower bound ρ^* for the robust semantics $\rho^\phi(X,\tau_0)$. This direct algorithm will provide a non-conservative verification result, but is lacking interpretability, i.e., if ϕ is violated no explanation is provided. The second algorithm therefore uses robust conformal prediction from Lemma 1 to obtain a probabilistic lower bound for the robust semantics $\rho^\pi(X,\tau)$ of each predicate π at each time τ, which are subsequently used to obtain a probabilistic lower bound ρ^* for $\rho^\phi(X,\tau_0)$. Besides being able to deal with distribution shifts, we remark that the indirect algorithm that we propose has significantly less conservatism compared to [1].

Trajectory Predictor. Our algorithms use trajectory predictors μ that map observations X_{obs} at time t into predictions $\hat{X}_{t+1|t},\hat{X}_{t+2|t}\ldots$ of future states X_{un}. Therefore, we train a trajectory predictor μ on an additional training dataset from $\mathcal{D}_0$ that is independent of the calibration data in S. Commonly used trajectory predictors range from recurrent neural networks (RNN) and long short-term memory (LSTM) networks [46], [47] to support vector machines [48], [49] and autoregressive integrated moving average models [50], [51].

Since we consider bounded STL formulas ϕ as per Assumption 3, only a finite prediction horizon is needed. Therefore, we define $H := \tau_0 + L^\phi - t$ as the prediction horizon needed for the computation of satisfaction of ϕ imposed at τ_0. To facilitate our discussion, we define the predicted trajectory

$$\hat{X} := (X_{\text{obs}},\hat{X}_{t+1|t},\ldots,\hat{X}_{t+H|t})$$

with predictions $(\hat{X}_{t+1|t},\ldots,\hat{X}_{t+H|t}) := \mu(X_{\text{obs}})$. We use the same notation for trajectories $X^{(i)} := (X^{(i)}_{\text{obs}},X^{(i)}_{\text{un}})$ from the calibration dataset S. Formally, we also define the predicted calibration trajectory $\hat{X}^{(i)} := (X^{(i)}_{\text{obs}},\hat{X}^{(i)}_{t+1|t},\ldots,\hat{X}^{(i)}_{t+H|t})$ where $(\hat{X}^{(i)}_{t+1|t},\ldots,\hat{X}^{(i)}_{t+H|t}) := \mu(X^{(i)}_{\text{obs}})$.



(a) Histogram of $R^{(i)}$ from (5) over calibration data in S along with robust prediction region $\tilde{C}$ from (4) (and C from (2)).

(b) Histogram of empirical coverage on $\rho^{\phi}(X, \tau_0) \geq \rho^*$ for the robust direct and the non-robust direct method from [1].

(c) Histogram of $R^{(i)}$ from (7) over calibration data in S along with robust prediction region $\tilde{C}$ from (4) (and C from (2)).

(d) Histogram of $R^{(i)}$ from (10) over calibration data in S along with robust prediction region $\tilde{C}$ from (4) (and C from (2)).

(e) Histogram of empirical coverage on $\rho^{\phi}(X, \tau_0) \geq \rho^*$ with C for the robust indirect methods (Variant I and II).

Fig. 2: Running example: Histograms of nonconformity scores (5), (7), and (10), and empirical coverage plots of $\rho^{\phi}(X, \tau_0) \geq \rho^*$ for the direct and indirect (Variant I and II) methods.

A. Direct Robust Predictive Runtime Verification

We first apply robust conformal prediction directly to the robust semantics ρ^{ϕ}. To do so, we need to account for prediction errors in $\hat{X}$ and the distribution shift between the calibration trajectories $X^{(i)} \sim \mathcal{D}_0$ and the test trajectory $X \sim \mathcal{D}$. Specifically, consider the nonconformity score

$$R^{(i)} = \rho^{\phi}(\hat{X}^{(i)}, \tau_0) - \rho^{\phi}(X^{(i)}, \tau_0) \tag{5}$$

for each calibration trajectory $X^{(i)} \in S$. Intuitively, this nonconformity score measures the difference between the predicted robust semantics $\rho^{\phi}(\hat{X}^{(i)}, \tau_0)$ and the true robust semantics $\rho^{\phi}(X^{(i)}, \tau_0)$. For the test trajectory X, we analogously define the test nonconformity score

$$R := \rho^{\phi}(\hat{X}, \tau_0) - \rho^{\phi}(X, \tau_0)$$

which we naturally cannot compute during runtime as X is unknown. We denote the induced distribution of $R^{(i)}$ and R by $\mathcal{R}_0$ and $\mathcal{R}$, respectively, so that $R^{(i)} \sim \mathcal{R}_0$ and $R \sim \mathcal{R}$, i.e., $\mathcal{R}$ is a shifted version of the distribution $\mathcal{R}_0$.

By this construction of $R^{(i)}$ and R, it is easy to see from Lemmas 1 and 2 that $\text{Prob}((\rho^{\phi}(\hat{X}, \tau_0) - \rho^{\phi}(X, \tau_0) \leq \tilde{C}) \geq 1 - \delta$ where $\tilde{C} := \text{Quantile}_{1-\tilde{\delta}}(R^{(1)}, \ldots, R^{(K)})$ and $\tilde{\delta} := 1 - g^{-1}(1 - \delta_n)$. We summarize these results in Theorem 1, and provide a brief and formal proof in Appendix B.

Theorem 1. *Let the conditions from Problem 1 hold. Then, it holds that $Prob(\rho^{\phi}(X, \tau_0) \geq \rho^{\phi}(\hat{X}, \tau_0) - \tilde{C}) \geq 1 - \delta$ where $\tilde{C}$ is computed as in* (4) *with the nonconformity score $R^{(i)}$ in* (5) *defined for all calibration trajectories $X^{(i)} \in S$.*

Intuitively, this result says that the robust semantics $\rho^{\phi}(X, \tau_0)$ is lower bounded by the predicted robust semantics $\rho^{\phi}(\hat{X}, \tau_0)$ adjusted by the value $\tilde{C}$, and that this holds with high probability. While the construction of $R^{(i)}$ and R for the direct method follows [1], the result in Theorem 1 accounts for deviations between training and test distributions and is not limited to the case where $\mathcal{D}$ is identical to $\mathcal{D}_0$.

Example 2. *Recall Example 1. We set $t := 100$ and train an LSTM on 500 trajectories from $\mathcal{D}_0$ to predict the next $H := 5$ time steps. We then perform the following experiment 50 times: we sample 2000 calibration trajectories from $\mathcal{D}_0$ and 100 test trajectories from $\mathcal{D}$. For one of these experiments, we show in Figure 2a the histogram of nonconformity scores $R^{(i)}$ from* (5) *along with the robust prediction region $\tilde{C}$ computed as in* (4)*. In this case, we computed $\tilde{C}$ for the total variation distance with $\epsilon := 0.142$, which is such that $TV(\mathcal{R}, \mathcal{R}_0) \leq \epsilon$. In Section V, we explain in more detail how we estimate $TV(\mathcal{R}, \mathcal{R}_0)$ in practice. For comparison, we also plot the non-robust prediction region C from [1] which corresponds to the case where $\epsilon = 0$ and which is smaller than $\tilde{C}$ as it does not account for distribution shifts. This becomes evident in Figure 2b where we plot the empirical coverage over all 50 experiments for both algorithms. In other words, for each experiment we compute the ratio of how many of the 100 test trajectories satisfy $\rho^{\phi}(X^{(i)}, \tau_0) \geq \rho^{\phi}(\hat{X}^{(i)}, \tau_0) - \tilde{C}$ (where $\tilde{C}$ is replaced by C for the non-robust version from [1]), and then plot the histogram over these ratios. As we aim for $1-\delta = 0.8$ coverage, we can observe that only the robust algorithm from*

Theorem 1 achieves the desired coverage.

B. Interpretable Robust Predictive Runtime Verification

This direct algorithm provides a precise verification result, but lacks interpretability when the specification is violated, i.e., no information for the cause of violation is provided. Therefore, assume that the formula ϕ is in positive normal form, i.e., that ϕ contains no negations. Note that every ϕ can be re-written in positive normal form, see e.g., [41]. Further, let the formula ϕ consists of m predicates π_i, and define

$$\mathcal{P} := \big\{(\pi_i, \tau) | i \in \{1, \ldots, m\}, \tau \in \{t+1, \ldots, t+H\}\big\}$$

as the set of all predicates and times. We now design an interpretable robust predictive runtime verification algorithm, which we refer to as the robust indirect method, that uses probabilistic lower bounds $\rho^*_{\pi,\tau}$ of the robust semantics $\rho^\pi(X, \tau)$ for all predicates and times $(\pi, \tau) \in \mathcal{P}$, i.e., such that

$$\text{Prob}(\rho^\pi(X, \tau) \geq \rho^*_{\pi,\tau}, \forall (\pi, \tau) \in \mathcal{P}) \geq 1 - \delta. \tag{6}$$

Intuitively, $\rho^*_{\pi,\tau} \geq 0$ certifies that π is satisfied at time τ with probability $1-\delta$. We thus obtain interpretable information via the lower bounds $\rho^*_{\pi,\tau}$ in the presence of distribution shifts. Before we propose two ways of computing $\rho^*_{\pi,\tau}$ from the predicted trajectory $\hat{X}$, we state our main results upfront. Therefore, we recursively define the probabilistic robust semantics $\bar{\rho}^\phi$, which provide the desired probabilistic lower bound ρ^* in Problem 1, starting from predicates as

$$\bar{\rho}^\pi(\hat{X}, \tau) := \begin{cases} h(X_\tau) & \text{if } \tau \leq t \\ \rho^*_{\pi,\tau} & \text{otherwise} \end{cases}$$

while the other Boolean and temporal operators follow standard semantics, as summarized in Appendix A. We remark that the construction of $\bar{\rho}^\pi$ is again inspired by [1], but that we generalize and provide two ways to compute bounds $\rho^*_{\pi,\tau}$ that are i) robust to distribution shift, and ii) less conservative as we elaborate on more below. We next state our main results for which we provide a proof in Appendix B.

Theorem 2. *Let the conditions from Problem 1 hold, and let ϕ further be in positive normal form. If the lower bounds $\rho^*_{\pi,\tau}$ satisfy equation* (6)*, then it holds that Prob($\rho^\phi(X, \tau_0) \geq \bar{\rho}^\phi(\hat{X}, \tau_0)$) $\geq 1-\delta$ where $\bar{\rho}^\phi(\hat{X}, \tau_0)$ is recursively constructed from $\rho^*_{\pi,\tau}$ as previously described.*

Computing $\rho^*_{\pi,\tau}$ on the state level (Variant I). We now present two ways to compute $\rho^*_{\pi,\tau}$ that satisfy equation (6). In the first method (called Variant I), we compute prediction regions for trajectory predictions via robust conformal prediction. Therefore, we define the nonconformity score

$$R^{(i)} := \max_{\tau \in \{t+1, \ldots, t+H\}} \frac{\|X_\tau^{(i)} - \hat{X}_{\tau|t}^{(i)}\|}{\alpha_\tau} \tag{7}$$

where $\alpha_\tau > 0$ are constants that normalize the prediction errors at times τ, following a similar idea to [52]. In this work, however, we simply propose to compute $\alpha_\tau := \max_i \|X_\tau^{(i)} - \hat{X}_{\tau|t}^{(i)}\|$ over an additional set of trajectories $X^{(i)}$ from $\mathcal{D}_0$ that is independent from the dataset S, such as the set of training trajectories used to train the predictor μ on. Next, we define $\mathcal{B}_\tau := \{\zeta \in \mathbb{R}^n | \|\zeta - \hat{X}_{\tau|t}\| \leq \tilde{C}\alpha_\tau\}$ which is a norm ball of radius $\tilde{C}\alpha_\tau$ with center at $\hat{X}_{\tau|t}$. We then compute the worst case value of $\rho^\phi(\zeta, \tau)$ over all $\zeta \in \mathcal{B}_\tau$, i.e., we let

$$\rho^*_{\pi,\tau} = \inf_{\zeta \in \mathcal{B}_\tau} h(\zeta). \tag{8}$$

Finally, we relate this construction to equation (6) and provide a proof for the following result in Appendix B.

Lemma 3. *Let the conditions from Problem 1 hold. If $\alpha_\tau > 0$ for all $\tau \in \{t+1, \ldots, t+H\}$, then it holds that*

$$\textit{Prob}(\|X_\tau - \hat{X}_{\tau|t}\| \leq \tilde{C}\alpha_\tau, \forall \tau \in \{t+1, \ldots, t+H\}) \geq 1 - \delta \tag{9}$$

where $\tilde{C}$ is computed as in (4) *with the nonconformity score $R^{(i)}$ in* (7) *defined for all calibration trajectories $X^{(i)} \in S$. Under the same conditions, it holds that $\rho^*_{\pi,\tau}$ in* (8) *satisfy* (6)*.*

Theorem 2 and Lemma 3 together present an interpretable predictive runtime monitor that can account for distribution shifts between $\mathcal{D}_0$ and $\mathcal{D}$ via the values of $\rho^*_{\pi,\tau}$. Finally, we emphasize that the normalization via the constants α_τ greatly reduces the conservatism presented in [1] where the confidence level $1 - \delta$ has to be adjusted to $1 - \delta/H$.

Computing $\rho^*_{\pi,\tau}$ on the predicate level (Variant II). While Variant I provides interpretability, it may be the case that taking the infimum in equation (8) is conservative, e.g., as we show later in Figure 5d in our case study. We thus present a second method (called Variant II) where we compute prediction regions for each predicate π directly. Therefore, consider the nonconformity score

$$R^{(i)} := \max_{(\pi,\tau) \in \mathcal{P}} \frac{\rho^\pi(\hat{X}^{(i)}, \tau) - \rho^\pi(X^{(i)}, \tau)}{\alpha_{\pi,\tau}} \tag{10}$$

where $\alpha_{\pi,\tau} > 0$ are again normalization constants. In this work, we use $\alpha_{\pi,\tau} := \max_i |\rho^\pi(\hat{X}^{(i)}, \tau) - \rho^\pi(X^{(i)}, \tau)|$ over an additional set of trajectories $X^{(i)}$ from $\mathcal{D}_0$ that is independent from S. We conclude with the following result for which we provide a proof in Appendix B.

Lemma 4. *Let the conditions from Problem 1 hold. If $\alpha_{\pi,\tau} > 0$ for all $\tau \in \{t+1, \ldots, t+H\}$, then it holds that*

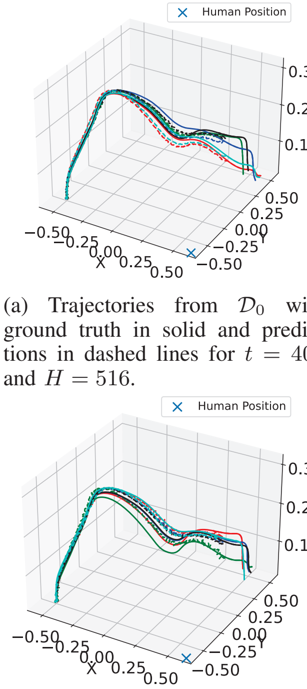
$$\textit{Prob}(\rho^\pi(\hat{X}, \tau) - \rho^\pi(X, \tau) \leq \tilde{C}\alpha_{\pi,\tau}, \forall (\pi, \tau) \in \mathcal{P}) \geq 1 - \delta \tag{11}$$

where $\tilde{C}$ is computed as in (4) *with the nonconformity score $R^{(i)}$ in* (10) *defined for all calibration trajectories $X^{(i)} \in S$. Under the same conditions, it holds that*

$$\rho^*_{\pi,\tau} := \rho^\pi(\hat{X}, \tau) - \tilde{C}\alpha_{\pi,\tau}$$

satisfy (6)*.*

Example 3. *Recall Example 1. Similar as for the direct algorithm in Example 2, we perform the same experiment of sampling* 2000 *calibration trajectories and* 100 *test trajectories* 50 *times. For one of these experiments, we show in Figures 2c and 2d the histograms of the nonconformity scores $R^{(i)}$ from* (7) *and* (10) *along with the robust prediction region $\tilde{C}$. As in Example 2, we use $\epsilon := 0.142$ which is such that $TV(\mathcal{R}, \mathcal{R}_0) \leq \epsilon$ where the induced distributions $\mathcal{R}$ and $\mathcal{R}_0$ are now with respect to equations* (7) *and* (10)*. For comparison, we also plot the non-robust prediction region C from [1]*



(a) Trajectories from $\mathcal{D}_0$ with ground truth in solid and predictions in dashed lines for $t = 400$ and $H = 516$.

(b) Trajectories from $\mathcal{D}$ with ground truth in solid and predictions in dashed lines when $t = 400$ and $H = 516$.

Fig. 3: Trajectories from $\mathcal{D}_0$ and $\mathcal{D}$ for the Franka robotic arm case study.

which corresponds to the the case where $\epsilon = 0$. In Figure 2e, we plot the histogram over all 50 experiments of the empirical coverage for Prob$(\rho^\phi(X^{(i)}, \tau_0) \geq \bar{\rho}^\phi(\hat{X}^{(i)}, \tau_0) \geq 1 - \delta$ on test trajectories $X^{(i)}$ for Variants I and II. We achieve the desired coverage of $1 - \delta = 0.8$ and compare to the non-robust versions (using C instead of $\tilde{C}$). In this case, these also achieve an empirical coverage of 0.8 as the indirect algorithms are more conservative than the direct algorithm.

C. Data Requirements and Feasibility

The algorithms presented in Theorems 1 and 2 (in combination with Lemmas 3 and 4) provide runtime verification guarantees when the prediction region $\tilde{C}$, which is computed according to Lemma 1, is finite and not unbounded (i.e., $\tilde{C} = \infty$). For this to happen, we require that $1 - \tilde{\delta} \in [0, 1]$ which is equivalent to the condition $1 - \delta_n = (1 + 1/K)g^{-1}(1 - \delta) \in [0, 1]$ as the function g and its inverse g^{-1} have domains $[0, 1]$. We note that the lower bound $0 \leq (1 + 1/K)g^{-1}(1 - \delta)$ is trivially satisfied for any $K > 0$. The upper bound $(1 + 1/K)g^{-1}(1 - \delta) \leq 1$, on the other hand, poses a lower bound on the number K of calibration trajectories as

$$K \geq \left\lceil \frac{g^{-1}(1 - \delta)}{1 - g^{-1}(1 - \delta)} \right\rceil. \tag{12}$$

This condition can be seen as a requirement on the number of calibration data K if $g^{-1}(1 - \delta) < 1$. However, it is important to observe the case where $g^{-1}(1 - \delta) = 1$. It is this condition that imposes additional conditions on the confidence $1 - \delta$ and the distribution shift ϵ for a given f-divergence. For instance, for $f(t) = \frac{1}{2}|t - 1|$, associated with the total variation distance, we know that

$$g^{-1}(1 - \delta) = \text{argsup}_{\beta \in [0,1]} \max(0, \beta - \epsilon) \leq 1 - \delta$$

which is equivalent to 1 if $\epsilon \geq \delta$. More generally, the condition $g^{-1}(1 - \delta) = 1$ will constrain the permissible distribution shift ϵ for a confidence of $1 - \delta$.

Corollary 1. *Let the conditions from Problem 1 hold. If equation* (12) *is satisfied with $g^{-1}(1 - \delta) < 1$, then the algorithms presented in Theorems 1 and 2 (in combination with Lemmas 3 and 4) provide nontrivial (i.e., $\tilde{C} < \infty$) results.*

The direct and the indirect RPRV algorithms consider the confidence level $1 - \delta$. This is opposed to the indirect method in [1] where the confidence level for the indirect method (Variant I) has to be adjusted to $1 - \delta/H$, which in this setting would impose the condition $\epsilon < \delta/H$ for $f(t) = \frac{1}{2}|t - 1|$.

Remark 1. ***Robust Locally Adaptive Predictive Runtime Verification:*** *Observing that the presented RPRV algorithms produce constant prediction regions $\tilde{C}$ regardless of the observation X_{obs}, we remark that our RPRV algorithms can be generalized to be locally adaptive [44, Section 2]. In this case, $\tilde{C}$ adapts to the observation X_{obs} and the confidence of the trajectory predictor μ via a transformation of the nonconformity score. Specifically, one can consider a locally adaptive nonconformity score $R^{(i)}_{adapt} := R^{(i)}/\omega(X^{(i)}_{obs})$ where $R^{(i)}$ is as in* (5), (7), *or* (10) *and where ω is a predictor that estimates $|R^{(i)}|$ from $X^{(i)}_{obs}$ with a trivial assumption on ω generating a positive value. At test time, it then holds that*

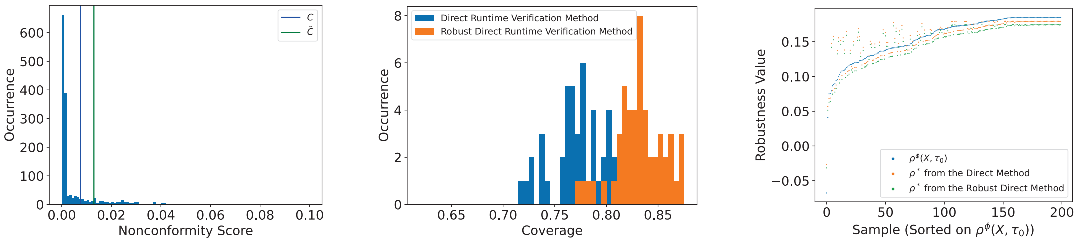
$$\text{Prob}(R_{adapt} \leq \tilde{C}\omega(X_{obs})) \geq 1 - \delta. \tag{13}$$

where $R_{adapt} \sim \mathcal{R}$ is derived from the test trajectory $X \sim \mathcal{D}$ and where $\tilde{C} :=$ Quantile$_{1-\tilde{\delta}}(R^{(1)}_{adapt}, \ldots, R^{(K)}_{adapt})$. Intuitively, a larger value of $\omega(X_{obs})$ indicates more uncertainty from the predictor μ, thus increasing the prediction region to accommodate for heteroskedascity within the test data.

V. Case Study: Franka Manipulator

To evaluate the proposed RPRV algorithms, we present a case study to verify the safety of a Franka robotic manipulator within the NVIDIA Isaac Simulator.

System Description: The Franka, located at the origin of the workspace, is given the task to pick up a cubic box of length 0.03 meters and place it at a target location $X^g := [-0.5, -0.5, 0]$. We use the Pick & Place Controller from NVIDIA Isaac Sim [53]. We also describe the state of the system at time t as $X_t := (X^p_t, X^s_t)$ where $X^p_t \in \mathbb{R}^3$ is the position of the box (controlled by the Franka robot) and X^s_t is its linear speed (which is the norm of the linear velocity). We describe the training and test distributions over trajectories X^p_0 as $\mathcal{D}_0 := [U(0.4, 0.6), U(0.4, 0.6), 0]$ and $\mathcal{D} := [U(0.41, 0.61), U(0.41, 0.61), 0]$, respectively, where $U(a, b)$ denotes a uniform distribution over the range $[a, b]$.



(a) Histogram of $R^{(i)}$ from (5) over calibration data in S along with robust prediction region $\tilde{C}$ from (4) (and C from (2)).

(b) Histogram of empirical coverage on $\rho^\phi(X, \tau_0) \geq \rho^*$ for the robust direct and the non-robust direct method from [1].

(c) Ground truth robustness $\rho^\phi(X, \tau_0)$ over a validation set in comparison with the predicted robustness ρ^* for the robust direct and the non-robust direct method from [1].

Fig. 4: Case Study: Histograms of nonconformity scores, empirical coverage, and robust semantics plots for the direct method.

We only consider trajectories X that successfully finish the placing task $\|X^p_{t+H} - X^g\|_2 < 0.3$ (i.e., we do not consider unsuccessful grasps). We assume to have access to a set of 1926 trajectories from $\mathcal{D}_0$, which we denote by Z_0, for predictor training and calibration, and a set of 934 trajectories from $\mathcal{D}$, denoted by Z, for validating our algorithms. For illustration, we show five trajectories from $\mathcal{D}_0$ and five trajectories from $\mathcal{D}$ in solid lines along with the predictions in dashed lines in Figures 3a and 3b, respectively.

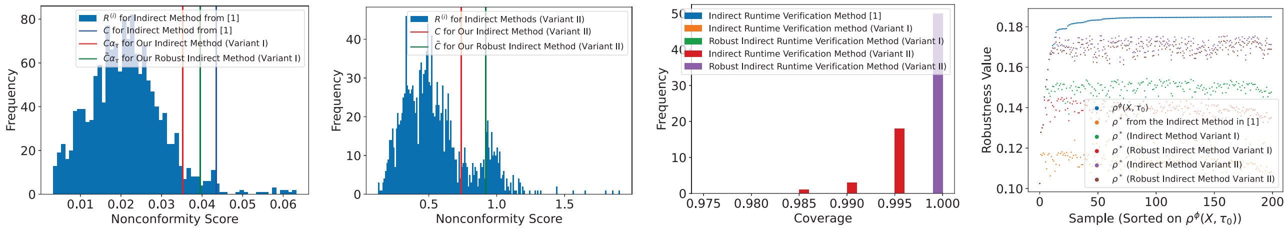
System Requirement: The robot should not only place the box at the desired location, but the robot should in the process also present the box to a human inspector located at $X^h := [0.6, -0.6, 0]$ with a moderate speed. Formally, we impose the following STL specification

$$\phi \models F_{[0,t+H]}(\|X^p_t - X^g_t\|_2 \leq \zeta_1) \wedge F_{[0,t+H]}(\|X^p_t - X^h\|_2 \\ \leq \zeta_3) \wedge G_{[0,t+H]}((\|X^p_t - X^h\|_2 < \zeta_3) \implies (X^s_t \leq \zeta_2))$$

for constants $\zeta_1 := 0.2$, $\zeta_2 := 0.6$, and $\zeta_3 := 0.9$. The formulas $F_{[0,t+H]}(\|X^p_t - X_g\|_2 \leq \zeta_1)$ and $F_{[0,t+H]}(\|X^p_t - X^h\| \leq \zeta_3)$ within ϕ are task completion requirements which specify that the robot successfully places the box at the goal location and in this process also presents the box to the inspector at X^h. The formula $G_{[0,t+H]}((\|X^p_t - X^h\|_2 < \zeta_3) \implies (X^s_t \leq \zeta_2))$ within ϕ uses the implication operator $\implies$ and expresses a safety requirement. Specifically, when the robot is close to the human inspector, the box should not exceed the speed limit of ζ_2. As we require ϕ to be in positive normal form for the indirect methods, we emphasize that this last formula is equivalent to $G_{[0,t+H]}((\|X^p_t - X^h\|_2 >= \zeta_3) \vee (X^s_t \leq \zeta_2))$, which we use in our implementation.

Distribution Shift Estimation: For validation, we estimate the distribution shift $D_f(\mathcal{D}, \mathcal{D}_0)$. Therefore, we randomly pick 800 trajectories from Z_0 and Z. For each of the nonconformity scores $R^{(i)}$ from (5), (7), and (10), we perform the following procedure: we calculate $R^{(i)}$ for the trajectories sampled from Z_0 and Z to obtain the empirical distributions of $\mathcal{R}_0$ and $\mathcal{R}$, respectively, where $\mathcal{R}_0$ and $\mathcal{R}$ are the induced distributions using the nonconformity scores over $\mathcal{D}$ and $\mathcal{D}_0$. Specifically, we use kernel density estimators with Gaussian kernels to estimate the empirical probability density functions (PDFs) of each distribution. We then numerically evaluate the distribution shift by computing $TV(\mathcal{R}, \mathcal{R}_0) = \frac{1}{2}\int_x |q(x) - p(x)| dx$ where $p(x)$ and $q(x)$ are the estimated PDFs associated with $\mathcal{R}$ and $\mathcal{R}_0$. We obtain the values ϵ_1, ϵ_2, and ϵ_3 that indicate the estimated distribution shifts on (5), (7), and (10). Finally, we take $\epsilon := \max(\epsilon_1, \epsilon_2, \epsilon_3)$ so that ϵ is greater than the estimated distribution shift of $D_f(\mathcal{R}_0, \mathcal{R})$ for all $\mathcal{R}_0$ and $\mathcal{R}$ in (5), (7), and (10). Intuitively, this allows us to validate the statistical guarantees for all our verification algorithms. We note that we used the same method to calculate the distribution shift in the running example (see Example 1), but with 1000 trajectories randomly generated from each distribution. We remark that ϵ can be thought of as a tuning parameter in practice, as it is common practice in robust control [54], and that the purpose of the estimation here is for validation of our algorithms.

Validation and Comparison of Direct Method: For this case study, we seek to find ρ^* from Problem 1 for a failure probability $\delta := 0.2$. We set $t := 400$ and predict the next $H := 516$ time steps via an LSTM trajectory predictor trained on 192 trajectories sampled from Z_0. We run the following experiments 50 times: We sample $K := 1500$ calibration trajectories from Z_0 but separate out the trajectories used for training the LSTM predictor. For one of these experiments, we show the histogram of nonconformity scores $R^{(i)}$ from (5) and the robust prediction region $\tilde{C}$ from (4) in Figure 4a with $\epsilon := 0.051$ based on the aforementioned method for distribution shift estimation. For comparison, we also show the non-robust prediction region C from the direct method in [1] (where $\epsilon = 0$), which is smaller than $\tilde{C}$ and cannot deal with the distribution shift. In Figure 4b, we plot the empirical coverage over the 50 experiments. Specifically, for each experiment, we sample a set of 200 test trajectories from Z and compute the percentage of trajectories satisfying $\rho^\phi(X^{(i)}) \geq \rho^\phi(\hat{X}^{(i)}) - C$ and $\rho^\phi(X^{(i)}) \geq \rho^\phi(\hat{X}^{(i)}) - \tilde{C}$ for the non-robust and robust methods. For one experiment, we



(a) Histogram of $R^{(i)} = \|X_\tau^{(i)} - \hat{X}_\tau^{(i)}\|$ over S along with robust prediction region $\tilde{C}\alpha_\tau$ from Lemma 3 (and $C\alpha_\tau$ with C from (2)).

(b) Histogram of $R^{(i)}$ from (10) over calibration data in S along with robust prediction region $\tilde{C}$ from (4) (and C from (2)).

(c) Histogram of empirical coverage on $\rho^\phi(X, \tau_0) \geq \rho^*$ for the robust indirect and the non-robust direct methods.

(d) Ground truth robustness $\rho^\phi(X, \tau_0)$ over a validation set in comparison with the predicted robustness ρ^* for the robust indirect methods.

Fig. 5: Case Study: Histograms of nonconformity scores, empirical coverage, and robust semantics for the indirect methods.

show the true robust semantics $\rho^\phi(X, \tau_0)$ for the 200 ground truth test data and the predicted worst-case robust semantics ρ^* in Figure 4c for the non-robust and robust RPRV algorithms.

Validation and Comparison of Indirect Methods: For practical reasons, we first downsample trajectories from 917 to 25 time steps in Z_0 and Z to reduce the computational overhead from solving the optimization problem in (3) for Variant I. We set $t := 11$ and predict the next $H := 13$ time steps via an LSTM predictor trained on 192 trajectories from Z_0, and we again select $\delta := 0.2$. We perform the same validation experiment as for the direct method for 50 times with $\epsilon := 0.076$, which we computed following the aforementioned method for estimating distribution shifts. *Variant I.* For one of these experiments, we show the histogram of the nonconformity scores $R^{(i)} := \|X_\tau^{(i)} - \hat{X}_\tau^{(i)}\|$ from the indirect method in [1] for time $\tau := 24$ in Figure 5a. For comparison, we also plot $C\alpha_\tau$ and $\tilde{C}\alpha_\tau$ following our result in Lemma 3 where C and $\tilde{C}$ are computed from equations (2) and (4) with α_τ and $R^{(i)}$ in (7). Noticeably, in Figure 5a, we see that $C\alpha_\tau$ is less conservative than C from [1]. *Variant II.* In Figure 5b, we show the histogram of $R^{(i)}$ in equation (10) over calibration data along with $\tilde{C}$ from (4) and C from (2). As expected, the indirect algorithms are more conservative than the direct algorithm and all achieve an empirical coverage for $\rho^\phi(X^{(i)}, \tau_0) \geq \rho^*$ greater than the desired coverage of $1 - \delta = 0.8$. This is demonstrated in Figure 5c where we plot the coverage over the 50 experiments. For one experiment, we show the true robust semantics $\rho^\phi(X, \tau_0)$ for the 200 ground truth test data and the predicted worst-case robust semantics ρ^* for Variants I and II in Figure 5d. Finally, in comparison with the non-robust indirect method from [1], we see that ρ^* attained by Variant I (green) and Variant II (purple) are higher than that from [1] (orange) further indicating a reduction in conservatism.

VI. Conclusion

We presented two predictive runtime verification algorithms that can predict system failures even when the test time system is different from the design time system. Specifically, our algorithms are robust against distribution shifts that are measured in terms of the f-divergence of their system trajectories. We first use trajectory predictors to predict the future motion of the system, and we robustly quantify prediction uncertainty with respect to signal temporal logic (STL) system specifications using robust conformal prediction and calibration data from the design time system. Our first algorithm (called direct algorithm) provides tight verification guarantees, while our second algorithm (called indirect algorithm), which we present in two variants, provides more interpretable runtime information. Finally, we quantify the relationship between calibration data, desired confidence, and permissible distribution shift. Both algorithms make no assumption on the trajectory predictor and only need to know an upper bound on the distribution shift. To the best of our knowledge, these are the first results in this setting. We provide an exhaustive validation of our algorithms on a Franka robot within NVIDIA Isaac Sim.

VII. Acknowledgements

This work was partially supported by the National Science Foundation through the following grants: CAREER award (SHF-2048094), CNS-1932620, CNS-2039087, FMitF-1837131, CCF-SHF-1932620, funding by Toyota R&D and Siemens Corporate Research through the USC Center for Autonomy and AI, an Amazon Faculty Research Award, and the Airbus Institute for Engineering Research. This work does not reflect the views or positions of any organization listed.

References

[1] L. Lindemann, X. Qin, J. V. Deshmukh, and G. J. Pappas, "Conformal prediction for stl runtime verification," in *Proc. of ICCPS*, 2023, pp. 142–153.

[2] M. Hekmatnejad, S. Yaghoubi, A. Dokhanchi, H. B. Amor, A. Shrivastava, L. Karam, and G. Fainekos, "Encoding and monitoring responsibility sensitive safety rules for automated vehicles in signal temporal logic," in *Proc. of MEMOCODE*, 2019, pp. 1–11.

[3] F. Cairoli, N. Paoletti, and L. Bortolussi, "Conformal quantitative predictive monitoring of stl requirements for stochastic processes," in *Proc. of HSCC*, 2023, pp. 1–11.

[4] M. Cauchois, S. Gupta, A. Ali, and J. C. Duchi, "Robust validation: Confident predictions even when distributions shift," *arXiv preprint arXiv:2008.04267*, 2020.

[5] E. Asarin and O. Maler, "Achilles and the tortoise climbing up the arithmetical hierarchy," *Journal of Computer and System Sciences*, vol. 57, no. 3, pp. 389–398, 1998.

[6] G. Agha and K. Palmskog, "A survey of statistical model checking," *ACM Transactions on Modeling and Computer Simulation (TOMACS)*, vol. 28, no. 1, pp. 1–39, 2018.

[7] A. Legay, A. Lukina, L. M. Traonouez, J. Yang, S. A. Smolka, and R. Grosu, "Statistical model checking," in *Computing and software science: state of the art and perspectives*. Springer, 2019, pp. 478–504.

[8] M. Zarei, Y. Wang, and M. Pajic, "Statistical verification of learning-based cyber-physical systems," in *Proc. Int. Conf. on Hybrid Syst.: Comp. and Control*, 2020, pp. 1–7.

[9] K. Sen, M. Viswanathan, and G. Agha, "Statistical model checking of black-box probabilistic systems," in *Proc. of CAV*. Springer, 2004, pp. 202–215.

[10] E. Bartocci, L. Bortolussi, L. Nenzi, and G. Sanguinetti, "On the robustness of temporal properties for stochastic models," in *Proc. Int. Workshop Hybrid Syst. Biology*, Taormina, Italy, Sept. 2013, pp. 3–19.

[11] ——, "System design of stochastic models using robustness of temporal properties," *Theoret. Comp. Science*, vol. 587, pp. 3–25, 2015.

[12] X. Qin, Y. Xian, A. Zutshi, C. Fan, and J. V. Deshmukh, "Statistical verification of cyber-physical systems using surrogate models and conformal inference," in *Proc. of ICCPS*, May 2022, pp. 116–126.

[13] A. Salamati, S. Soudjani, and M. Zamani, "Data-driven verification of stochastic linear systems with signal temporal logic constraints," *Automatica*, vol. 131, p. 109781, 2021.

[14] ——, "Data-driven verification under signal temporal logic constraints," *IFAC-PapersOnLine*, vol. 53, no. 2, pp. 69–74, 2020.

[15] G. Pedrielli, T. Khandait, Y. Cao, Q. Thibeault, H. Huang, M. Castillo-Effen, and G. Fainekos, "Part-x: A family of stochastic algorithms for search-based test generation with probabilistic guarantees," *IEEE Transactions on Automation Science and Engineering*, 2023.

[16] S. Dutta, M. Caprio, V. Lin, M. Cleaveland, K. J. Jang, I. Ruchkin, O. Sokolsky, and I. Lee, "Distributionally robust statistical verification with imprecise neural networks," *arXiv preprint arXiv:2308.14815*, 2023.

[17] H. Wu, T. Tagomori, A. Robey, F. Yang, N. Matni, G. Pappas, H. Hassani, C. Pasareanu, and C. Barrett, "Toward certified robustness against real-world distribution shifts," in *2023 IEEE Conference on Secure and Trustworthy Machine Learning (SaTML)*. IEEE, 2023, pp. 537–553.

[18] E. Wong and J. Z. Kolter, "Learning perturbation sets for robust machine learning," *arXiv preprint arXiv:2007.08450*, 2020.

[19] E. Bartocci, Y. Falcone, A. Francalanza, and G. Reger, "Introduction to runtime verification," *Lectures on Runtime Verification: Introductory and Advanced Topics*, pp. 1–33, 2018.

[20] A. Bauer, M. Leucker, and C. Schallhart, "Runtime verification for ltl and tltl," *ACM Transactions on Software Engineering and Methodology (TOSEM)*, vol. 20, no. 4, pp. 1–64, 2011.

[21] A. Lukina, C. Schilling, and T. A. Henzinger, "Into the unknown: Active monitoring of neural networks," in *International Conference on Runtime Verification*. Springer, 2021, pp. 42–61.

[22] M. Jaeger, K. G. Larsen, and A. Tibo, "From statistical model checking to run-time monitoring using a bayesian network approach," in *International Conference on Runtime Verification*. Springer, 2020, pp. 517–535.

[23] M. Ma, J. Stankovic, E. Bartocci, and L. Feng, "Predictive monitoring with logic-calibrated uncertainty for cyber-physical systems," *ACM Transactions on Embedded Computing Systems (TECS)*, vol. 20, no. 5s, pp. 1–25, 2021.

[24] J. V. Deshmukh, A. Donzé, S. Ghosh, X. Jin, G. Juniwal, and S. A. Seshia, "Robust online monitoring of signal temporal logic," *Formal Methods in System Design*, vol. 51, no. 1, pp. 5–30, 2017.

[25] A. Dokhanchi, B. Hoxha, and G. Fainekos, "On-line monitoring for temporal logic robustness," in *RV*. Springer, 2014, pp. 231–246.

[26] L. Gressenbuch and M. Althoff, "Predictive monitoring of traffic rules," in *Proc. of ITSC*. IEEE, 2021, pp. 915–922.

[27] A. P. Sistla and A. R. Srinivas, "Monitoring temporal properties of stochastic systems," in *International Workshop on Verification, Model Checking, and Abstract Interpretation*. Springer, 2008, pp. 294–308.

[28] A. P. Sistla, M. Žefran, and Y. Feng, "Runtime monitoring of stochastic cyber-physical systems with hybrid state," in *International Conference on Runtime Verification*. Springer, 2011, pp. 276–293.

[29] L. Bortolussi, F. Cairoli, N. Paoletti, S. A. Smolka, and S. D. Stoller, "Neural predictive monitoring," in *International Conference on Runtime Verification*. Springer, 2019, pp. 129–147.

[30] X. Qin and J. V. Deshmukh, "Clairvoyant monitoring for signal temporal logic," in *International Conference on Formal Modeling and Analysis of Timed Systems*. Springer, 2020, pp. 178–195.

[31] F. Cairoli, L. Bortolussi, and N. Paoletti, "Learning-based approaches to predictive monitoring with conformal statistical guarantees," in *International Conference on Runtime Verification*. Springer, 2023, pp. 461–487.

[32] X. Yu, W. Dong, S. Li, and X. Yin, "Model predictive monitoring of dynamical systems for stl specifications," *Automatica*, 2024.

[33] X. Yu, W. Dong, X. Yin, and S. Li, "Online monitoring of dynamic systems for signal temporal logic specifications with model information," in *2022 IEEE 61st Conference on Decision and Control (CDC)*. IEEE, 2022, pp. 1553–1559.

[34] C. Wang, X. Yu, J. Zhao, L. Lindemann, and X. Yin, "Sleep when everything looks fine: Self-triggered monitoring for signal temporal logic tasks," *arXiv preprint arXiv:2311.15531*, 2023.

[35] V. Vovk, A. Gammerman, and G. Shafer, *Algorithmic learning in a random world*. Springer Science & Business Media, 2005.

[36] Y. Romano, E. Patterson, and E. Candes, "Conformalized quantile regression," *Advances in neural information processing systems*, vol. 32, 2019.

[37] Z. Mao, C. Sobolewski, and I. Ruchkin, "How safe am i given what i see? calibrated prediction of safety chances for image-controlled autonomy," *arXiv preprint arXiv:2308.12252*, 2023.

[38] O. Maler and D. Nickovic, "Monitoring temporal properties of continuous signals," in *Proc. Int. Conf. FORMATS FTRTFT*, Grenoble, France, September 2004, pp. 152–166.

[39] A. Donzé and O. Maler, "Robust satisfaction of temporal logic over real-valued signals," in *Proc. Int. Conf. FORMATS*, Klosterneuburg, Austria, September 2010, pp. 92–106.

[40] G. E. Fainekos and G. J. Pappas, "Robustness of temporal logic specifications for continuous-time signals," *Theoret. Comp. Science*, vol. 410, no. 42, pp. 4262–4291, 2009.

[41] S. Sadraddini and C. Belta, "Robust temporal logic model predictive control," in *Proceedings of the 53rd Annual Allerton Conference on Communication, Control, and Computing*, Monticello, IL, September 2015, pp. 772–779.

[42] P. Heidlauf, A. Collins, M. Bolender, and S. Bak, "Verification challenges in f-16 ground collision avoidance and other automated maneuvers." in *ARCH@ ADHS*, 2018, pp. 208–217.

[43] G. Shafer and V. Vovk, "A tutorial on conformal prediction." *Journal of Machine Learning Research*, vol. 9, no. 3, 2008.

[44] A. N. Angelopoulos and S. Bates, "A gentle introduction to conformal prediction and distribution-free uncertainty quantification," *arXiv preprint arXiv:2107.07511*, 2021.

[45] J. Lei, M. G'Sell, A. Rinaldo, R. J. Tibshirani, and L. Wasserman, "Distribution-free predictive inference for regression," *Journal of the American Statistical Association*, vol. 113, no. 523, pp. 1094–1111, 2018.

[46] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[47] C. Fjellström, "Long short-term memory neural network for financial time series," in *Proceedings of 2022 International Conference on Big Data*. IEEE, 2022, pp. 3496–3504.

[48] C. Cortes and V. Vapnik, "Support-vector networks," *Machine learning*, vol. 20, pp. 273–297, 1995.

[49] G. Ristanoski, W. Liu, and J. Bailey, "A time-dependent enhanced support vector machine for time series regression," in *Proc. SIGKDD int. conf. on Knowledge discovery and data mining*, 2013, pp. 946–954.

[50] G. E. Box, G. M. Jenkins, G. C. Reinsel, and G. M. Ljung, *Time series analysis: forecasting and control*. John Wiley & Sons, 2015.

[51] S. Mehrmolaei and M. R. Keyvanpour, "Time series forecasting using improved arima," in *2016 Artificial Intelligence and Robotics (IRANOPEN)*. IEEE, 2016, pp. 92–97.

[52] M. Cleaveland, I. Lee, G. J. Pappas, and L. Lindemann, "Conformal prediction regions for time series using linear complementarity programming," *arXiv preprint arXiv:2304.01075*, 2023.

[53] V. Makoviychuk, L. Wawrzyniak, Y. Guo, M. Lu, K. Storey, M. Macklin, D. Hoeller, N. Rudin, A. Allshire, A. Handa *et al.*, "Isaac gym: High performance gpu-based physics simulation for robot learning," *arXiv preprint arXiv:2108.10470*, 2021.

[54] J. Doyle, "Robust and optimal control," *Control Engineering Practice*, vol. 4, no. 8, pp. 1189–1190, 1996.

[55] L. Lindemann and D. V. Dimarogonas, "Robust control for signal temporal logic specifications using discrete average space robustness," *Automatica*, vol. 101, pp. 377–387, 2019.

Appendix

A. Semantics of Signal Temporal Logic

For a trajectory $x := (x_0, x_1, \ldots)$, the semantics of an STL formula ϕ that is enabled at time τ_0, denoted by $(x, \tau_0) \models \phi$, can be recursively computed based on the structure of ϕ using the following rules:

$$\begin{aligned}
(x,\tau) &\models \text{True} && \text{iff} && \text{True},\\
(x,\tau) &\models \pi && \text{iff} && h(x_\tau) \geq 0,\\
(x,\tau) &\models \neg\phi && \text{iff} && (x,\tau) \not\models \phi,\\
(x,\tau) &\models \phi' \wedge \phi'' && \text{iff} && (x,\tau) \models \phi' \text{ and } (x,\tau) \models \phi'',\\
(x,\tau) &\models \phi' U_I \phi'' && \text{iff} && \exists \tau'' \in (\tau \oplus I) \cap \mathbb{N} \text{ s.t. } (x,\tau'') \models \phi''\\
&&&&& \text{and } \forall \tau' \in (\tau, \tau'') \cap \mathbb{N}, (x, \tau') \models \phi'.
\end{aligned}$$

The robust semantics $\rho^\phi(x, \tau_0)$ provide more information than the semantics $(x, \tau_0) \models \phi$, and indicate how robustly a specification is satisfied. We can again recursively calculate $\rho^\phi(x, \tau_0)$ based on the structure of ϕ using the following rules:

$$\begin{aligned}
\rho^{\text{True}}(x,\tau) &:= \infty,\\
\rho^{\pi}(x,\tau) &:= h(x_\tau)\\
\rho^{\neg\phi}(x,\tau) &:= -\rho^{\phi}(x,\tau),\\
\rho^{\phi'\wedge\phi''}(x,\tau) &:= \min(\rho^{\phi'}(x,\tau), \rho^{\phi''}(x,\tau)),\\
\rho^{\phi' U_I \phi''}(x,\tau) &:= \sup_{\tau'' \in (\tau\oplus I)\cap\mathbb{N}} \Big(\min \big(\rho^{\phi''}(x,\tau''),\\
&\inf_{\tau' \in (\tau,\tau'')\cap\mathbb{N}} \rho^{\phi'}(x,\tau') \big) \Big).
\end{aligned}$$

The formula length L^ϕ of a bounded STL formula ϕ can be recursively calculated based on the structure of ϕ using the following rules:

$$\begin{aligned}
L^{\text{True}} = L^{\pi} &:= 0\\
L^{\neg\phi} &:= L^{\phi}\\
L^{\phi'\wedge\phi''} &:= \max(L^{\phi'}, L^{\phi''})\\
L^{\phi' U_I \phi''} &:= \max\{I \cap \mathbb{N}\} + \max(L^{\phi'}, L^{\phi''}).
\end{aligned}$$

The probabilistic robust semantics $\bar{\rho}^\phi(X, \tau_0)$, which are defined over the random trajectory $X \sim \mathcal{D}$ and the predicted trajectory $\hat{X} := (X_{\text{obs}}, \hat{X}_{t+1|t}, \ldots, \hat{X}_{t+H|t})$, are recursively defined based on the structure of ϕ using the following rules:

$$\begin{aligned}
\bar{\rho}^{\text{True}}(\hat{X},\tau) &:= \infty,\\
\bar{\rho}^{\pi}(\hat{X},\tau) &:= \begin{cases} h(X_\tau) & \text{if } \tau \leq t\\ \rho^*_{\pi,\tau} & \text{otherwise}\end{cases}\\
\bar{\rho}^{\phi'\wedge\phi''}(\hat{X},\tau) &:= \min(\bar{\rho}^{\phi'}(\hat{X},\tau), \bar{\rho}^{\phi''}(\hat{X},\tau)),\\
\bar{\rho}^{\phi' U_I \phi''}(\hat{X},\tau) &:= \sup_{\tau'' \in (\tau\oplus I)\cap\mathbb{N}} \Big(\min \big(\bar{\rho}^{\phi''}(\hat{X},\tau''),\\
&\inf_{\tau' \in (\tau,\tau'')\cap\mathbb{N}} \bar{\rho}^{\phi'}(\hat{X},\tau') \big) \Big)
\end{aligned}$$

where the constant $\rho^*_{\pi,\tau}$ defines probabilistic prediction regions that can be computed as explained in detail in Section IV.

B. Proofs for Technical Theorems and Lemmas

1) Proof of Theorem 1

Proof. By the data processing inequality in Lemma 2 and since $D_f(\mathcal{D}, \mathcal{D}_0) \leq \epsilon$ holds by Assumption 2, we know that $D(\mathcal{R}, \mathcal{R}_0) \leq \epsilon$. We can thus apply Lemma 1 and construct $\tilde{C}$ according to equation (4) with $R^{(i)}$ as in (5). We then know that $Prob(\rho^\phi(\hat{X}, \tau_0) - \rho^\phi(X, \tau_0) \leq \tilde{C}) \geq 1 - \delta$. Therefore, it follows that $Prob(\rho^\phi(X, \tau_0) \geq \rho^\phi(\hat{X}, \tau_0) - \tilde{C}) \geq 1 - \delta$. □

2) Proof of Theorem 2

Proof. By assumption we know that equation (6) holds, i.e., that $\text{Prob}(\rho^\pi(X,\tau) \geq \rho^*_{\pi,\tau}, \forall(\pi,\tau) \in \mathcal{P}) \geq 1 - \delta$. Since we define $\bar{\rho}^\pi(\hat{X}, \tau) := h(X_\tau)$ if $\tau \leq t$ and $\bar{\rho}^\pi(\hat{X},\tau) := \rho^*_{\pi,\tau}$ otherwise, we know that $\text{Prob}(\rho^\pi(X,\tau) \geq \bar{\rho}^\pi(\hat{X},\tau), \forall(\pi,\tau) \in \mathcal{P}) \geq 1 - \delta$. Since ϕ is in positive normal form, we further know that, for any two signals $y, y' : \mathbb{N} \to \mathbb{R}^n$, it holds that $\rho^\pi(y,\tau) \geq \rho^\pi(y',\tau)$ for all $(\pi,\tau) \in \mathcal{P}$ implies $\rho^\phi(y,\tau_0) \geq \rho^\phi(y',\tau_0)$ [55, Corollary 1]. Consequently, we conclude that $\text{Prob}(\rho^\phi(X,\tau_0) \geq \bar{\rho}^\phi(\hat{X},\tau_0)) \geq 1-\delta$ since the Boolean and temporal operators for $\bar{\rho}^\phi$ follow the same semantics as for ρ^ϕ. □

3) Proof of Lemma 3

Proof. By the data processing inequality in Lemma 2 and since $D_f(\mathcal{D}, \mathcal{D}_0) \leq \epsilon$ holds by Assumption 2, we know that $D(\mathcal{R}, \mathcal{R}_0) \leq \epsilon$. We can thus apply Lemma 1 and construct $\tilde{C}$ according to equation (4) with $R^{(i)}$ as in (7). We then know that $\text{Prob}(\max_{\tau \in \{t+1,\ldots,t+H\}} \frac{\|X_\tau - \hat{X}_{\tau|t}\|}{\alpha_\tau} \leq \tilde{C}) \geq 1-\delta$, which implies that $\text{Prob}(\frac{\|X_\tau - \hat{X}_{\tau|t}\|}{\alpha_\tau} \leq \tilde{C}, \forall \tau \in \{t+1,\ldots,t+H\}) \geq 1-\delta$. Since $\alpha_\tau > 0$, this is equivalent to $\text{Prob}(\|X_\tau - \hat{X}_{\tau|t}\| \leq \tilde{C}\alpha_\tau, \forall\tau \in \{t+1,\ldots,t+H\}) \geq 1-\delta$. Finally, by the definition of $\rho^*_{\pi,\tau}$ in equation (8), which considers the worst case value of $h(\zeta)$ over $\zeta \in \mathcal{B}_\tau$, it follows that $\text{Prob}(\rho^\pi(X,\tau) \geq \rho^*_{\pi,\tau}, \forall(\pi,\tau) \in \mathcal{P}) \geq 1-\delta$ holds. □

4) Proof of Lemma 4

Proof. By the data processing inequality in Lemma 2 and since $D_f(\mathcal{D}, \mathcal{D}_0) \leq \epsilon$ holds by Assumption 2, we know that $D(\mathcal{R}, \mathcal{R}_0) \leq \epsilon$. We can thus apply Lemma 1 and construct $\tilde{C}$ according to equation (4) with $R^{(i)}$ as in (10). We then know that $\text{Prob}(\max_{(\pi,\tau)\in\mathcal{P}} \frac{\rho^\pi(\hat{X},\tau) - \rho^\pi(X,\tau)}{\alpha_{\pi,\tau}} \leq \tilde{C}) \geq 1-\delta$, which implies that $\text{Prob}(\frac{\rho^\pi(\hat{X},\tau)-\rho^\pi(X,\tau)}{\alpha_{\pi,\tau}} \leq \tilde{C}, \forall(\pi,\tau) \in \mathcal{P}) \geq 1-\delta$. Since $\alpha_{\pi,\tau} > 0$, this is equivalent to $\text{Prob}(\rho^\pi(\hat{X},\tau) - \rho^\pi(X,\tau) \leq \tilde{C}\alpha_{\pi,\tau}, \forall(\pi,\tau) \in \mathcal{P}) \geq 1-\delta$. From here, it follows that $\text{Prob}(\rho^\pi(X,\tau) \geq \rho^*_{\pi,\tau}, \forall(\pi,\tau) \in \mathcal{P}) \geq 1-\delta$ with $\rho^*_{\pi,\tau} := \rho^\pi(\hat{X},\tau) - \tilde{C}\alpha_{\pi,\tau}$. □