Quantitative Verification and Design Space Exploration Under Uncertainty with Parametric Stochastic Contracts

Chanwook Oh[1], Michele Lora[1,2], Pierluigi Nuzzo[1]
[1]University of Southern California, Los Angeles, California, USA
[2]University of Verona, Verona, Italy
{chanwooo,loramich,nuzzo}@usc.edu

Abstract

This paper proposes an automated framework for quantitative verification and design space exploration of cyber-physical systems in the presence of uncertainty, leveraging assume-guarantee contracts expressed in Stochastic Signal Temporal Logic (StSTL). We introduce quantitative semantics for StSTL and formulations of the quantitative verification and design space exploration problems as bi-level optimization problems. We show that these optimization problems can be effectively solved for a class of stochastic systems and a fragment of bounded-time StSTL formulas. Our algorithm searches for partitions of the upper-level design space such that the solutions of the lower-level problems satisfy the upper-level constraints. A set of optimal parameter values are then selected within these partitions. We illustrate the effectiveness of our framework on the design of a multi-sensor perception system and an automatic cruise control system.

ACM Reference Format:
Chanwook Oh, Michele Lora, Pierluigi Nuzzo. 2022. Quantitative Verification and Design Space Exploration Under Uncertainty with Parametric Stochastic Contracts. In *IEEE/ACM International Conference on Computer-Aided Design (ICCAD '22), October 30-November 3, 2022, San Diego, CA, USA.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3508352.3549446

1 Introduction

Design and verification of modern cyber-physical systems (CPSs) are challenging for several reasons. The number of heterogeneous requirements to be satisfied has increased as safety-critical CPSs are deployed to perform missions under strict regulations. The design space size has also expanded, requiring design decisions over multiple interconnected dimensions, e.g., control reliability, sensing accuracy, and energy consumption, among others. Finally, CPSs operate in the presence of uncertainty due, among other sources, to noise and disturbances in the system as well as the highly dynamic, unstructured, and even adversarial, environments.

Several approaches have been proposed to aid design and verification of CPSs [1]. Among these, the approaches based on assume-guarantee (A/G) contracts [2–4], often specified using temporal logics [5, 6], offer effective mechanisms to analyze system behaviors in a modular way and reason about the correctness of systems assembled from independently developed components [7–11]. However, automated compositional reasoning about system properties in the presence of uncertainty remains challenging.

This paper addresses this challenge by building on a compositional framework of *stochastic A/G contracts* [12], which relies on Stochastic Signal Temporal Logic (StSTL), a specification language for capturing probabilistic behaviors, to conveniently balance expressiveness with support for computationally tractable encodings of analysis and synthesis tasks in terms of optimization problems. StSTL contracts have shown to be effective when dealing with *qualitative verification* tasks, aiming to establish whether a system, or a class of systems, satisfies a certain property with high probability by verifying, for example, that a stochastic system satisfies a stochastic contract, or that a stochastic contract refines another stochastic contract. However, many design problems are rather concerned with *quantitative reasoning*, aiming to find, for instance, by which "margin" a certain probability bound on the satisfaction of a requirement can be guaranteed. While a number of efforts have resorted to notions of robust satisfaction of logic formulas [13–16] in a deterministic setting, effective frameworks to formulate and solve these problems in a stochastic setting have been elusive.

Our contributions is summarized as follows. We introduce quantitative semantics for StSTL in terms of a *robustness estimate*, capable of quantifying the degree of satisfaction of an StSTL formula. The robustness estimate allows reasoning about the robust satisfaction of probabilistic temporal constraints. Moreover, we show that it can be mapped to another estimate, termed *predicate robustness estimate*, which allows translating often intractable probabilistic constraints into more tractable, deterministic constraints over the statistics, e.g., expectations and variances, of the system variables.

Based on the quantitative semantics of StSTL, we formulate quantitative verification and parameter synthesis problems over parametric system models and StSTL contracts. In particular, we show that the parameter synthesis problem can be cast as a bi-level optimization problem. Moreover, for linear systems with additive Gaussian uncertainty [17, 18], the robust satisfaction of a fragment of bounded-time StSTL can be encoded into mixed integer linear programs (MILPs) or mixed integer quadratically-constrained programs (MIQCPs) which can be effectively solved using off-the-shelf solvers. Based on these encodings, we provide effective algorithms to solve the quantitative verification and parameter synthesis problems with StSTL contracts. We implement the proposed framework and algorithms in the PyCASSE toolbox and illustrate their effectiveness on the design of a multi-sensor perception and adaptive cruise control (ACC) systems.

2 Preliminaries

We introduce the class of stochastic systems we consider in this paper together with background notions on A/G contracts, StSTL, and bi-level optimization. We then provide a brief discussion of the related work. In the following, we denote the set of real numbers, non-negative real numbers, and non-negative integers by $\mathbb{R}$, $\mathbb{R}_{\geq 0}$, and $\mathbb{N}_0$, respectively. The transpose of a vector a and a matrix A are a^T and A^T, respectively. For a set B, its complement is $\overline{B}$. The empty set is denoted by $\emptyset$. $\cup$ and $\cap$ represent the union and intersection between sets, respectively. The logical symbols $\top$ denotes *true* while $\bot$ denotes *false*. Finally, $\neg$, $\wedge$, and $\vee$ denote the logical negation, conjunction, and disjunction, respectively.

ICCAD '22, October 30-November 3, 2022, San Diego, CA, USA

ACM ISBN 978-1-4503-9217-4/22/10.
https://doi.org/10.1145/3508352.3549446

2.1 Assume-Guarantee (A/G) Contracts

Let M denote a component, i.e., an element of a system, characterized by a set of *variables* V and a set of behaviors $[[M]]$ over V. We denote the *composition* of components M_1 and M_2 by $M_1 \times M_2$. An A/G contract C is a triple (V, A, G), where V is the set of variables, and A and G are sets of behaviors over V. The assumptions A are the set of behaviors that M expects from its environment. The guarantees G are the set of behaviors that M promises given that the environment provides behaviors within A. We omit V in the contract tuple, when it is clear from the context. A component E is a valid environment of C, i.e., $E \models_E C$, if $[[E]]$ is contained within A, i.e., $[[E]] \subseteq A$. A component M is a valid implementation of C, i.e., $M \models C$, if the set of its behaviors $[[M]]$ intersected with A is contained in G, i.e., $[[M]] \cap A \subseteq G$, or $[[M]] \subseteq G \cup \overline{A}$.

A contract $C = (A, G)$ is *compatible* if and only if there exists a valid environment for it, i.e., $A \neq \emptyset$, and *consistent* if and only if there exists a valid implementation, i.e., $G \cup \overline{A} \neq \emptyset$. The *refinement* relation between contracts allows reasoning about the replaceability of a contract by another contract. We say that C_2 refines C_1, written $C_2 \preceq C_1$, if and only if C_2 has weaker assumptions, i.e., $A_1 \subseteq A_2$, and stronger guarantees (in the context of the assumptions), i.e., $G_2 \cup \overline{A}_2 \subseteq G_1 \cup \overline{A}_1$. We can then replace C_1 with C_2. Contracts can be combined to construct more complex contracts using the *conjunction* ($\wedge$) and *composition* ($\otimes$) operations. Conjunction can be used, for example, to combine multiple requirements that must be satisfied simultaneously by a single component. If a component M implements the conjunction of C_1 and C_2, i.e., $M \models C_1 \wedge C_2$, then M also implements C_1 and C_2 individually, i.e., $M \models C_1$ and $M \models C_2$. On the other hand, composition can be used to combine multiple component-level requirements to obtain a system-level requirement. Given M_1 such that $M_1 \models C_1$ and M_2 such that $M_2 \models C_2$, we can reason about the properties of the composite system $M_1 \times M_2$ using $C_1 \otimes C_2$. We refer to the literature [2] for further details on the formal semantics of contracts and their algebra.

2.2 Discrete-Time Stochastic Control System

We use contracts to specify the behaviors of discrete-time stochastic systems, defined below. Let $(\Omega, \mathcal{F}, \mathbb{P})$ be a *probability space*, where Ω is a set of *outcomes*, $\mathcal{F}$ is a set of *events*, and $\mathbb{P} : \mathcal{F} \to [0, 1]$ is a function that assigns probabilities to *events* [19]. The *cumulative distribution function (cdf)* of a random variable (RV) w is defined as the probability of the event $\{w \leq \overline{w}\}$, i.e., $F_w(\overline{w}) = \mathbb{P}\{w \leq \overline{w}\}$.

Definition 1 (Discrete-Time Stochastic Control System)**.** A *discrete-time stochastic control system (dt-SCS)* is a tuple $(X, U, Z, W, V, \mathbf{w}, \mathbf{v}, f, g, h)$, where, at time step $k \in \mathbb{N}_0$:

- $x_k \in X$ is the system *state* and X is the *state space*;
- $u_k \in U$ is the *control input* and U is the *control input space*;
- $z_k \in Z$ is the *measurement* vector and Z is the *measurement space*;
- $w_k \in W$ is the *process input* and W is the *process input space*;
- $v_k \in V$ is the *measurement input* and V is the *measurement input space*;
- $\mathbf{w} := \{w_k : \Omega \to W, k \in \mathbb{N}_0\}$ and $\mathbf{v} := \{v_k : \Omega \to V, k \in \mathbb{N}_0\}$ are random processes, consisting of sequences of random vectors w_k and v_k over the probability space $(\Omega, \mathcal{F}, \mathbb{P})$;
- The system *process model* $f : X \times U \times W \to X$, the *measurement model* $g : X \times V \to Z$, and the *control law* $h : Z^{k+1} \to U$ are measurable functions [19] which describe the evolution of the stochastic system. Given the value of the initial state $\overline{x_0}$, the system *dynamics* can be written as:

$$x_{k+1} = f(x_k, u_k, w_k),\ z_k = g(x_k, v_k),\ u_k = h(z_0, \ldots, z_k). \quad (1)$$

A *behavior* of a dt-SCS is a sequence: $\boldsymbol{\xi} = \xi_0, \xi_1, \cdots$, where $\xi_k = (x_k^T, u_k^T, z_k^T, w_k^T, v_k^T)^T \in \mathbb{R}^{n_X + n_U + n_Z + n_W + n_V}$ and n_X, n_U, n_Z, n_W, n_V are the dimensions of the corresponding spaces. We also denote the behavior starting at time step k by $(\boldsymbol{\xi}, k) = \xi_k, \xi_{k+1}, \cdots$.

2.3 Stochastic Signal Temporal Logic

StSTL [12] extends Signal Temporal Logic (STL) [15] to enable the expression of probabilistic temporal properties of real-valued stochastic systems. We use StSTL to represent contract assumptions and guarantees.

2.3.1 StSTL Syntax. StSTL formulas are defined over atomic predicates (APs) represented by *chance constraints* of the form: $\mu^{[p]} := \mathbb{P}\{\mu(v) \leq 0\} \geq p$, where $\mu : \mathbb{R}^n \to \mathbb{R}$ is a real-valued measurable function, $v \in \mathbb{R}^n$ is a random vector on the probability space $(\Omega, \mathcal{F}, \mathbb{P})$, and $p \in [0, 1]$ is the *probability threshold*. An AP evaluates to $\top$ if and only if $\mu(v) \leq 0$ holds with probability larger than or equal to p. The syntax of an StSTL formula is given as follows:

$$\phi := \mu^{[p]} \mid \neg\phi \mid \phi \wedge \psi \mid \phi \mathbf{U}_{[t_1, t_2]} \psi, \quad (2)$$

where $\mu^{[p]}$ is an AP, ϕ and ψ are StSTL formulas, $t_1, t_2 \in \mathbb{N}_0 \cup \{+\infty\}$, and $\mathbf{U}$ is the *until* operator. Temporal operators such as *globally* (**G**) and *eventually* (**F**) can be expressed using the operators in (2).

2.3.2 StSTL Semantics. The semantics of StSTL are defined recursively as follows:

$$\begin{aligned}
(\boldsymbol{\xi}, k) &\models \mu^{[p]} && \leftrightarrow \mathbb{P}\{\mu(\xi_k) \leq 0\} \geq p \\
(\boldsymbol{\xi}, k) &\models \neg\phi && \leftrightarrow \neg\left((\boldsymbol{\xi}, k) \models \phi\right) \\
(\boldsymbol{\xi}, k) &\models \phi \wedge \psi && \leftrightarrow \left((\boldsymbol{\xi}, k) \models \phi\right) \wedge \left((\boldsymbol{\xi}, k) \models \psi\right) \\
(\boldsymbol{\xi}, k) &\models \phi \mathbf{U}_{[t_1, t_2]} \psi && \leftrightarrow \exists i \in [k + t_1, k + t_2] : \left((\boldsymbol{\xi}, i) \models \psi\right) \wedge \\
& && \quad (\forall j \in [k, i - 1] : (\boldsymbol{\xi}, j) \models \phi).
\end{aligned}$$

An interval $[t_1, t_2]$ in an StSTL formula can be unbounded, e.g., of the form $[t_1, +\infty)$. In this paper, we focus on *bounded-time* StSTL formulas, i.e., formulas that only include bounded intervals.

Under additional assumptions on the system model, the satisfaction of an StSTL formula can be encoded into a set of deterministic constraints over the statistics of the system variables [12]. Consider, for instance, the encoding of an AP. If the inverse of the cdf $F_{\mu(\xi_k)}$ is available, then we can translate the chance constraint $\mu^{[p]} := \mathbb{P}\{\mu(\xi_k) \leq 0\} \geq p$ into the deterministic constraint $F^{-1}_{\mu(\xi_k)}(p) \leq 0$. Previous work leverages such a transformation to provide sound encodings of the satisfaction of bounded-time StSTL formulas for discrete-time stochastic systems in terms of more tractable, deterministic mixed integer linear constraints [12]. In this paper, we extend this approach to deal with the quantitative semantics of bounded-time StSTL and the solution of robust contract-based verification and synthesis problems.

2.4 Bi-Level Optimization

A bi-level optimization problem is an optimization problem which embeds another optimization problem as a constraint. Its variables can be partitioned into two vectors, $a \in \mathbb{R}^n$ and $b \in \mathbb{R}^m$, and an optimal solution a for the upper-level optimization problem is to be selected over the set of optimal solutions B^* of the lower-level optimization problem. A generic bi-level optimization problem can

be posed in the form [20]:

$$\min_{a\in\mathbb{R}^n} F(a,b) \quad \text{s.t. } G(a,b) \le 0, \tag{3}$$

$$b \in \arg\min_{b\in\mathbb{R}^m}\{f(a,b) : g(a,b) \le 0\}, \tag{4}$$

where (3) is the upper-level optimization problem, with its objective function $F : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}$ and constraint function $G : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}$, and (4) is the lower-level problem with its objective function $f : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}$ and constraint function $g : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}$.

2.5 Related Work

A few specification languages [12, 21–25] have been proposed to capture probabilistic behaviors of hybrid systems for verification and synthesis under uncertainty. A comparison between some of these approaches can be found in the literature [12]. Robust semantics were formulated for Stochastic Temporal Logic (StTL) [24], which has similar expressiveness as StSTL, to capture margins of satisfaction in terms of probability values and predicate values. In this paper, we define robust semantics for StSTL in the probability space and then show that they can be directly mapped to a robustness estimate in the signal space, which is only a function of the statistics of the system variables and can be regarded as the "determinized" version of the original StSTL robustness estimate. Risk STL (RiSTL) has also been proposed to incorporate risk metrics into a probabilistic STL framework [25]. A fragment of RiSTL can be translated into STL formulas in the spirit of the determinization approach adopted in this paper. Differently from previous approaches, our focus is on an automated compositional framework for verification and design space exploration via notions of refinement that help quantify the level of uncertainty in multi-component systems and its propagation across different abstraction layers in a design.

The parameter synthesis problems in this paper can be cast as bi-level optimization problems, which are known to be NP-hard [26]. Simply proving the optimality of a solution is also shown to be NP-hard for this class of problems [27]. When the lower-level problem is convex and sufficiently regular [28], the bi-level problem can be reduced to a single optimization problem [29]. On the other hand, *nested approaches* using heuristic algorithms are applicable under weaker assumptions [28] but tend to be computationally expensive and provide poor optimality guarantees. Our algorithm, based on the nested approach, leverages the particular structure of the bi-level problems of interest, where the objective of the lower-level problem serves as the constraint for the upper-level problem. We solve the lower-level problem over partitions of the upper-level solution space and select only those partitions such that the solutions of the lower-level problem always satisfy the upper-level constraint. We can then effectively explore the upper-level solution space while providing optimality guarantees within a user-defined tolerance based on the granularity of the partitions.

3 Quantitative Reasoning with StSTL

While the qualitative semantics of StSTL [12] can be used to provide a yes-or-no answer to whether a system satisfies a formula, we introduce the quantitative semantics for StSTL to be able to quantify the margin by which a formula is satisfied. We then extend the logic to support parameters in the predicates.

3.1 Robustness Semantics of StSTL

By extending the quantitative semantics of STL [14], we define the quantitative semantics of StSTL via a *robustness estimate* ρ, which maps a set of system behaviors to a real number as follows:

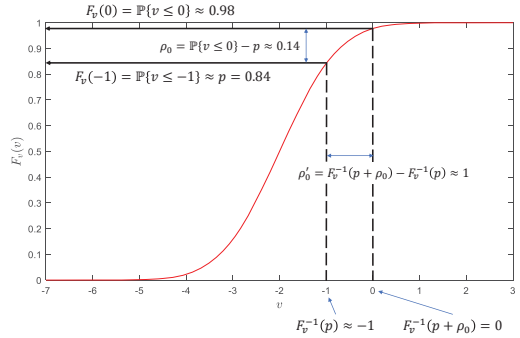

Figure 1. Robustness estimate ρ and predicate robustness estimate ρ' for the predicate $\mathbb{P}\{v \le 0\} \ge p = 0.84$ shown on the cdf F_v of v, where $v \sim N(-2,1)$. $\rho_0 = \mathbb{P}\{v \le 0\} - 0.84 \approx 0.14$ and $\rho_0' = F_v^{-1}(0.84+\rho_0) - F_v^{-1}(0.84) \approx 1$ where F_v^{-1} is the inverse cdf of v.

$$\begin{aligned}
\rho(\mu^{[p]}, \xi, k) &= \mathbb{P}\{\mu(\xi_k) \le 0\} - p \\
\rho(\neg\phi, \xi, k) &= -\rho(\phi, \xi, k) \\
\rho(\phi \wedge \psi, \xi, k) &= \min(\rho(\phi, \xi, k), \rho(\psi, \xi, k)) \\
\rho(\phi \mathbf{U}_{[t_1,t_2]} \psi, \xi, k) &= \max_{i\in[k+t_1,k+t_2]} (\rho(\psi, \xi, i), \min_{j\in[k,i-1]} \rho(\phi, \xi, j))
\end{aligned}$$

In the same way the quantitative semantics of STL have the fundamental properties of *soundness* [15, Theorem 1] and *correctness* [15, Theorem 2], the robustness estimate ρ of an StSTL can be proven to be sound and correct. Intuitively, ρ measures the margin between $\mathbb{P}\{\mu(\xi_k) \le 0\}$ and p. (ξ, k) satisfies $\mu^{[p]}$ with robustness ρ_0, if it satisfies $\mu^{[p+\Delta p]}$ for any perturbation Δp less than or equal to ρ_0, i.e., $\forall \Delta p \in [0, \rho_0]$, $(\xi, k) \models \mu^{[p+\Delta p]}$ holds.

Example 1. Consider a sensor whose noise can be modeled by a normally distributed RV x with expectation $\mathbb{E}[x] = 0$ and variance $Var[x] = 1$. We want to find the robustness margin by which the sensor satisfies the StSTL formula $(x-2)^{[0.84]}$, stating that "x is less than or equal to 2 with probability larger than or equal to 0.84." Let $v = x - 2$. As shown in Figure 1, the robustness estimate is $\rho_0 = \mathbb{P}\{v \le 0\} - 0.84 \approx 0.14$. Therefore, for any perturbation $\Delta p \le 0.13 \le \rho_0$, we still have $x \models (x-2)^{[0.84+\Delta p]}$. In other words, the noise x is less than or equal to 2 with probability as large as 0.97.

3.2 Deterministic Encoding of Robust StSTL Satisfaction

Inspired by the qualitative encoding of the satisfaction of an StSTL formula into deterministic constraints, we look for transformations of the probabilistic robustness estimate into a deterministic estimate. We assume that the inverse of the cdf $F_{\mu(\xi_k)}$ is available. We can then define the *predicate robustness estimate* ρ' of an StSTL AP as follows:

$$\rho'(\mu^{[p]}, \xi, k, \rho) = F^{-1}_{\mu(\xi_k)}(p+\rho) - F^{-1}_{\mu(\xi_k)}(p), \tag{5}$$

where ρ' measures the margin between $F^{-1}_{\mu(\xi_k)}(p+\rho)$ and $F^{-1}_{\mu(\xi_k)}(p)$. ρ' preserves the order and sign of ρ by the monotonicity of $F^{-1}_{\mu(\xi_k)}$.

ρ' can then be used in the same way as ρ to quantitatively reason about the satisfaction of an StSTL formula. (ξ, k) satisfies $\mu^{[p]}$ with predicate robustness ρ_0' if, for any perturbation $\Delta\mu$ less than or equal to ρ_0', $(\xi, k) \models (\mu + \Delta\mu)^{[p]}$ holds. Since $\mathbb{P}\{\mu(\xi_k) \le 0\} = p + \rho$ if

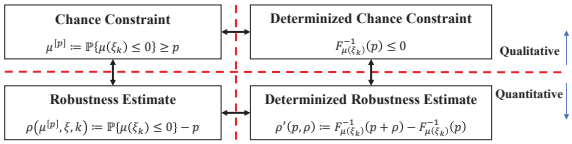

Figure 2. Relationships between qualitative and quantitative satisfaction of StSTL APs and their deterministic encodings.

and only if $F^{-1}_{\mu(\xi_k)}(p+\rho) = 0$, and $\rho = \mathbb{P}\{\mu(\xi_k) \leq 0\} - p$, by the robustness semantics of StSTL, we conclude:

$$\rho'(\mu^{[p]}, \boldsymbol{\xi}, k) = -F^{-1}_{\mu(\xi_k)}(p)\,. \tag{6}$$

Example 2. Consider the StSTL AP $v^{[0.84]} = (x-2)^{[0.84]}$ in Example 1. As shown in Figure 1, we have $\rho'_0 = -F^{-1}_v(0.84) \approx 1$ and, accordingly, $(\boldsymbol{\xi}, k) \models (v + \Delta\mu)^{[0.84]}$ for all $\Delta\mu \leq 1 \leq \rho'_0$. In other words, with probability larger than or equal to 0.84, the sensor noise x is less than or equal to c, with c as small as 1.

Figure 2 summarizes the relationships between qualitative and quantitative satisfaction of an StSTL AP and their encodings into deterministic constraints. The robustness estimate ρ (bottom-left constraint in Figure 2) provides a quantitative measure of satisfaction for the StSTL AP $\mu^{[p]}$ as a real number between -1 and 1. We can convert ρ into a predicate robustness estimate ρ' (bottom-right constraint in Figure 2), which translates a margin defined in the probability space into a margin in terms of the statistics of the predicate μ and, ultimately, the statistics of the system variables. ρ' can assume any real value.

3.3 Parametric Stochastic Signal Temporal Logic

Parametric StSTL (PStSTL) extends StSTL [12] with parameters. Let $\pi \in \Pi$ be a set of parameters partitioned into two disjoint sets of *signal parameters*, $\pi_s = \{s_1, s_2, \ldots\}$, with domain Π_s, and *probability threshold parameters*, $\pi_p = \{p_1, p_2, \ldots\}$, with domain Π_p. We denote a PStSTL formula ϕ parametrized by π by $\phi(\pi)$. For example, $\phi(\pi) \coloneqq \mathbf{F}_{[0,5]}(x+s)^{[p]}$ has parameter set $\pi = \{s, p\}$.

4 Problem Formulation

We express the sets A and G of a contract $C = (A, G)$ using StSTL formulas ϕ_A and ϕ_G, respectively. The formula $\phi_C = \phi_A \rightarrow \phi_G$ specifies the set of behaviors of any valid implementation of C. Operations and relations between assumptions and guarantees in contracts can then be mapped to operations and relations between formulas [12]. Compatibility, consistency, and refinement checking can be translated into satisfiability or validity checking of formulas. Robust verification translates, instead, into *robust satisfaction* and *robust refinement* problems, defined below.

Problem 1 (Robust Satisfaction)**.** *Given a stochastic system M, a stochastic contract $C = (\phi_A, \phi_G)$, and $\rho^* \in \mathbb{R}_{\geq 0}$, check whether M satisfies C with robustness (at least) ρ^*, written $M \models_{\rho^*} C$, i.e., whether $\rho(\phi_C, \boldsymbol{\xi}_M, k) \geq \rho^*$ for all system behaviors $(\boldsymbol{\xi}_M, k)$.*

Problem 2 (Robust Refinement)**.** *Let $C_1 = (\phi_{A1}, \phi_{G1})$ and $C_2 = (\phi_{A2}, \phi_{G2})$ be stochastic contracts defined on a stochastic system M. Given $\rho^* \in \mathbb{R}_{\geq 0}$, check whether C_2 refines C_1 with robustness (at least) ρ^*, written $C_2 \preceq_{\rho^*} C_1$, i.e., whether $\rho(\phi_{A1} \rightarrow \phi_{A2}, \boldsymbol{\xi}_M, k) \geq \rho^*$ and $\rho(\phi_{C_2} \rightarrow \phi_{C_1}, \boldsymbol{\xi}_M, k) \geq \rho^*$ for all system behaviors $(\boldsymbol{\xi}_M, k)$.*

On the other hand, we cast design space exploration problems as *parameter synthesis problems* where a set of optimal parameter values must be selected such that the contracts are satisfied. To formulate the problem, we first introduce parametric systems and contracts.

During the design process, some constants in the stochastic system may be regarded as design parameters. We represent such scenario with a *parametric stochastic system (component)* $M(\pi_M)$, where π_M is a set of parameters and Π_M is the domain for π_M. Similarly, some elements in the stochastic contract specification may also be parametrized. We model such scenario with a *parametric stochastic contract* which has assumptions and guarantees written in PStSTL, i.e., $C(\pi_C) = (\phi_A(\pi_C), \phi_G(\pi_C))$, where π_C is a set of parameters, and Π_C is the domain for π_C.

Problem 3 (Parameter Synthesis)**.** *Given a parametric stochastic system $M(\pi_M)$, a parametric stochastic contract $C(\pi_C) = (\phi_A(\pi_C), \phi_G(\pi_C))$, and a cost function $J : \Pi_C \times \Pi_M \rightarrow \mathbb{R}$, find an optimal set of parameters $\pi = \pi_C \cup \pi_M$ such that $M(\pi_M) \models_{\rho^*} C(\pi_C)$.*

Problem 3 can be written as a bi-level optimization problem:

$$\min_{\pi \in \Pi_C \times \Pi_M} J(\pi) \text{ s.t. } \rho(\phi_C(\pi_C), \boldsymbol{\xi}_M, k) \geq \rho^* \tag{7}$$

$$(\boldsymbol{\xi}_M, k) \in \arg\min_{(\boldsymbol{\xi}_M, k)} \{\rho(\phi_C(\pi_C), \boldsymbol{\xi}_M, k) : (\boldsymbol{\xi}_M, k) \models \phi_A(\pi_C)\} \tag{8}$$

(7) is the upper-level problem and (8) is the lower-level problem. Problem (8) searches for system behaviors that minimize the robustness estimate $\rho(\phi_C(\pi_C), \boldsymbol{\xi}_M, k)$. In (8), we require that $\phi_A(\pi_C)$ hold since we are interested in behaviors which robustly satisfy the contract $\phi_C(\pi_C)$ under valid environments. If the resulting minimum robustness estimate $\rho_{min}(\phi_C(\pi_C), \boldsymbol{\xi}_M, k)$ for the lower-level problem is larger than or equal to ρ^*, then all the system behaviors robustly satisfy $\phi_C(\pi_C)$. Only if the parameters provide such a guarantee, they are considered as the feasible space for the upper-level problem (7), where we search for the set of optimal parameter values π^*. We can also synthesize parameters such that refinement is guaranteed to hold robustly.

Problem 4 (Parameter Synthesis Under Refinement)**.** *Let $C_1 = (\phi_{A1}, \phi_{G1})$ be a stochastic contract and $C_2(\pi_C) = (\phi_{A2}(\pi_C), \phi_{G2}(\pi_C))$ be a parametric stochastic contract defined on a parameteric stochastic system $M(\pi_M)$. Given a cost function $J : \Pi_C \times \Pi_M \rightarrow \mathbb{R}$, find an optimal set of parameters $\pi = \pi_C \cup \pi_M$ such that $C_2(\pi_C) \preceq_{\rho^*} C_1$ and $M_2(\pi_M) \models_{\rho^*} C_2(\pi_C)$.*

Problem 4 can also be written as a bi-level optimization problem:

$$\min_{\pi \in \Pi_C \times \Pi_M} J(\pi) \text{ s.t. } \rho(\phi_{\preceq_G}(\pi_C) \wedge \phi_{C2}(\pi_C), \boldsymbol{\xi}_M, k) \geq \rho^* \tag{9}$$

$$(\boldsymbol{\xi}_M, k) \in \arg\min_{(\boldsymbol{\xi}_M, k)} \{\rho(\phi_{\preceq_G}(\pi_C) \wedge \phi_{C2}(\pi_C), \boldsymbol{\xi}_M, k) : (\boldsymbol{\xi}_M, k) \models \phi_{\preceq_A}(\pi_C) \wedge \phi_{A1}\} \tag{10}$$

where $\phi_{\preceq_A}(\pi_C) \coloneqq \phi_{A1} \rightarrow \phi_{A2}(\pi_C)$ and $\phi_{\preceq_G}(\pi_C) \coloneqq \phi_{C_2}(\pi_C) \rightarrow \phi_{C_1}$ are the conditions for the robust refinement $C_2(\pi_C) \preceq_{\rho^*} C_1$ to hold, respectively; $\phi_{C_2}(\pi_C)$ is the condition for $M_2(\pi_M) \models C_2(\pi_C)$ to hold; (9) is the upper-level, and (10) is the lower-level problem. Problem (10) searches for the minimum robustness value for the refinement under the constraints that the system is implementable and operates in a valid environment. If the minimum robustness estimate $\rho_{min}(\phi_{\preceq_G}(\pi_C) \wedge \phi_{C2}(\pi_C), \boldsymbol{\xi}_M, k)$ is larger than or equal to ρ^*, we solve (9) to find the set of optimal parameter values π^*. In the following, we discuss effective methods to solve the problems above under additional assumptions on the system.

5 Deterministic Encoding of StSTL Satisfaction

Given the stochastic system model, the predicate robustness estimate (6) allows translating the robust satisfaction of an StSTL formula into a set of deterministic mixed integer constraints which can be used to formulate the quantitative verification problems in Section 4. The verification problems can be used as elements for the solution of the parameter synthesis problems. In this paper, we instantiate these problems for the class of discrete-time linear systems with Gaussian process and measurement inputs, expressive enough to capture a large number of applications [17, 30, 31]. We assume that the dt-SCS M is governed by the following dynamics:

$$x_{k+1} = Ax_k + Bu_k + w_k,\ z_k = Cx_k + v_k,\ u_k = Dz_k + E, \tag{11}$$

where $A \in \mathbb{R}^{n_X \times n_X}$, $B \in \mathbb{R}^{n_X \times n_U}$, $C \in \mathbb{R}^{n_Z \times n_X}$, $D \in \mathbb{R}^{n_U \times n_Z}$, $E \in \mathbb{R}^{n_U \times 1}$, and the process and measurement inputs are white Gaussian independent processes with $w_k \sim \mathcal{N}(0, Q)$ and $v_k \sim \mathcal{N}(0, R)\ \forall\ k \in \mathbb{N}_0$, $Q \in \mathbb{R}^{n_X \times n_X}$ and $R \in \mathbb{R}^{n_Z \times n_Z}$ being the process and measurement input covariance matrices, respectively. Let StSTL APs be in the form $\mathbb{P}\{\mu(\xi_k) \leq 0\} \geq p$ with:

$$\mu(\xi_k) = a^T x_k + b^T u_k + c^T z_k + d^T w_k + e^T v_k + f, \tag{12}$$

where $a \in \mathbb{R}^{n_X \times 1}$, $b \in \mathbb{R}^{n_U \times 1}$, $c \in \mathbb{R}^{n_Z \times 1}$, $d \in \mathbb{R}^{n_W \times 1}$, $e \in \mathbb{R}^{n_V \times 1}$, and $f \in \mathbb{R}$. We can then state the following result.

Theorem 1. *For a stochastic system M governed by* (11) *and the StSTL AP $\mathbb{P}\{\mu(\xi_k) \leq 0\} \geq p$ with $\mu(\cdot)$ in* (12)*, the predicate robustness estimate* (6) *is equivalent to*

$$\rho'_k(\mu^{[p]}, M) = -\mathbb{E}[\mu(\xi_k)] - F^{-1}(p)\,\sigma_{\mu(\xi_k)}, \tag{13}$$

where $\mathbb{E}[\mu(\xi_k)]$ and $Var[\mu(\xi_k)] = \sigma^2_{\mu(\xi_k)}$ can be computed as:

$$\mathbb{E}[\mu(\xi_k)] = \beta\alpha^k \overline{x_0} + \beta \sum_{i=1}^{k} \alpha^{k-i} BE + b^T E + f, \tag{14}$$

$$\begin{aligned} Var[\mu(\xi_k)] = &\sum_{i=1}^{k} (\beta\alpha^{k-i}) Q (\beta\alpha^{k-i})^T + d^T Q d + \gamma R \gamma^T \\ &+ \sum_{i=1}^{k} (\beta\alpha^{k-i-1} BD) R (\beta\alpha^{k-i-1} BD)^T, \end{aligned} \tag{15}$$

$F^{-1}(\cdot)$ is the inverse cdf of a standard Gaussian RV, $\alpha = A + BDC$, $\beta = a^T + b^T DC + c^T C$, and $\gamma = b^T D + c^T + e^T$.

Proof. From (11), x_k, u_k, and z_k are linear combinations of the Gaussian processes w_k and v_k; thus, they follow Gaussian distributions. $\mu(\xi_k)$ is a linear combination of x_k, u_k, z_k, w_k, and v_k; hence, $\mu(\xi_k)$ follows a Gaussian distribution. By substituting x_k, u_k, and z_k in (12), we get exact expressions for $\mu(\xi_k)$ which is normally distributed. From this expression, we obtain (14) and (15). ∎

Given the deterministic encoding of an StSTL AP in (13), the satisfaction of a bounded-time StSTL formula ϕ can be encoded as a conjunction of deterministic constraints by applying the same techniques proposed in the literature for StSTL [12]. Quantitative encoding requires, however, taking the min and max of the robustness estimates. We encode these constraints by using the methods previously proposed for robust satisfaction of STL formulas [32] into mixed integer constraints. In the general case, the deterministic encoding of bounded-time StSTL satisfaction becomes a mixed integer nonlinear program (MINLP) which is usually intractable. However, if f in (12) and p in (6) are the only parameters and a piecewise linear approximation of $F^{-1}(\cdot)$ is available, the encoding reduces to a MILP that can be effectively solved [33], as further exemplified below.

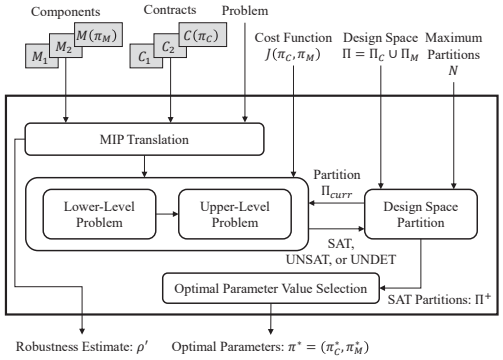

Figure 3. Overview of the proposed framework for finding an optimal set of parameter values for a stochastic system.

Example 3. Consider a dt-SCS M and the PStSTL formula $\phi(c, p) := \mathbf{G}_{[0,2]}(x - c)^{[p]}$ where $c \in [0, 5]$ and $p \in [0.85, 0.95]$. The initial state is $x_0 = 0$; the system dynamics are $x_{k+1} = x_k + w_k$; and $w_k \sim \mathcal{N}(1, 2)$ for each time step $k = 0, 1, 2$. The distributions of x_k at each time step k can be found as follows: $x_0 = 0$, $x_1 = x_0 + w_0 \sim \mathcal{N}(1, 2)$, and $x_2 = x_1 + w_1 \sim \mathcal{N}(2, 4)$. The robustness estimate is $\rho_k = \mathbb{P}\{x_k - c \leq 0\} - p$, but we resort to the predicate robustness $\rho'_k(\phi(c, p), M) = -\mathbb{E}[x_k - c] - \widetilde{F^{-1}}(p)\,\sigma[x_k - c]$, $k = 0, 1, 2$, from (13), where $\widetilde{F^{-1}}(p)$ is a piecewise linear approximation of $F^{-1}(p)$. From [32], we get the robustness $\rho'(\phi(c, p), M) = \min(\rho'_0, \rho'_1, \rho'_2)$, where the min operator encodes the globally (**G**) operator. The constraint $\rho'(\phi(c, p), M) \geq \rho^*$ enforces the robust satisfaction of $\phi(c, p)$.

If the elements in Q or R are further used as parameters, quadratic terms are introduced in the expression for $Var[\mu(\xi_k)] = \sigma^2_{\mu(\xi_k)}$; hence, the encoding reduces to a MIQCP, for which there are also a number of effective solution methods [34] and off-the-shelf tools.

6 Robust Verification and Parameter Synthesis

Our automated framework addresses the problems formulated in Section 4 using the encoding detailed in Section 5. Figure 3 depicts the overall structure of the framework, taking as inputs the component models, the contracts, the problem, the cost function, the design space, and the maximum number of partitions used for parameter synthesis. The framework either returns 1) the robustness margin by which the system satisfies the contracts or 2) a set of optimal parameters with respect to the cost function.

We denote a set of mixed integer constraints, encoding the satisfaction problem for StSTL formula ϕ and the stochastic system M at time step k, by $C_k[\phi, M]$. The inequality $\rho'_k(\phi, M) \geq \rho^*$ is then a constraint in $C_k[\phi, M]$. A behavior (ξ_M, k) of M satisfies an StSTL formula ϕ, i.e., $(\xi_M, k) \models \phi$, if $C_k[\phi, M]$ is feasible.

Problem 1. Given the stochastic system M, a stochastic contract $C = (\phi_A, \phi_G)$, and $\phi_C := \phi_A \rightarrow \phi_G$, we compute the minimum value of the predicate robustness $\rho'_k(\phi_C, M)$ by solving the following optimization problem:

$$\min_{(\xi_M, k)} \rho'_k(\phi_C, M) \text{ s.t. } C_k[\phi_C, M] \setminus (\rho'_k(\phi_C, M) \geq \rho^*)$$

If the minimum value of $\rho'_k(\phi_C, M)$ is larger than or equal to ρ^*, then M satisfies C with robustness (at least) ρ^*, i.e., $M \models_{\rho^*} C$.

Algorithm 1: $findOptParam(M(\pi_M), C(\pi_C), \Pi, J, N)$

input : A Parameteric System, $M(\pi_M)$,
A Parameteric Contract, $C(\pi_C)$,
A Parameter Space, $\Pi = \Pi_C \cup \Pi_M$,
A Cost Function, $J : \Pi_C \times \Pi_M \to \mathbb{R}$,
Maximum Number of Partitions, N_{max}

output : A Set of Optimal Parameter Values,
$\pi^* = \pi^*_C \cup \pi^*_M \in \Pi$

1 $\Pi^+, \Pi^- \leftarrow \emptyset, \Pi^U \leftarrow \{\Pi\}, r_{min} \leftarrow \infty$
2 **while** $\Pi^U \neq 0$ *and* $|\Pi^+| + |\Pi^-| + |\Pi^U| \leq N_{max}$ **do**
3 $\quad \Pi_{curr} \leftarrow$ DEQUEUE(Π^U)
4 $\quad$ **if** *isSATPartition*$(M(\pi_M), C(\pi_C), \Pi_{curr})$ **then**
5 $\quad\quad \Pi^+ \leftarrow \Pi^+ \cup \Pi_{curr}$
6 $\quad$ **else if** *isUNSATPartition*$(M(\pi_M), C(\pi_C), \Pi_{curr})$ **then**
7 $\quad\quad \Pi^- \leftarrow \Pi^- \cup \Pi_{curr}$
8 $\quad$ **else**
9 $\quad\quad \Pi^U \leftarrow \Pi^U \cup$ paramSpacePartition(Π_{curr})
10 **for** Π_{curr} *in* Π^+ **do**
11 $\quad$ **if** $\max_{\pi \in \Pi_{curr}} J(\pi) > r_{max}$ **then**
12 $\quad\quad \pi^* \leftarrow \arg\min_{\pi \in \Pi_{curr}} J(\pi)$
13 $\quad\quad r_{min} \leftarrow J(\pi^*)$
14 **return** π^*

Problem 2. Given $C_1 = (\phi_{A1}, \phi_{G1})$ and $C_2 = (\phi_{A2}, \phi_{G2})$ where $\phi_{\preceq_A} := \phi_{A1} \to \phi_{A2}$ and $\phi_{\preceq_G} := (\phi_{A2} \to \phi_{G2}) \to (\phi_{A1} \to \phi_{G1})$, we compute the minimum values of $\rho'_k(\phi_{\preceq_A}, M)$ and $\rho'_k(\phi_{\preceq_G}, M)$ by solving the following optimization problems.

$$\min_{(\xi_M, k)} \rho'_k(\phi_{\preceq G}, M) \text{ s.t. } C_k\big[\phi_{\preceq G}, M\big] \setminus (\rho'_k(\phi_{\preceq G}, M) \geq \rho^*)$$

$$\min_{(\xi_M, k)} \rho'_k(\phi_{\preceq A}, M) \text{ s.t. } C_k\big[\phi_{\preceq A}, M\big] \setminus (\rho'_k(\phi_{\preceq A}, M) \geq \rho^*)$$

If both values are larger than or equal to ρ^*, C_2 refines C_1 with robustness (at least) ρ^*, i.e., $C_2 \preceq_{\rho^*} C_1$.

Problem 3. We solve the parameter synthesis problem using Algorithm 1. Given a parametric stochastic system $M(\pi_M)$, a parameteric stochastic contract $C(\pi_C)$, the parameter space $\Pi = \Pi_C \times \Pi_M$, a cost function $J(\pi)$, and the maximum number of partitions N_{max}, we find the set of optimal parameter values $\pi^* = (\pi^*_C, \pi^*_M)$ by solving the bi-level optimization problem defined in (7) and (8).

We apply a nested method, where the lower-level problem (8) is solved for each partition of the upper-level problem parameter space. The parameter space Π is split into multiple partitions and each partition is classified as a SAT partition, an UNSAT partition, or an UNDET partition. Any set of parameter values chosen within a SAT partition ensures that $M(\pi_M) \models_{\rho^*} C(\pi_C)$. There exists no set of parameter values in an UNSAT partition such that $M(\pi_M) \models_{\rho^*} C(\pi_C)$ holds, while a set of parameter values chosen from an UNDET partition may or may not ensure $M(\pi_M) \models_{\rho^*} C(\pi_C)$. The upper-level solution is optimized only over the SAT partitions.

Algorithm 1 initializes the SAT and UNSAT partitions as empty sets and the UNDET partition as $\{\Pi\}$ (line 1). While UNDET partitions exist and the number of all partitions is less than or equal to N_{max} (line 2), the algorithm selects a partition Π_{curr} from a set of UNDET partitions using the DEQUEUE function (line 3). We classify the partition Π_{curr} as a SAT partition, i.e., isSATPartition$(\cdot) = \top$, if the value of the objective function for the following optimization problem is larger than or equal to ρ^*:

$$\min_{\pi \in \Pi_{curr}, (\xi_M, k)} \rho'_k\big(\phi_C(\pi_C), M(\pi_M)\big)$$
$$\text{s.t. } C_k\big[\phi_C(\pi_C), M(\pi_M)\big] \setminus (\rho'_k\big(\phi_C(\pi_C), M(\pi_M)\big) \geq \rho^*)$$
$$\text{and } C_k\big[\phi_A(\pi_C), M(\pi_M)\big].$$

If $\rho'_k\big(\phi_C(\pi_C), M(\pi_M)\big) \geq \rho^*$, then any set of parameter values within Π_{curr} is a robust set of parameter values and it is classified as a SAT partition (line 4-5).

The partition Π_{curr} is classified as an UNSAT partition, i.e., isUNSATPartition$(\cdot) = \top$ if the value of the objective function for the following optimization problem is smaller than ρ^*:

$$\max_{\pi \in \Pi_{curr}, (\xi_M, k)} \rho'_k\big(\phi_C(\pi_C), M(\pi_M)\big)$$
$$\text{s.t. } C_k\big[\phi_C(\pi_C), M(\pi_M)\big] \setminus (\rho'_k\big(\phi_C(\pi_C), M(\pi_M)\big) \geq \rho^*)$$
$$\text{and } C_k\big[\phi_A(\pi_C), M(\pi_M)\big].$$

If $\rho'_k\big(\phi_C(\pi_C), M(\pi_M)\big) < \rho^*$, no set of parameter values within Π_{curr} is a robust set of parameter values and Π_{curr} is classified as an UNSAT partition (line 6-7).

If the partition Π_{curr} is neither SAT nor UNSAT partition, Π_{curr} is classified as an UNDET partition and is further partitioned into smaller partitions using the paramSpacePartition$(\cdot)$ function (line 8-9). Any partitioning scheme can be used in Algorithm 1. We implement the *greatest uncertainty split* which halves the range of each dimension, partitioning any n-dimension partition into 2^n partitions. Lastly, a set of optimal parameter values is selected and the corresponding cost is computed for each SAT partition (line 10-13). The set of optimal parameter values π^* with the least cost r_{min} is returned as a solution (line 14).

Problem 4. Given $C_1 = (\phi_{A1}, \phi_{G1})$ and $C_2(\pi_C) = (\phi_{A2}(\pi_C), \phi_{G2}(\pi_C))$, the parameter synthesis under refinement problem can be solved using Algorithm 1 with the contract $C_{\preceq}(\pi_C) = (\phi_{\preceq_A}(\pi_C), \phi_{\preceq_G}(\pi_C))$, whose assumptions and guarantees are defined as $\phi_{\preceq_A}(\pi_C) := (\phi_{A1} \to \phi_{A2}(\pi_C)) \wedge \phi_{A1}$ and $\phi_{\preceq_G}(\pi_C) := (\phi_{C2}(\pi_C) \to \phi_{C1}) \wedge \phi_{C2}(\pi_C)$, respectively.

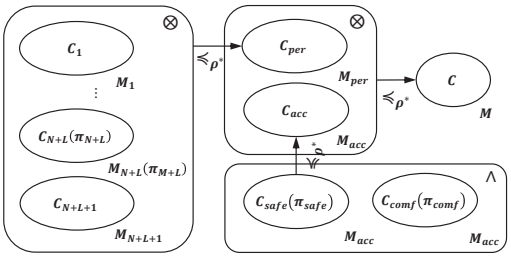

Figure 4. Design examined in the case studies, including a multi-sensor perception system M_{per} with N fixed sensors and L unknown sensors, and an ACC system M_{acc} with safety and comfort requirements.

7 Case Studies

We implemented our approach in the Python Contract-based Analysis for Stochastic System Exploration (PyCASSE)[1] library, and evaluated its effectiveness on the vehicle design scenario in Figure 4 by executing our experiments on an Intel core i7 processor with 16-GB RAM. The ego vehicle M, i.e., the system under control with its

[1] https://github.com/descyphy/pycasse

requirements specified in a top-level contract C, consists of a multi-sensor perception and an ACC components M_{per} and M_{acc}, specified by contracts C_{per} and C_{acc}, respectively. The multi-sensor perception system M_{per}, specified in a contract C_{per}, consists of multiple sensors $M_i, i = 1, \cdots, N+L$, and a data fusion module M_{N+L+1} specified by $C_i, i = 1, \cdots, N+L$ and C_{N+L+1}, respectively. Its correctness can be checked by showing that $\bigotimes_{i=1}^{N+L+1} C_i \preceq_{\rho^*} C_{per}$ and $M_i \models_{\rho^*} C_i, i = 1, \ldots, N+L+1$. The ACC system M_{acc} must satisfy both the safety and comfort contracts C_{safe} and C_{comf}, respectively, which can be checked by showing that $M_{acc} \models_{\rho^*} C_{safe}$ and $M_{acc} \models_{\rho^*} C_{comf}$.

In this scenario, we focus on two case studies in Figure 4. In the first case study, we design a multi-sensor perception system M_{per} consisting of sensors with different noise levels. We synthesize the sensor and the requirement parameters such that refinement $\bigotimes_{i=1}^{N+L+1} C_i \preceq_{\rho^*} C_{per}$ holds, while varying the number of sensors, N and L. In the second case study, we explore the safety and comfort requirements C_{safe} and C_{comf} for the ACC system which regulates the ego vehicle's speed, based on the sensor measurements of the distance and velocity of the ego and the leading vehicles. We model the multi-sensor perception and ACC systems using the variables:

- x_e, x_l: location of the ego and the leading vehicle;
- v_e, v_l: velocity of the ego and the leading vehicle;
- $d = x_l - x_e, v = v_l - v_e$: relative distance and velocity between the ego and the leading vehicles;
- a_e, a_l: acceleration of the ego and the leading vehicle.

We denote the values of these variables at time step k with the subscript k, e.g., the relative velocity v between the ego and the leading vehicles at time step k is denoted by v_k.

7.1 Multi-Sensor Perception System

The multi-sensor perception system has N specified sensors and L unspecified sensors captured by parametric stochastic contracts. In this case study, we aim to search for inexpensive sensors that can be affected by noise, i.e., large standard deviations σ_j within the $[0.2, 2]$ range, while still guaranteeing with probability (confidence) p_j in $[0.8, 1]$ that the sensor error lies in the $[-1, 1]$ interval, where $j = 1, \cdots, L$. The system M_{per} has its contract $C_{per} = (\mathbf{G}_{[0,10]}(d \leq 250),\ \mathbf{G}_{[0,10]}(|n_d| - 1)^{[0.99]})$, which expresses the requirement that, if the distance d is always less than or equal to 250 m, the absolute value of the distance sensor noise n_d is always less than or equal to 1 with probability larger than or equal to 0.99. Let the dynamics of the specified sensors be $z_{i,k} = d_k + n_{d,i,k}$ for $i = 1, \cdots, N$, where $n_{d,i,k} \sim \mathcal{N}(0, 0.5^2)$. According to the cost function $J = -\sum_{j=1}^{M}(\sigma_{N+j} + p_{N+j})$, aiming to maximize the standard deviations σ_j and probabilities p_j, $j = 1, \cdots, L$, we determine the set of optimal parameter values for the unspecified sensors with dynamics $z_{N+j,k} = d_k + n_{d,N+j,k}$ where $n_{d,N+j,k} \sim \mathcal{N}(0, \sigma_{N+j}^2)$, and the corresponding contracts C_{N+j}, $j = 1, \cdots, L$.

We define the contracts for the specified and unspecified sensors and the data fusion module as follows:

$$C_i = \left(\mathbf{G}_{[0,10]}(d \leq 300),\ \mathbf{G}_{[0,10]}\left(|n_{d,i}| - 2\right)^{[0.95]}\right)\ i = 1, 2, \cdots, N,$$

$$C_{N+j}\ (p) = \left(\mathbf{G}_{[0,10]}(d \leq 250),\ \mathbf{G}_{[0,10]}\left(|n_{d,N+j}| - 1\right)^{[p]}\right)\ j = 1, 2, \cdots, L,$$

$$C_{N+L+1} = \left(\top,\ \mathbf{G}_{[0,10]}\left(n_d = \sum_{i=1}^{N+L} \frac{n_{d,i}}{N+L}\right)\right).$$

We first assume that there exists only one unspecified sensor, i.e., L = 1, and find the optimal set of parameters (p_{N+1}, σ_{N+1}) for the unspecified sensor such that the composition of the component-level contracts robustly refines the system-level contract, i.e., $\bigotimes_{i=1}^{N} C_i \otimes C_{N+1} \otimes C_{N+2} \preceq_{\rho^*} C_{per}$ and $M_{N+1}(\sigma_{N+1}) \models_{\rho^*} C(p_{N+1})$. Figure 5 illustrates the parameter synthesis results. All the parameter values in the green boxes (SAT partitions) guarantee that the robust refinement holds. The red boxes indicate UNSAT partitions and the grey boxes indicate UNDET partitions. We report the results for a system made of one (left), 2 (center), and 3 (right) specified sensors, respectively, and one unspecified sensor. The result suggests that an accurate, but possibly expensive, sensor has to be chosen to satisfy the system requirements in C when only one specified sensor is available. As the number of available high-quality sensors increases, we can choose a cheaper sensor M_{N+1} with a large standard deviation σ_{N+1} for its error.

# of Specified Sensors, N	# of Unspecified Sensors, L (Parameters, $2L$)	Execution Time [s] # of Maximum Partitions, N_{max}				
		10	100	200	500	1,000
1	1 (2)	0.75	5.61	7.12	16.18	34.01
10	1 (2)	0.90	6.49	14.11	42.40	75.80
50	1 (2)	2.69	20.09	37.47	105.03	246.93
100	1 (2)	4.76	44.66	78.81	208.23	402.82
200	1 (2)	14.94	112.68	271.54	506.17	1,024.74
100	2 (4)	*1.75	33.60	72.40	183.98	365.81
100	3 (6)	*1.52	*36.81	83.33	191.79	406.42
100	4 (8)	*0.69	*0.75	*0.68	*144.92	384.57

Table 1. Execution time of parametric refinement for the multi-sensor perception system. When the optimal parameters could not be found before reaching the maximum number of partitions, the execution time is marked with $*$.

Algorithm 1 requires at most $2N_{max}$ MILPs or MIQCPs iterations to find the optimal parameter values. In this case study, MIQCPs are solved since the σ_{N+j} parameters lead to quadratic constraints. Table 1 reports the execution times of PyCASSE for multiple combinations of specified and unspecified sensors and N_{max}. While the execution time for solving each MIQCP problem depends on the number of parameters and sensors, the results in Table 1 demonstrate that the execution time scales linearly as N_{max} increases. We also observe that, as the number of parameters increases, finer partitions are required, hence a higher N_{max}, to determine whether a box is SAT or UNSAT.

7.2 Adaptive Cruise Control

The adaptive cruise control (ACC) system controls the ego vehicle to keep it as close as possible to a target distance $d_{target} = d_{safe} + \tau v_e$, while adapting to the leading vehicle's behavior. d_{safe} is the predetermined safe distance and τ is the time gap. Several parts of such system are intrinsically of stochastic nature, e.g., the noise of the sensors detecting the distance and velocity, and the behavior of the leading vehicle. In this case study, we illustrate the parameter synthesis process on an ACC system whose safety and comfort requirements are specified by two parametric stochastic contracts.

The dynamics of the ACC system is given by the state-space representation in (11) with:

$$A = \begin{bmatrix} 1 & \Delta t & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & \Delta t \\ 0 & 0 & 0 & 1 \end{bmatrix}, B = \begin{bmatrix} 0 \\ \Delta t \\ 0 \\ 0 \end{bmatrix}, C = \begin{bmatrix} -1 & 0 & 1 & 0 \\ 0 & -1 & 0 & 1 \\ 0 & 1 & 0 & 0 \end{bmatrix}, \quad (16)$$

where Δt is the size of each time step, $x_k = [x_{e,k}, v_{e,k}, x_{l,k}, v_{l,k}]^T$ is the state vector, $u_k = [a_{e,k}]$ is the control input, $z_k = [\hat{d}_k, \hat{v}_k, \hat{v}_{e,k}]^T$

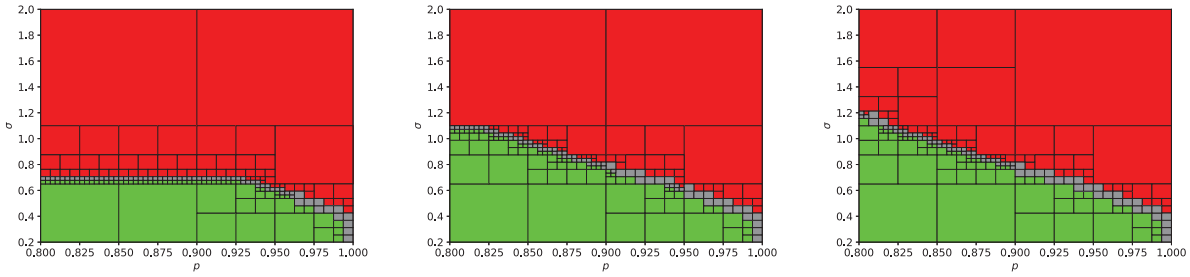

Figure 5. Results for the parameter synthesis problem under refinement $\bigotimes_{i=1}^{N} C_i \otimes C_{N+1}\,(p) \otimes C_{N+2} \preceq C$ with $N_{max} = 200$. Each figure reports on the horizontal axis the confidence level p of the unspecified sensor, and the standard deviation of the noise σ of the unspecified sensor on the vertical axis.

is the measurement vector modeling the sensor readings of d_k, v_k, and $v_{e,k}$. $w_k = [0, 0, 0, a_{l,k}]^T \sim \mathcal{N}(0, Q)$ is the process input modeling the behavior of the leading vehicle where $Var[a_{l,k}] = (\Delta t \sigma_{a_l})^2$ is the variance of the leading vehicle's acceleration; $v_k = [n_{d,k}, n_{v,k}, n_{v_e,k}]^T \sim \mathcal{N}(0, R)$ is the measurement noise where $Var[n_{d,k}] = Var[n_{v,k}] = 1^2$ and $Var[n_{v_e,k}] = 0.5^2$ for time step k. We assume that $n_{d,k}$, $n_{v,k}$, and $n_{v_e,k}$ are independent and identically distributed (i.i.d.). Finally, the ACC control policy [35, Figure 5] is given by $u_k = Dz_k + E$ where $D = \begin{bmatrix} K_I & K_P & -\tau K_I \end{bmatrix}$ and $E = \begin{bmatrix} -d_{safe} K_I \end{bmatrix}$. Larger $\sigma_{a_l}^2$ indicates more aggressive acceleration and deceleration of the leading vehicle. On the other hand, large values of K_P and K_I produce more aggressive acceleration and deceleration for the ego vehicle.

In this case study, we search for the sets of optimal parameter values $\pi_{safe} = (c_s, p_s)$ and $\pi_{comf} = (c_c, p_c)$ for two requirements expressed as the parametric stochastic contracts $C_{safe}(\pi_{safe})$ and $C_{comf}(\pi_{comf})$. C_{safe} requires that the probability of maintaining the distance d larger than or equal to c_s is greater than or equal to p_s when the initial distance is greater than or equal to d_{target} and the initial relative velocity between the ego and the leading vehicle is smaller than or equal to 5 m/s. We define the safety contract as $C_{safe} = (\phi_{A,safe}, \phi_{G,safe})$ where $\phi_{A,safe} := (d \geq d_{target}) \wedge (|v| \leq 5)$ and $\phi_{G,safe} := \mathbf{G}_{[0,20]}(c_s - d)^{[p_s]}$. On the other hand, C_{comf} requires that the acceleration of the ego vehicle be larger than or equal to c_c m/s^2 with a probability larger than or equal to p_c, to avoid abrupt decelerations [36] under the same assumptions as C_{safe}. We define the comfort contract as $C_{comf} = (\phi_{A,safe}, \phi_{G,comf})$ where $\phi_{G,comf} := \mathbf{G}_{[0,20]}(c_c - a_e)^{[p_c]}$.

Our goal is to find a set of optimal parameters such that $M_{acc} \models_{\rho^*} C_{safe}(c_s, p_s)$ and $M_{acc} \models_{\rho^*} C_{comf}(c_c, p_c)$ (Problem 3). We assume that $\Delta t = 0.5$ s, $\sigma_{a_l}^2 = 0.5$, $K_P = K_I = 0.5$, $d_{safe} = 10$ m, $\tau = 1.6$ s, $J(c_s, p_s) = -100p_s - c_s$, and $J(c_c, p_c) = -100p_c - c_c$ to maximize the probabilities of robust satisfaction while guaranteeing safety and comfort of the ACC. The larger c_s, the safer the ACC becomes; the larger c_c, the more comfortable the ride becomes for the passengers.

As shown in Figure 6, Algorithm 1 provides the optimal parameter sets $(c_s^*, p_s^*) = (5.625, 0.99375)$ and $(c_c^*, p_c^*) = (-7.5, 0.99375)$. To validate the result, we ran 10^5 simulations in MATLAB under a pre-determined initial state that satisfies the assumptions specification $\phi_{A,safe}$, i.e., $x_e = 0$ $m, v_e = 0$ $m/s, x_l = 50$ m, and $v_l = 0$ m/s. In Table 2, the rate of violation of the constraint $d \geq c_s^* = 5.625$ m in simulations is approximately equal to $1 - p_s^* = 0.00625$. The rate of violation of the constraint $a_e \geq c_c^* = -7.5$ is smaller than

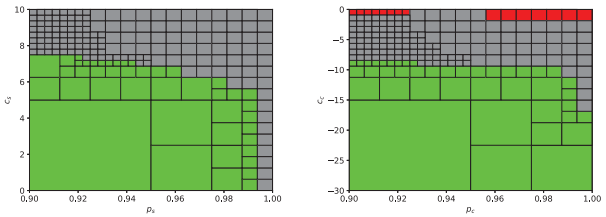

Figure 6. Parameter synthesis results for C_{safe} (left) and C_{comf} (right).

Approaches	Requirements	C_{safe}	C_{comf}
Our Approach	Execution Time [s]	433.76078	444.94791
	Violation Probability	0.00625	0.00625
10^5 Simulations	Execution Time [s]	263.18550	201.91763
	Violation Probability	0.00819	0.00546

Table 2. Results from our approach and 10^5 simulation runs.

$1 - p_c^* = 0.00625$. Solving the parameter synthesis problem takes longer time than executing 10^5 simulation runs for estimating the probability of violating C_{safe} and C_{comf}. However, PyCASSE performs exhaustive search over a range of the parameter design space and initial conditions. On the other hand, simulations are only limited to the evaluation of the requirements in a particular case, with fixed parameter values and initial state.

8 Conclusions

We presented an automated framework for quantitative verification and design space exploration of CPSs under uncertainty, leveraging the robust semantics of stochastic A/G contracts expressed in StSTL. We illustrated the effectiveness of our approach on the design and verification of a multi-sensor perception system and an ACC system. Future work includes the extension of the proposed framework to non-Gaussian processes and the investigation of adaptive partition mechanisms guided by robustness estimates to improve the scalability of our algorithms.

Acknowledgments

This research was partially supported by the National Science Foundation (NSF) under Awards 1839842, 1846524, and 2139982, the Office of Naval Research (ONR) under Award N00014-20-1-2258, the Defense Advanced Research Projects Agency (DARPA) under Award HR00112010003, and the European Union under the Marie Skłodowska-Curie grant agreement No. 894237.

References

[1] S. A. Seshia, S. Hu, W. Li, and Q. Zhu, "Design automation of cyber-physical systems: Challenges, advances, and opportunities," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 36, no. 9, pp. 1421–1434, 2017.

[2] A. Benveniste, B. Caillaud, D. Nickovic, R. Passerone, J.-B. Raclet, P. Reinkemeier, A. Sangiovanni-Vincentelli, W. Damm, T. A. Henzinger, and K. G. Larsen, "Contracts for system design," *Foundations and Trends® in Electronic Design Automation*, vol. 12, no. 2-3, pp. 124–400, 2018.

[3] P. Nuzzo, A. L. Sangiovanni-Vincentelli, D. Bresolin, L. Geretti, and T. Villa, "A platform-based design methodology with contracts and related tools for the design of cyber-physical systems," *Proceedings of the IEEE*, vol. 103, no. 11, pp. 2104–2132, Nov 2015.

[4] P. Nuzzo, M. Lora, Y. A. Feldman, and A. L. Sangiovanni-Vincentelli, "CHASE: Contract-based requirement engineering for cyber-physical system design," in *2018 Design, Automation Test in Europe Conference Exhibition (DATE)*, March 2018, pp. 839–844.

[5] A. Pnueli, "The temporal logic of programs," in *18th Annual Symposium on Foundations of Computer Science (sfcs 1977)*, 1977, pp. 46–57.

[6] O. Maler and D. Nickovic, "Monitoring temporal properties of continuous signals," in *Formal Techniques, Modelling and Analysis of Timed and Fault-Tolerant Systems*, Y. Lakhnech and S. Yovine, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2004, pp. 152–166.

[7] A. Sangiovanni-Vincentelli, W. Damm, and R. Passerone, "Taming Dr. Frankenstein: Contract-based design for cyber-physical systems," *European Journal of Control*, vol. 18, no. 3, pp. 217 – 238, 2012.

[8] P. Nuzzo, H. Xu, N. Ozay, J. B. Finn, A. L. Sangiovanni-Vincentelli, R. M. Murray, A. Donzé, and S. A. Seshia, "A contract-based methodology for aircraft electric power system design," *IEEE Access*, vol. 2, pp. 1–25, Jan 2014.

[9] P. Nuzzo, J. B. Finn, A. Iannopollo, and A. L. Sangiovanni-Vincentelli, "Contract-based design of control protocols for safety-critical cyber-physical systems," in *2014 Design, Automation Test in Europe Conference Exhibition (DATE)*, March 2014, pp. 1–4.

[10] C. Oh, E. Kang, S. Shiraishi, and P. Nuzzo, "Optimizing assume-guarantee contracts for cyber-physical system design," in *2019 Design, Automation Test in Europe Conference Exhibition (DATE)*, March 2019, pp. 246–251.

[11] J. DeCastro, L. Liebenwein, C.-I. Vasile, R. Tedrake, S. Karaman, and D. Rus, "Counterexample-guided safety contracts for autonomous driving," in *Proceedings of the 13th International Workshop on the Algorithmic Foundations of Robotics (WAFR)*, December 2018.

[12] P. Nuzzo, J. Li, A. L. Sangiovanni-Vincentelli, Y. Xi, and D. Li, "Stochastic assume-guarantee contracts for cyber-physical system design," *ACM Trans. Embed. Comput. Syst.*, vol. 18, no. 1, pp. 2:1–2:26, Jan. 2019.

[13] G. E. Fainekos and G. J. Pappas, "Robustness of temporal logic specifications for continuous-time signals," *Theor. Comput. Sci.*, vol. 410, no. 42, p. 4262–4291, Sep. 2009.

[14] A. Donzé and O. Maler, "Robust satisfaction of temporal logic over real-valued signals," in *Formal Modeling and Analysis of Timed Systems*, K. Chatterjee and T. A. Henzinger, Eds. Springer Berlin Heidelberg, 2010, pp. 92–106.

[15] A. Donzé, T. Ferrère, and O. Maler, "Efficient robust monitoring for STL," in *Computer Aided Verification*, N. Sharygina and H. Veith, Eds. Springer Berlin Heidelberg, 2013, pp. 264–279.

[16] 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, Aug. 2017.

[17] G. Welch and G. Bishop, "An introduction to the Kalman filter," TR 95-041, Tech. Rep., 2006.

[18] Y. Pei, S. Biswas, D. S. Fussell, and K. Pingali, "An elementary introduction to Kalman filtering," *Commun. ACM*, vol. 62, no. 11, p. 122–133, oct 2019.

[19] R. Durrett, *Probability: Theory and Examples*, 4th ed., ser. Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge University Press, 2010.

[20] S. Dempe, *Foundations of Bilevel Programming*. Springer New York, NY, 2002.

[21] D. Sadigh and A. Kapoor, "Safe control under uncertainty with probabilistic signal temporal logic," in *Proceedings of Robotics: Science and Systems*, ser. RSS '16, 2016.

[22] S. Jha, V. Raman, D. Sadigh, and S. A. Seshia, "Safe autonomy under perception uncertainty using chance-constrained temporal logic," *Journal of Automated Reasoning*, vol. 60, no. 1, pp. 43–62, Jan 2018.

[23] J. Li, P. Nuzzo, A. Sangiovanni-Vincentelli, Y. Xi, and D. Li, "Stochastic contracts for cyber-physical system design under probabilistic requirements," in *Proceedings of the 15th ACM-IEEE International Conference on Formal Methods and Models for System Design*, ser. MEMOCODE '17. New York, NY, USA: ACM, 2017, p. 5–14.

[24] P. Kyriakis, J. V. Deshmukh, and P. Bogdan, "Specification mining and robust design under uncertainty: A stochastic temporal logic approach," *ACM Trans. Embed. Comput. Syst.*, vol. 18, no. 5s, pp. 96:1–96:21, Oct. 2019.

[25] L. Lindemann, G. J. Pappas, and D. V. Dimarogonas, "Control barrier functions for nonholonomic systems under risk signal temporal logic specifications," in *2020 59th IEEE Conference on Decision and Control (CDC)*, 2020, pp. 1422–1428.

[26] P. Hansen, B. Jaumard, and G. Savard, "New branch-and-bound rules for linear bilevel programming," *SIAM Journal on Scientific and Statistical Computing*, vol. 13, no. 5, pp. 1194–1217, 1992.

[27] L. Vicente, G. Savard, and J. Júdice, "Descent approaches for quadratic bilevel programming," *J. Optim. Theory Appl.*, vol. 81, no. 2, p. 379–399, May 1994.

[28] A. Sinha, P. Malo, and K. Deb, "A review on bilevel optimization: From classical to evolutionary approaches and applications," *IEEE Transactions on Evolutionary Computation*, vol. 22, no. 2, pp. 276–295, 2018.

[29] J. F. Bard and J. E. Falk, "An explicit solution to the multi-level programming problem," *Computers & Operations Research*, vol. 9, no. 1, pp. 77–100, 1982.

[30] S. Gillijns and B. De Moor, "Unbiased minimum-variance input and state estimation for linear discrete-time systems," *Automatica*, vol. 43, no. 1, pp. 111–116, 2007.

[31] M. Farina, L. Giulioni, L. Magni, and R. Scattolini, "An approach to output-feedback mpc of stochastic linear discrete-time systems," *Automatica*, vol. 55, pp. 140–149, 2015.

[32] V. Raman, A. Donzé, M. Maasoumy, R. M. Murray, A. Sangiovanni-Vincentelli, and S. A. Seshia, "Model predictive control for signal temporal logic specification," 2017.

[33] R. M. Lima and I. E. Grossmann, "Computational advances in solving mixed integer linear programming problems," January 2011.

[34] S. Burer and A. N. Letchford, "Non-convex mixed-integer nonlinear programming: A survey," *Surveys in Operations Research and Management Science*, vol. 17, no. 2, pp. 97–106, 2012.

[35] M. Canale and S. Malan, "Robust design of PID based ACC S&G systems," *IFAC Proceedings Volumes*, vol. 36, no. 18, pp. 333–338, 2003, 2nd IFAC Conference on Control Systems Design (CSD '03), Bratislava, Slovak Republic, 7-10 September 2003.

[36] I. Bae, J. Moon, and J. Seo, "Toward a comfortable driving experience for a self-driving shuttle bus," *Electronics*, vol. 8, no. 9, p. 943, 2019.