Safe and Performant Control via Efficient Overapproximation of the Reachable Set Probability Distribution

Michael E. Cao and Samuel Coogan

***Abstract*— We consider a nonlinear system subject to an unknown state-dependent disturbance input and assume availability of state-dependent upper and lower bounds on the disturbance that hold with any user-prescribed probability available from, e.g., Gaussian Process estimation. Using methods from mixed monotone systems theory, we then propose an efficient technique for overbounding the probabilistic reachable set of the system for any prescribed probability. Next, we consider a reach-avoid control synthesis problem and propose using a weighted sum of reachability quantiles as the control objective to balance safety and performance. We show via a case study of a kinematic bicycle vehicle model that this approach generally outperforms using a single fixed probability bound.**

I. Introduction

Control of systems with partially unknown dynamics often requires compromising performance to satisfy safety guarantees. This is generally because worst-case disturbance behavior must be considered in order to maintain these guarantees. To achieve greater performance, it is advantageous to consider not only the worst-case disturbance but also the overall distribution of possible disturbance behavior.

For linear systems, it is possible to approximate the effects of a disturbance modeled using, e.g., a Gaussian distribution by evolving key parameters through the system and then reconstituting the distribution to maintain safety [1]. However, for nonlinear systems, especially systems that are nonlinear in the disturbance input, these nonlinearities distort the impact of the disturbance behaviors through the system. Motivated by this, we develop a method for overapproximating multiple probability levels of the reachable sets of the system. This allows us to capture the effects of the nonlinearities on the disturbance behavior and utilize them in an optimization framework for achieving greater performance while maintaining safety.

Reachability-based methods are generally well-suited for the problem of ensuring safe control of partially unknown systems. For example, [1] achieves safety via ellipsoidal overapproximations of the linearized system that account for the unknown dynamics via Gaussian Processes (GPs), and [2] utilizes Hamilton-Jacobi reachability tools to develop a general safety framework for uncertain robotic systems. Alternatively, [3] develops a linearization-based reachability overapproximation technique for systems with unknown parameters and inputs which is demonstrated in a collision avoidance scenario for autonomous vehicles in [4]. In [5], the authors use GPs to learn the unknown components of a control-affine system and form a reachability-based safety metric for reinforcement learning, and [6] utilizes Hamilton-Jacobi reachability to iteratively compute the safe set of a system towards the same end.

Additionally, there exists a body of literature concerning chance-constrained optimization that addresses similar problems. For example, [7] introduces a probabilistic resolvability condition and develops a joint chance-constrained model predictive controller that guarantees this condition for robust control of systems with unbounded stochasticity, and similarly, [8] presents a convex chance-constrained model predictive controller for polynomial, discrete-time stochastic systems. Meanwhile, [9] extends chance-constrained control techniques to handle uncertainty in both the system state and constraint parameters for linear discrete-time stochastic systems, and [10] develops a distributionally robust data-enabled predictive control algorithm for unknown stochastic linear time-invariant systems. We note that these works mainly concern the stochastic setting, while in this work we consider the deterministic unknown setting.

In contrast to these works, we consider general nonlinear systems and approximate multiple probability levels of the unknown disturbance input to compute the optimal control action that is both performant and safe. We achieve this by utilizing previously developed computationally efficient and scalable reachability techniques [11] to overapproximate the different quantiles of the reachable set distribution. These quantiles are able to capture asymmetries that arise from propagating the distribution of possible disturbance inputs through the nonlinear dynamics. Finally, we demonstrate that using these overapproximations in a model predictive control scheme leads to increased performance while maintaining safety on a case study of a kinematic bicycle operating on terrain with an unknown friction coefficient.

The rest of the paper is structured as follows: Section II introduces key notation, and Section III formally defines the problem. We provide preliminaries on mixed monotonicity in Section IV, before utilizing it to estimate the probability distribution of reachable sets in Section V. We insert this estimation into a control algorithm that balances performance and safety in Section VI and demonstrate this control algorithm with our case study in Section VII. Finally, we conclude with a short discussion in Section VIII.

M. E. Cao and S. Coogan are with the School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, 30332, USA {mcao34, sam.coogan}@gatech.edu. S. Coogan is also with the School of Civil and Environmental Engineering, Georgia Institute of Technology, Atlanta, 30332, USA. This work was supported in part by the National Science Foundation under award #1924978, the Air Force Office of Scientific Research under Grant FA9550-23-1-0303, and the NASA University Leadership Initiative under grant #80NSSC20M0161. This article solely reflects the opinions and conclusions of its authors.

II. Notation

Let (x, y) denote the vector concatenation of $x, y \in \mathbb{R}^n$, i.e., $(x, y) := [x^T \; y^T]^T \in \mathbb{R}^{2n}$. Additionally, $\preceq$ denotes the componentwise vector order, i.e., $x \preceq y$ if and only if $x_i \leq y_i$ for all $i \in \{1, ..., n\}$ where vector components are indexed via subscript.

Given $x, y \in \mathbb{R}^n$ such that $x \preceq y$, we denote the hyperrectangle defined by the endpoints x and y using the notation $[x, y] := \{z \in \mathbb{R}^n \mid x \preceq z \text{ and } z \preceq y\}$. Also, given $a = (x, y) \in \mathbb{R}^{2n}$ with $x \preceq y$, $[\![a]\!]$ denotes the hyperrectangle formed by the first and last n components of a, i.e., $[\![a]\!] := [x, y]$. Finally, let $\preceq_{\text{SE}}$ denote the *southeast order* on $\mathbb{R}^{2n}$ defined by $(x, x') \preceq_{\text{SE}} (y, y')$ if and only if $x \preceq y$ and $y' \preceq x'$. In particular, observe that when $x \preceq x'$ and $y \preceq y'$,

$$(x, x') \preceq_{\text{SE}} (y, y') \iff [y, y'] \subseteq [x, x']. \tag{1}$$

III. Problem Setup

We consider the continuous-time, nonlinear system

$$\dot{x} = f(x, u, w) \tag{2}$$

with f Lipschitz continuous in its arguments, where $x \in \mathbb{R}^n$ is the system state, $u \in \mathbb{R}^m$ is the system input, and $w \in \mathbb{R}^p$ is a deterministic unknown, state-dependent component of the dynamics such that $w_i = g_i(x)$ where g_i is unknown. We denote by $\phi(t, x_0, \pi)$ the resulting closed-loop state trajectory of (2) under control strategy $u = \pi(t, x)$ when $w = g(x)$ and the system is initialized at x_0 at time 0.

We assume that there exists a known subset of the state space $\mathcal{X}_{\text{unsafe}} \subset \mathbb{R}^n$ which must be avoided, and a goal region $\mathcal{X}_{\text{goal}} \subset \mathbb{R}^n$ which we must drive the system into. Thus, we formally define our problem statement as follows.

Problem Statement. *Consider the system* (2)*. Given known goal region* $\mathcal{X}_{\text{goal}}$*, compute a feedback control strategy* $u = \pi(t, x)$ *that reaches the goal in minimal timesteps while avoiding known unsafe region* $\mathcal{X}_{\text{unsafe}}$.

We explored a similar problem setting in [11] where we showcased an exploration-exploitation algorithm ensuring safe control for all time until the goal was reached. However, we did not previously consider performance. Thus, in this work, we develop a formulation for overapproximating multiple probability quantiles of the distribution of reachable sets, and show that accounting for this additional information in the optimization problem that solves for the next control action produces actions that are not only more aggressive and reach the goal in fewer timesteps, but that are also safe.

IV. Preliminaries on Mixed Monotonicity

In this section we provide an overview of the mixed monotonicity property of dynamical systems, which we build upon throughout the rest of this work.

The system (2) is *mixed monotone with respect to a decomposition function* δ if δ satisfies the following:

1) For all x, u, w, $\delta(x, u, w, x, w) = f(x, u, w)$
2) For all $x \preceq \widehat{x}$, u, $w \preceq \widehat{w}$, and all $a \in [x, \widehat{x}]$, $c \in [w, \widehat{w}]$,

$$\delta(x, u, w, \widehat{x}, \widehat{w}) \preceq f(a, u, c) \preceq \delta(\widehat{x}, u, \widehat{w}, x, w) \tag{3}$$

3) For all $i \in \{1, ..., n\}$,

$$\delta_i(x, u, w, \widehat{x}, \widehat{w}) \leq \delta_i(x', u, w', \widehat{x}', \widehat{w}') \tag{4}$$

for all $(x, \widehat{x}) \preceq_{\text{SE}} (x', \widehat{x}')$ such that $x_i = x_i'$, u, $(w, \widehat{w}) \preceq_{\text{SE}} (w', \widehat{w}')$

Additionally, δ is said to be *tight* if it is such that

$$\delta_i(\underline{x}, u, w, \overline{x}, \widehat{w}) = \min_{a \in [\underline{x}, \overline{x}], a_i = \underline{x}_i} \; \min_{c \in [w, \widehat{w}]} f_i(a, u, c) \tag{5}$$

$$\delta_i(\overline{x}, u, \widehat{w}, \underline{x}, w) = \max_{a \in [\underline{x}, \overline{x}], a_i = \underline{x}_i} \; \max_{c \in [w, \widehat{w}]} f_i(a, u, c). \tag{6}$$

For any system, there exists some decomposition function δ satisfying the above conditions [12], although one may not be readily available in closed form. In general, obtaining a decomposition function is problem specific, but automated tools exist for computing certain classes of decomposition functions [13]. We demonstrate construction of a decomposition function in the case study of Section VII. For more examples of decomposition functions and practical applications of mixed monotonicity, see [11], [14].

Given (2) and a corresponding decomposition function, we then construct the *embedding system* with state $(\underline{x}, \overline{x}) \in \mathbb{R}^n \times \mathbb{R}^n$, input $u \in \mathbb{R}^m$, and disturbance $(w, \widehat{w}) \in \mathbb{R}^p \times \mathbb{R}^p$ defined by the dynamics

$$\begin{bmatrix} \dot{\underline{x}} \\ \dot{\overline{x}} \end{bmatrix} = \varepsilon(\underline{x}, u, w, \overline{x}, \widehat{w}) := \begin{bmatrix} \delta(\underline{x}, u, w, \overline{x}, \widehat{w}) \\ \delta(\overline{x}, u, \widehat{w}, \underline{x}, w) \end{bmatrix}. \tag{7}$$

Denote the state of (2) at time t when initialized at x_0 under some input signal $u : [0, \infty) \to \mathbb{R}^m$ and some disturbance signal $w : [0, \infty) \to \mathbb{R}^p$ by $\phi(t; x_0, u, w)$, and denote the state of (7) at time t initialized at $(\underline{x}_0, \overline{x}_0)$ under the same input signal u, and disturbance signal $(w, \widehat{w}) : [0, \infty) \to \mathbb{R}^p \times \mathbb{R}^p$ by $\Phi^\varepsilon(t; (\underline{x}_0, \overline{x}_0), u, (w, \widehat{w}))$. The fundamental result of mixed monotone systems theory is that (7) is a monotone control system with respect to the southeast order on state and disturbance; that is, given $a, a' \in \mathbb{R}^n \times \mathbb{R}^n$ and $c, c' : [0, \infty) \to \mathbb{R}^p \times \mathbb{R}^p$ such that $a \preceq_{\text{SE}} a'$ and $c(t) \preceq_{\text{SE}} c'(t)$ for all $t \geq 0$, then for all $t \geq 0$,

$$\Phi^\varepsilon(t; a, u, c) \preceq_{\text{SE}} \Phi^\varepsilon(t; a', u, c'). \tag{8}$$

In other words, provided that the system is initialized within $[\underline{x}_0, \overline{x}_0]$ and the disturbance signal is overapproximated by $[w, \widehat{w}]$, then the hyperrectangle defined by $[\![\Phi^\varepsilon(t; (\underline{x}_0, \overline{x}_0), u, (w, \widehat{w}))]\!]$ overapproximates the true reachable set of (2), *i.e.*

$$\phi(T, x_0, u) \subseteq [\![\Phi^\varepsilon(T; (\underline{x}_0, \overline{x}_0), u, (w, \widehat{w}))]\!] \tag{9}$$

for all $T \geq 0$ and $x_0 \in [\underline{x}_0, \overline{x}_0]$.

V. OVERAPPROXIMATING THE PROBABILITY DISTRIBUTION OF REACHABLE SETS

In this section, we provide an overview of how mixed monotonicity enables efficient calculation of probabilistically correct reachable set overapproximations for systems with unknown components. We then introduce the capability of estimating probability distributions for reachable sets via nested bounding functions that hold with known probability.

If there exist known bounding functions $\underline{\gamma}_i(\underline{x},\overline{x};\rho)$ and $\overline{\gamma}_i(\underline{x},\overline{x};\rho)$, $\underline{\gamma}_i,\overline{\gamma}_i : \mathbb{R}^n \times \mathbb{R}^n \mapsto \mathbb{R}$, for all $i \in \{1,\dots,p\}$, parameterized by $\rho \in [0,1]$, such that

$$\underline{\gamma}_i(\underline{x},\overline{x};\rho) \leq g_i(x) \leq \overline{\gamma}_i(\underline{x},\overline{x};\rho), x \in [\underline{x},\overline{x}], \tag{10}$$

holds for all $\underline{x},\overline{x} \in \mathbb{R}^n$, $\underline{x} \preceq \overline{x}$ with probability at least ρ, then these functions can be inserted into the previously described embedding system to produce valid reachable set overapproximations. This is achieved by taking the embedding system (7) and inserting $\underline{\gamma}(\underline{x},\overline{x};\rho),\overline{\gamma}(\underline{x},\overline{x};\rho)$ into $w,\widehat{w}$ to produce a new embedding system parameterized by ρ as

$$\begin{bmatrix}\dot{\underline{x}}\\ \dot{\overline{x}}\end{bmatrix} = e(\underline{x},u,\overline{x};\rho) := \begin{bmatrix}\delta(\underline{x},u,\underline{\gamma}(\underline{x},\overline{x};\rho),\overline{x},\overline{\gamma}(\underline{x},\overline{x};\rho))\\ \delta(\overline{x},u,\overline{\gamma}(\underline{x},\overline{x};\rho),\underline{x},\underline{\gamma}(\underline{x},\overline{x};\rho))\end{bmatrix}. \tag{11}$$

The resulting state of (11) at time T when initialized at some initial condition $\underline{x}_0,\overline{x}_0$ is thus denoted by $\Phi^\rho(T;(\underline{x}_0,\overline{x}_0),u)$. Moving forward, we use $\underline{\gamma}^\rho,\overline{\gamma}^\rho,e^\rho$ as shorthand for the respective equations with parameter ρ. Likewise, we use $\underline{x}^\rho,\overline{x}^\rho$ to denote the resulting states of e^ρ.

A key property of $\underline{\gamma}^\rho,\overline{\gamma}^\rho$ is that, since they bound the unknown disturbance behavior over the entire state space, the probability that these functions are valid bounds directly translates to the probability that the resulting reachable set overapproximations are also valid [15]. In other words, since $\underline{\gamma}^\rho,\overline{\gamma}^\rho$ hold with probability ρ, the resulting reachable set overapproximations defined by $[\underline{x}^\rho,\overline{x}^\rho]$ also hold with probability at least ρ.

The main insight of this work is that, by considering multiple values of ρ, we obtain richer information regarding the potential future behavior of the system. We impose the following mild assumption on the construction of $\underline{\gamma}^\rho,\overline{\gamma}^\rho$:

Assumption 1. *For all $\rho_1,\rho_2 \in [0,1], \rho_1 < \rho_2$, it holds that*

$$\underline{\gamma}^{\rho_2}(\underline{x},\overline{x}) \preceq \underline{\gamma}^{\rho_1}(\underline{x},\overline{x}) \preceq \overline{\gamma}^{\rho_1}(\underline{x},\overline{x}) \preceq \overline{\gamma}^{\rho_2}(\underline{x},\overline{x}) \tag{12}$$

for all $\underline{x},\overline{x} \in \mathbb{R}^n$ with $\underline{x} \preceq \overline{x}$.

Assumption 1 states that as ρ increases, the bounding functions always expand, and as ρ decreases, the bounding functions always contract.

Remark. *We have shown in previous work [15] that it is possible to construct bounds that fulfill Assumption 1 by modeling the unknown functions $g_i(x)$ as GPs, though this is not the only method to produce valid bounding functions, as demonstrated in Section VII.*

This leads to our first result.

Proposition 1. *Consider the embedding system* (11) *with initial condition $\underline{x}_0,\overline{x}_0$. For all $\rho_1,\rho_2 \in [0,1], \rho_1 < \rho_2$,*

$$\Phi^{\rho_2}(T;(\underline{x}_0,\overline{x}_0),u) \preceq_{SE} \Phi^{\rho_1}(T;(\underline{x}_0,\overline{x}_0),u) \tag{13}$$

holds for all $T \geq 0$.

The proof of the above follows from Assumption 1 and [15, Theorem 2] and is thus omitted.

We denote by ϱ an ordered finite sequence of discrete ρ percentile values chosen for the embedding system (i.e. the (i)th component of ϱ is always less than or equal to the $(i+1)$th component). We demonstrate in Section VII that having more than one ρ, and thus including a richer approximation of the probability distribution of the unknown components, enables control strategies that increase performance while maintaining safety.

Finally, suppose the Lipschitz constants of the system are known. In that case, it is possible to overapproximate quantiles of the reachable set distribution which hold with some probability $\rho_o \notin \varrho$ without directly solving for the trajectory Φ^{ρ_o}. This is encapsulated in the following assumption and theoretical result.

Assumption 2. *The system* (2) *is Lipschitz continuous with respect to x,w with known Lipschitz constants L_x,L_w. Additionally, the bounding functions $\underline{\gamma}^\rho,\overline{\gamma}^\rho$ are Lipschitz continuous with respect to ρ with Lipschitz constant L_ρ.*

Proposition 2. *Consider* (2) *and $\underline{\gamma}^\rho,\overline{\gamma}^\rho$ fulfilling Assumptions 1 and 2. Given $\rho_o \in [\varrho_i,\varrho_{i+1}]$, and the resulting tight hyperrectangular reachable set overapproximations $[\underline{x}^{\varrho_i},\overline{x}^{\varrho_i}]$ and $[\underline{x}^{\varrho_{i+1}},\overline{x}^{\varrho_{i+1}}]$ at time $t=T$ initialized from $[\underline{x}_0,\overline{x}_0]$, the hyperrectangular reachable set overapproximation $[\underline{x}^{\rho_o},\overline{x}^{\rho_o}]$ where*

$$\underline{x}^{\rho_o} = \max\left\{\underline{x}^{\varrho_{i+1}}, \underline{x}^{\varrho_i} - \frac{L_w}{L_x}(e^{L_xT}-1)L_\rho(\rho_o-\varrho_i)\right\} \tag{14}$$

$$\overline{x}^{\rho_o} = \min\left\{\overline{x}^{\varrho_{i+1}}, \overline{x}^{\varrho_i} + \frac{L_w}{L_x}(e^{L_xT}-1)L_\rho(\rho_o-\varrho_i)\right\} \tag{15}$$

overapproximates the true reachable set of (2) *at time $t=T$ with probability at least ρ_o.*

Proof. Consider $(\underline{y}^{\rho_o},\overline{y}^{\rho_o}) = \Phi^{\rho_o}(T;(\underline{x}_0,\overline{x}_0),u)$ with δ tight. The hyperrectangle defined by $[\underline{y}^{\rho_o},\overline{y}^{\rho_o}]$ overapproximates the true reachable set of (2) with probability at least ρ_o via the properties of $\underline{\gamma}^\rho,\overline{\gamma}^\rho$.

We then consider the two possible values of $\overline{x}^{\rho_o}$. In the case where $\overline{x}^{\rho_o} = \overline{x}^{\varrho_{i+1}}$, it holds that $\overline{x}^{\rho_o} \succeq \overline{y}^{\rho_o}$ via Proposition 1 as $\varrho_{i+1} > \rho_o$. In the case where $\overline{x}^{\rho_o} = \overline{x}^{\varrho_i} + \frac{L_w}{L_x}(e^{L_xT}-1)L_\rho(\rho_o-\varrho_i)$, it holds that $\overline{x}^{\rho_o} \succeq \overline{y}^{\rho_o}$ via [16, Corollary 3.17]. Thus, it always holds that $\overline{x}^{\rho_o} \succeq \overline{y}^{\rho_o}$. Similar logic can be used to show that $\underline{x}^{\rho_o} \preceq \underline{y}^{\rho_o}$ always holds as well.

Consequently, since $[\underline{y}^{\rho_o},\overline{y}^{\rho_o}]$ overapproximates the true reachable set of (2) with probability at least ρ_o, it must hold that $[\underline{x}^{\rho_o},\overline{x}^{\rho_o}]$ overapproximates the true reachable set of (2) with probability at least ρ_o. ■

VI. A Control Algorithm to Balance Performance and Safety

We now insert the previously developed approximation of the probability distribution of the system's reachable sets into an optimization problem that, when solved, produces a control strategy that enables high performance while maintaining safety. This optimization is then utilized in a model predictive control framework to produce an algorithm that enables safe, high-performance actions for all time.

We discretize both the original system and the crafted embedding system via forward Euler and then design the following optimization problem:

$$\underset{\Pi=\{u[0],\ldots,u[D-1]\}}{\text{minimize}} \; J([\underline{x}^{\varrho}[0], \overline{x}^{\varrho}[0]], ..., [\underline{x}^{\varrho}[D], \overline{x}^{\varrho}[D]]) \quad (16)$$

subject to:

$$(11),\ [\underline{x}^{\rho}[0], \overline{x}^{\rho}[0]] = [\underline{x}[0], \overline{x}[0]] \text{ given, } u[d] \in \mathcal{U}$$
$$\forall \rho \in \varrho, \forall d \in \{0, \ldots, D-1\}$$

where the cost function is defined as

$$J([\underline{x}^{\varrho}[0], \overline{x}^{\varrho}[0]], ..., [\underline{x}^{\varrho}[D], \overline{x}^{\varrho}[D]]) := \quad (17)$$
$$\sum_{\rho\in\varrho}\sum_{d=0}^{d}\Big(a\cdot \text{dist}([\underline{x}^{\rho}[d], \overline{x}^{\rho}[d]], \mathcal{X}_{\text{goal}})$$
$$+ b\cdot\rho\cdot \text{inter}([\underline{x}^{\rho}[d], \overline{x}^{\rho}[d]], \mathcal{X}_{\text{goal}})$$
$$+ c\cdot\rho\cdot \text{inter}([\underline{x}^{\rho}[d], \overline{x}^{\rho}[d]], \mathcal{X}_{\text{unsafe}})\Big)$$

where $\text{dist}([\underline{x}^{\rho}[d], \overline{x}^{\rho}[d]], \mathcal{X}_{\text{goal}})$ is the distance between the center of each reachable set overapproximation and the center of $\mathcal{X}_{\text{goal}}$, and $\text{inter}([\underline{x}^{\rho}[d], \overline{x}^{\rho}[d]], \mathcal{X}_{\text{goal}})$ and $\text{inter}([\underline{x}^{\rho}[d], \overline{x}^{\rho}[d]], \mathcal{X}_{\text{unsafe}})$ are the area of intersection between each overapproximation and $\mathcal{X}_{\text{goal}}$ and $\mathcal{X}_{\text{unsafe}}$.

The control strategy Π that solves (16) is such that the overlap with the goal region is maximized while the overlap with the obstacle is minimized, with a bias toward driving closer to the goal region if the system is too far from it to intersect. The constants a, b, c can be tuned to prioritize performance or safety as needed. In Section VII-A, we demonstrate that a richer approximation (i.e. including multiple values of ρ in ϱ) of the full probability distribution is what enables synthesized control strategies to be both performant and safe.

We then insert the optimization problem (16) into a model predictive control scheme, where the problem is solved to synthesize a control strategy Π, the first control action is executed, and then the optimization problem is run again at the resulting state. This process repeats until the system arrives in the goal region.

We demonstrate in Section VII-B that, again, including overapproximations of multiple quantiles of the probability distribution produces control actions that are simultaneously aggressive toward the goal and safe from the obstacle. This results in the system taking fewer timesteps to reach the goal while maintaining safety, showcasing that this controller solves our problem statement.

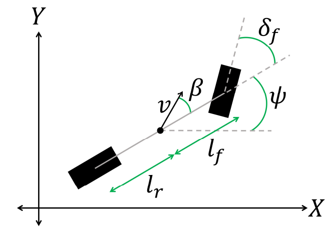

Fig. 1: The kinematic bicycle model has positions X and Y, center-of-mass velocity v, heading angle ψ, side-slip angle $\beta(\delta_f)$, front and rear distances from center of mass l_f and l_r, and inputs acceleration u_1 and steering angle $u_2 = \delta_f$.

VII. Case Studies

In this section we present results showcasing the benefit of considering a set of probabilistic reachability distributions compared to exclusively enforcing safety or exclusively assuming best-case. We provide both a Monte-Carlo simulation of a one-shot control scenario as well as a demonstration of the model predictive control formulation[1].

We consider the four-dimensional kinematic planar bicycle, which has state $x = [X, Y, \psi, v]^T$ and relates positional coordinates X and Y, center-of-mass velocity v, heading angle ψ, side-slip angle $\beta(u_2)$, and front and rear distances from center of mass $l_f = 2.2$m and $l_r = 3.3$m as

$$\dot{X} = v\cos(\psi + \beta(u_2)), \qquad \dot{Y} = v\sin(\psi+\beta(u_2)),$$
$$\dot{\psi} = \frac{v}{l_r}\sin(\beta(u_2)), \qquad \dot{v} = u_1, \quad (18)$$

where

$$\beta(u_2) = \arctan\left(\frac{l_r}{l_f + l_r}\tan(u_2)\right), \quad (19)$$

with inputs being the desired acceleration u_1 and steering angle u_2. The bicycle is subject to constraints $\mathcal{U} = [(-10, -1.5), (10, 1.5)]$ and is also affected by an unknown friction coefficient between the wheels and the surface of the road. As a result, the actual velocity update dynamics are given by $\dot{v}_{\text{actual}} = (1-\mu)u_1$, where $\mu \in [0, 1]$. This system is visualized in Figure 1.

The associated decomposition function takes the form

$$\delta(x, u, w, \widehat{x}, \widehat{w}) = \begin{bmatrix} d^X & d^Y & d^{\psi} & d^v \end{bmatrix}^T \quad (20)$$

$$d^v = d^{b_1 b_2}\left(\begin{bmatrix}1-\widehat{w}\\ u_1\end{bmatrix}, \begin{bmatrix}1-w\\ u_1\end{bmatrix}\right)$$

where, for $b, \widehat{b} \in \mathbb{R}^2$,

$$d^{b_1 b_2}(b, \widehat{b}) = \begin{cases}\min\{b_1 b_2, \widehat{b}_1 b_2, b_1\widehat{b}_2, \widehat{b}_1\widehat{b}_2\}, & \text{if } b \preceq \widehat{b}\\ \max\{b_1 b_2, \widehat{b}_1 b_2, b_1\widehat{b}_2, \widehat{b}_1\widehat{b}_2\}, & \text{if } \widehat{b} \preceq b,\end{cases}$$

and d^X, d^Y, d^{ψ} take the same forms as in [15, Equation 73]. We then construct the associated embedding system parameterized by ρ as described by (11).

[1] Code for both case studies is available at https://github.com/gtfactslab/Cao_ACC2025.

The distribution M which dictates the possible value of μ is known and follows the probability density function

$$f_M(\mu) = \begin{cases} 1.25 & \mu \in [0.2, 0.6) \\ 5 & \mu \in [0.1, 0.2) \\ 0 & \text{otherwise.} \end{cases} \tag{21}$$

which was chosen to highlight the effect of nonlinearities on the reachable sets. The bounding functions are

$$\underline{\gamma}^{\{0.25,0.5,0.75,1.0\}}(\underline{x}, \overline{x}) = \{0.175, 0.15, 0.125, 0.1\} \tag{22}$$

$$\overline{\gamma}^{\{0.25,0.5,0.75,1.0\}}(\underline{x}, \overline{x}) = \{0.3, 0.4, 0.5, 0.6\}. \tag{23}$$

The system is tasked with reaching a goal region defined by $X \in [3.9, 4.9],\ Y \in [-0.5, 0.5]$ while also avoiding the obstacle next to it defined by $X \in [5, 6],\ Y \in [-5, 5]$. We apply two different control synthesis strategies and compare their performance when given full distribution information ($\varrho = [0.25, 0.5, 0.75, 1.0]$), when only given a conservative estimation of the worst-case friction values ($\varrho = [1.0]$), and when given an overly confident underestimation of the possible friction values ($\varrho = [0.25]$). We refer to these as Full, Conservative, and Reckless, respectively.

A. One-Shot Monte Carlo

We first perform a Monte Carlo simulation to illustrate the ability of the Full strategy to increase performance while maintaining safety.

The system is randomly initialized within the hyperrectangle $[\underline{x}_0, \overline{x}_0] = [(0.5, -2.51, 0, 0), (0.51, -2.5, 0, 0)]$ and is tasked with synthesizing a one-shot input strategy that will drive the system into the goal while avoiding the obstacle. Specifically, the system is discretized using timesteps of 0.1 seconds and the one-shot control strategy will be run for $t_f = 1.5$ seconds, resulting in an overall control strategy $\Pi = \{u[0], ..., u[D-1]\}, D = 15$ obtained by solving (16) with constants $a = 0.005$, $b = -10$, and $c = 10$.

The optimization problem is implemented in Python on a personal laptop computer utilizing the `immrax` library [17], and solved using IPOPT [18]. After a solution is found for Π, 1000 Monte Carlo iterations are run by randomly determining an initial state within $[\underline{x}_0, \overline{x}_0]$ as well as drawing a value of μ from the known distribution. We then apply the synthesized strategy and count the number of instances where the system hits the goal as well as the number of times it hits the obstacle, and report these in Table I. We also report the computation time to solve each optimization problem.

Case	Hit Goal	Hit Obstacle	Comp. Time
Conservative	580	0	162 sec.
Reckless	719	256	154 sec.
Full	651	0	527 sec.

TABLE I: Results from 1000 runs in the Monte Carlo simulation.The Full strategy performs better than the Conservative strategy while maintaing the same safety level. While the Reckless strategy achieves the most goal hits, it is also the only strategy that hits the obstacle. We also note that the Full strategy, despite considering four levels of probability, takes less than four times the amount of computation time compared to the other strategies.

By considering the full distribution of possible values of μ, the solver finds a control strategy that results in a higher percentage of goals reached without decreasing the safety of the system. Additionally, in Figure 2 we showcase the differences in behavior between each level of information. With only one level of information, the solver does not allow any intersection between the reachable set overapproximations and the obstacle for the Conservative and Reckless strategies. By contrast, the Full strategy captures the nonlinear effects on the disturbance behavior, and allows the outermost reachable set overapproximations to intersect the obstacle as these represent the worst case scenario.

B. Model Predictive Control

We now implement the model predictive scheme outlined in Section VI to demonstrate that the Full strategy is able to make aggressive but safe maneuvers.

The system is initialized at $x_0 = [1.0, 2.0, 0.0, 0.0]$ and must arrive at the same goal region while avoiding the obstacle defined as the region $X \in [5, 6],\ Y \in [-10, 10]$. We instantiate ten simulation instances with differing values of μ drawn from the same probability distribution, and record the average computation time per step, the number of timesteps needed to reach the goal, and the total amount of acceleration input applied (i.e. the sum of all applied u_1). The resulting average values of each are listed in Table II.

Case	Timesteps	Total Acceleration	Step Comp. Time
Conservative	16.2	40.15	11.6 sec.
Reckless	13.9	57.38	21.0 sec.
Full	12.3	63.02	75.3 sec.

TABLE II: Average step computation time, number of timesteps to goal, and total acceleration applied from 10 runs in the MPC simulation. As shown, the Full strategy arrives at the goal in the fewest number of timesteps and is able to apply the largest amount of acceleration, while not being significantly more computationally expensive.

Overall, the Full strategy arrives at the goal in the fewest timesteps, as it is able to make large moves toward the goal while maintaining correct predictions as to avoid erratic behavior. This is especially apparent in cases like the one illustrated in Figure 3, where the value of $\mu = 0.485$ is outside the bounds utilized by the Reckless strategy. This causes the resulting states from the controller applied by the Reckless strategy to fall outside its predicted reachable sets, causing the system to miss its desired state and producing said erratic behavior. The Conservative bounds encapsulate μ, but result in smaller movements due to the size of the overapproximations. Thus, the Full strategy can make larger movements like the Reckless strategy but still encapsulates the true system behavior like the Conservative strategy.

VIII. Conclusion

We have presented a formulation to efficiently overapproximate the different probability levels of the reachable set of a system with partially unknown components. This formulation captures the effects of nonlinearities on the unknown components and reflects the resulting behavior on the distribution of reachable sets. We then developed a tunable optimization problem using these overapproximations that produces control actions that are both high performance

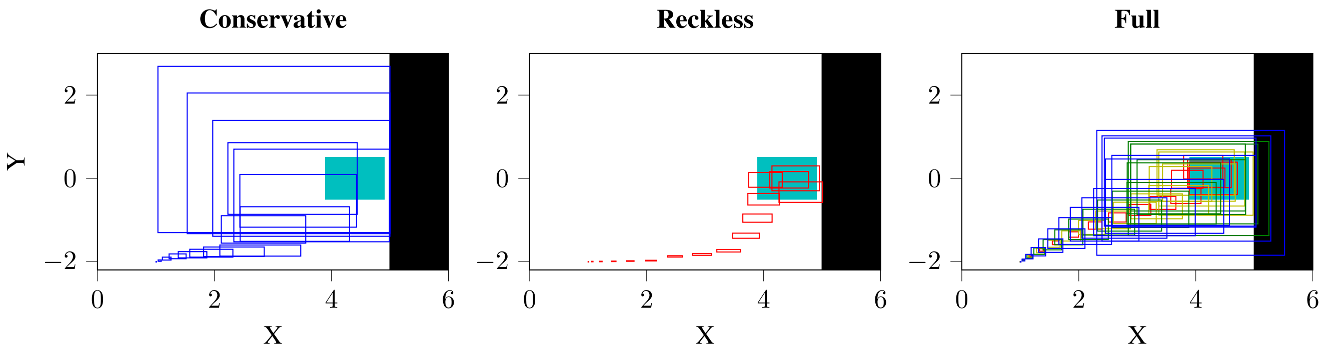

Fig. 2: Resulting control behaviors in the Monte Carlo simulation, showcasing the estimated reachable sets for each synthesized control strategy from solving (16). With only one level of information, the Conservative and Reckless strategies avoid all overlap with the obstacle. In the Conservative case, this results in a loss of performance, while in the Reckless case, this results in a loss of safety. By contrast, the Full strategy allows for some overlap between the worst-case hyperrectangles and the obstacle, preserving safety while still achieving high performance.

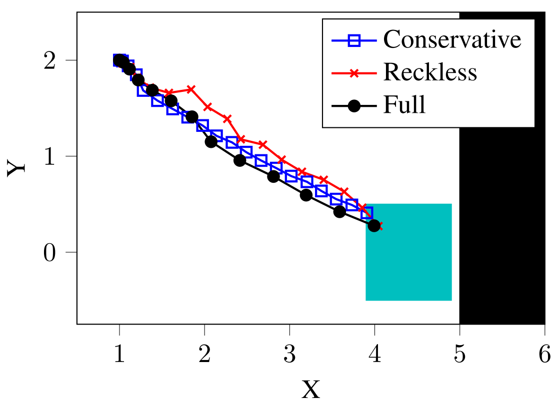

Fig. 3: Instantiation of the MPC Case Study, where the value of μ is outside the bounds that the Reckless strategy utilizes, causing its reachable set overapproximations to be incorrect and producing erratic behavior. The Conservative strategy has a smooth path, but takes longer than it needs to due to the amount of overapproximation. The Full strategy performs large movements akin to the Reckless strategy and maintains the correct predictions of the Conservative strategy.

and safe. We implemented this optimization problem into a model predictive control formulation that can produce safe, aggressive movements toward the goal region, showcased in a case study of a kinematic bicycle on a surface with an unknown friction coefficient.

References

[1] T. Koller, F. Berkenkamp, M. Turchetta, and A. Krause, "Learning-based model predictive control for safe exploration," in *2018 IEEE Conference on Decision and Control (CDC)*, pp. 6059–6066, 2018.

[2] J. F. Fisac, A. K. Akametalu, M. N. Zeilinger, S. Kaynama, J. Gillula, and C. J. Tomlin, "A general safety framework for learning-based control in uncertain robotic systems," *IEEE Transactions on Automatic Control*, vol. 64, no. 7, pp. 2737–2752, 2019.

[3] M. Althoff, O. Stursberg, and M. Buss, "Reachability analysis of nonlinear systems with uncertain parameters using conservative linearization," in *2008 47th IEEE Conference on Decision and Control*, pp. 4042–4048, 2008.

[4] M. Althoff, O. Stursberg, and M. Buss, "Model-based probabilistic collision detection in autonomous driving," *IEEE Transactions on Intelligent Transportation Systems*, vol. 10, no. 2, pp. 299–310, 2009.

[5] A. K. Akametalu, J. F. Fisac, J. H. Gillula, S. Kaynama, M. N. Zeilinger, and C. J. Tomlin, "Reachability-based safe learning with gaussian processes," in *53rd IEEE Conference on Decision and Control*, pp. 1424–1431, 2014.

[6] J. C. Shih, F. Meier, and A. Rai, "A framework for online updates to safe sets for uncertain dynamics," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 5994–6001, 2020.

[7] M. Ono, "Joint chance-constrained model predictive control with probabilistic resolvability," in *2012 American Control Conference (ACC)*, pp. 435–441, 2012.

[8] A. Jasour and C. Lagoa, "Convex chance constrained model predictive control," in *2016 IEEE 55th Conference on Decision and Control (CDC)*, pp. 6204–6209, 2016.

[9] M. P. Vitus, Z. Zhou, and C. J. Tomlin, "Stochastic control with uncertain parameters via chance constrained control," *IEEE Transactions on Automatic Control*, vol. 61, no. 10, pp. 2892–2905, 2016.

[10] J. Coulson, J. Lygeros, and F. Dörfler, "Distributionally robust chance constrained data-enabled predictive control," *IEEE Transactions on Automatic Control*, vol. 67, no. 7, pp. 3289–3304, 2022.

[11] M. E. Cao and S. Coogan, "Safe learning-based predictive control from efficient reachability," in *AACC American Control Conference (ACC)*, 2023.

[12] M. Abate, M. Dutreix, and S. Coogan, "Tight decomposition functions for continuous-time mixed-monotone systems with disturbances," *IEEE Control Systems Letters*, vol. 5, no. 1, pp. 139–144, 2021.

[13] S. Coogan, "Mixed monotonicity for reachability and safety in dynamical systems," in *2020 59th IEEE Conference on Decision and Control (CDC)*, pp. 5074–5085, 2020.

[14] M. Abate, M. Mote, M. Dor, C. Klett, S. Phillips, K. Lang, P. Tsiotras, E. Feron, and S. Coogan, "Run time assurance for spacecraft attitude control under nondeterministic assumptions," *IEEE Transactions on Control Systems Technology*, pp. 1–12, 2023.

[15] M. E. Cao, M. Bloch, and S. Coogan, "Efficient learning of hyperrectangular invariant sets using gaussian processes," *IEEE Open Journal of Control Systems*, pp. 1–14, 2022.

[16] F. Bullo, *Contraction Theory for Dynamical Systems*. Kindle Direct Publishing, 1.2 ed., 2024.

[17] A. Harapanahalli, S. Jafarpour, and S. Coogan, "immrax: A parallelizable and differentiable toolbox for interval analysis and mixed monotone reachability in jax," *IFAC-PapersOnLine*, vol. 58, no. 11, pp. 75–80, 2024. 8th IFAC Conference on Analysis and Design of Hybrid Systems ADHS 2024.

[18] A. Wächter and L. T. Biegler, "On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming," *Mathematical programming*, vol. 106, pp. 25–57, 2006.