



HAL
open science

On-the-Fly Control of Unknown Systems: From Side Information to Performance Guarantees Through Reachability

Franck Djeumou, Abraham P Vinod, Eric Goubault, Sylvie Putot, Ufuk Topcu

► **To cite this version:**

Franck Djeumou, Abraham P Vinod, Eric Goubault, Sylvie Putot, Ufuk Topcu. On-the-Fly Control of Unknown Systems: From Side Information to Performance Guarantees Through Reachability. IEEE Transactions on Automatic Control, 2023, 68, pp.4857 - 4872. 10.1109/tac.2022.3217260 . hal-04336315

HAL Id: hal-04336315

<https://polytechnique.hal.science/hal-04336315>

Submitted on 13 Dec 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

On-The-Fly Control of Unknown Systems: From Side Information to Performance Guarantees through Reachability

Franck Djeumou, Abraham P. Vinod, Eric Goubault, Sylvie Putot, and Ufuk Topcu

Abstract—We develop data-driven algorithms for reachability analysis and control of systems with a priori unknown nonlinear dynamics. The resulting algorithms not only are suitable for settings with real-time requirements but also provide provable performance guarantees. To this end, they merge noisy data from only a single finite-horizon trajectory and, if available, various forms of side information. Such side information may include knowledge of the regularity of the dynamics, algebraic constraints on the states, monotonicity, or decoupling in the dynamics between the states. Specifically, we develop two algorithms, `DaTaReach` and `DaTaControl`, to over-approximate the reachable set and design control signals for the system on the fly. `DaTaReach` constructs a differential inclusion that contains the unknown dynamics. Then, in a discrete-time setting, it over-approximates the reachable set through interval Taylor-based methods applied to systems with dynamics described as differential inclusions. We provide a bound on the time step size that ensures the correctness and termination of `DaTaReach`. `DaTaControl` enables convex-optimization-based control using the computed over-approximation and the receding-horizon control framework. Besides, `DaTaControl` achieves near-optimal control and is suitable for real-time control of such systems. We establish a bound on its suboptimality and the number of primitive operations it requires to compute control values. Then, we theoretically show that `DaTaControl` achieves tighter suboptimality bounds with an increasing amount of data and richer side information. Finally, experiments on a unicycle, quadrotor, and aircraft systems demonstrate the efficacy of both algorithms over existing approaches.

I. INTRODUCTION

Consider a scenario in which significant and unexpected changes in the dynamics of a system occur. The changes in the dynamics are such that the a priori known model cannot be used, and there is a need to learn the new dynamics on the fly. In such a scenario, the system has access to data from only its current trajectory and needs to retain a certain degree of control. This paper considers the problem of data-driven, on-the-fly control of systems with unknown nonlinear dynamics under severely limited data.

We develop data-driven algorithms, `DaTaReach` and `DaTaControl`, for the reachability analysis and control of systems with a priori unknown dynamics. `DaTaReach` exploits the data from a given and ongoing trajectory of the system to compute an over-approximation of the set of states it

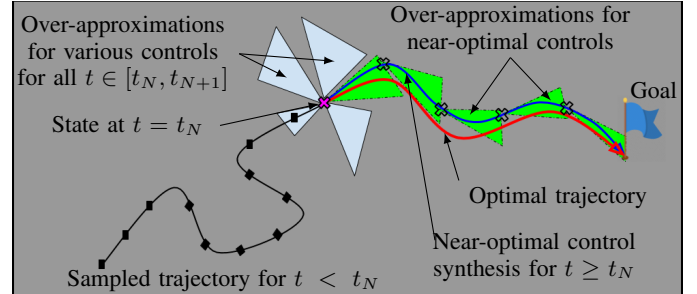


Fig. 1. We use limited data and side information for on-the-fly control of systems with unknown dynamics. At each sampling time, we compute an over-approximation of the set of states the system may reach to describe the uncertainty in its unknown trajectory and construct one-step optimal controllers via convex optimization.

may reach. `DaTaControl` incorporates the computed over-approximation into a constrained optimal control problem, which is then solved on the fly.

Specifically, `DaTaReach` and `DaTaControl` can work with possibly noisy data from only a single finite-horizon trajectory of the system and take advantage of various forms of side information on the dynamics. More specifically, the data include finite (noisy) samples of the states, the derivatives of the states, and the control signals applied. The side information may be a priori knowledge of the regularity of the dynamics, bounds on the vector field locally, monotonicity of the vector field, decoupling in the dynamics among the states, algebraic constraints on the states, or knowledge of parts of the dynamics. For example, such side information may be due to known elementary laws of physics or may be directly extracted from the data, as we illustrate in the numerical simulations.

`DaTaReach` provides closed-form expressions for over-approximations of the reachable set of unknown dynamical systems. It first uses the available data and the given side information to construct differential inclusions that contain the unknown vector field. Unlike existing work [1] that provides state-independent differential inclusions, the constructed differential inclusions account for dependencies on the states and control signals. Then, `DaTaReach` provides closed-form expressions for over-approximations of the reachable set associated with these data-driven differential inclusions. Specifically, it builds on interval Taylor-based methods [2]–[4] to over-approximate the reachable set of dynamics described by the differential inclusions. The closed-form expressions can incorporate the available side information, and increasing amount of data and richer side information provide tighter over-approximations of the reachable set.

The closed-form expressions enable convex-optimization-based, on-the-fly control of unknown dynamical systems

This material is based on work partly supported by Air Force Office of Scientific Research (FA9550-19-1-0005) and National Aeronautics and Space Administration (80NSSC19K0209).

F. Djeumou, A. Vinod, and U. Topcu are with the Department of Electrical and Computer Engineering, Oden Institute for Computational Engineering and Sciences, and the Department of Aerospace Engineering and Engineering Mechanics at the University of Texas at Austin, Austin, TX, USA. Email: {fdjeumou, utopcu}@utexas.edu, aby.vinod@gmail.com.

E. Goubault and S. Putot are with the LIX, CNRS, Ecole Polytechnique, Institut Polytechnique de Paris, France. Email: {goubault, putot}@lix.polytechnique.fr.

through the one-step receding-horizon control framework [5]. Specifically, we seek to sequentially minimize a given one-step cost function in a discrete-time setting. We refer to such an optimization problem as a *one-step optimal control* problem. The one-step cost function, which encodes the desired behavior of the system, has to be optimized in a black-box manner since it is typically a function of the unknown future state, in addition to the known current state and the currently applied control input. `DaTaControl` computes *approximate solutions* to the one-step optimal control problem through convex optimization relaxations. The convex relaxations are obtained by replacing the unknown future state with a control-affine linearization of the corresponding over-approximation of the reachable set.

More specifically, we prove that `DaTaControl` achieves near-real-time and near-optimal control of the unknown system. Indeed, we provide bounds on the suboptimality of the relaxed convex problems' solutions with respect to the optimal solutions if the dynamics were known. Then, we theoretically show that the obtained bounds on the suboptimality become tighter with an increasing amount of data and richer side information. Besides, we prove that `DaTaControl` is suitable for real-time control of the system through an explicit upper bound on the number of primitive operations required to terminate.

Empirically, through a series of experiments on a unicycle, quadrotor, and aircraft systems, we show that `DaTaControl` is significantly more computationally efficient, robust to noise, and less suboptimal than the baselines `SINDYc` [6], `DeePC` [7], `C2Opt` [8], and `CGP-LCB` [9]. Indeed, the experiments show that `DaTaControl` is *three to five* orders of magnitude faster than the baselines, achieves the control objective using at least *three times* fewer system's interactions (less suboptimal), and outperforms the baselines in terms of robustness to noise. **Contributions.** This paper significantly extends our conference paper [10] by first augmenting `DaTaControl` and `DaTaReach` with the ability to handle noise in the data. Indeed, the approach in [10] assumes exact measurements of the state and the state derivative. Then, we provide a bound on the discrete-time step size such that `DaTaReach` and `DaTaControl` are guaranteed to be correct and to terminate. The algorithms in [10] assume "sufficiently small" time step to work, without any explicit bound to characterize such small time step. Finally, we prove an explicit bound on the number of primitive operations required by `DaTaControl` to terminate and provide additional numerical experiments. With respect to state-of-the-art data-driven techniques, this paper proposes near-optimal and real-time data-driven algorithms that work with an extremely scarce amount of data and that can incorporate side information on the underlying dynamics when learning to control nonlinear systems. We provide performance guarantees such as bounds on the suboptimality and computational complexity of the algorithms. At last, through extensive comparisons with state-of-the-art baselines, we empirically demonstrate the efficiency of the proposed approach.

Related work. Several approaches for data-driven control combine system identification with model predictive control [6], [8], [11]–[13]. In [11], the authors use Koopman theory to lift the unknown nonlinear dynamics to a higher-dimensional space where they perform linear system identifi-

cation. `SINDYc` [6] utilizes a sparse regression over a library of nonlinear functions for nonlinear system identification. `DMDc` [12] uses the spectral properties of the collected data to obtain approximate linear models. Myopic control [13] uses a finite sequence of perturbations to learn a local linear model, which is then used to optimize a model-driven goodness function that encodes desirable behaviors. `DMDc`, `SINDYc`, Myopic control, and approaches based on the Koopman theory require significantly more data than the proposed approach, do not provide real-time guarantees, and, in their current forms, cannot incorporate any of the side information in this paper.

Contextual optimization-based approaches tackle the data-driven control problem via surrogate optimization and skip the system identification step [8], [9]. These approaches iteratively minimize the one-step cost function in a black-box manner using the data. `C2Opt` [8] exploits the structure in the given problem and utilizes side information (for example, smoothness) and convex optimization to solve this problem. It overcomes the drawbacks of the traditional Gaussian process-based approaches [9], namely high computational costs, expensive hyperparameter tuning, and inability to incorporate side information. Unlike `DaTaControl`, `C2Opt` considers limited forms of side information and relies on the knowledge of the gradient of the one-step cost, that may not be accessible.

Recent work [7], [14]–[18] have proposed data-driven control techniques for unknown dynamical systems that bypass the system identification step. These techniques mostly assume linear time-invariant dynamical systems and are extremely performant in such a setting. In contrast, the approach in this paper works with general unknown nonlinear dynamical systems and can incorporate side information on the underlying dynamics. Besides, on a linear version of the aircraft dynamics, we provide a comparison with `DeePC` [7], which is a traditional baseline using the same behavioral systems theory foundation [19] as the approaches in [14]–[18]. We show that `DaTaControl` is *two orders* of magnitude faster than `DeePC` while being more accurate with a low amount of data and as accurate with a high amount of data.

The authors of [20]–[22] have considered the problem of data-driven estimation of the reachable sets of partially unknown dynamical systems. The approaches in these work rely on system identification using either supervised learning algorithms [22] or Gaussian process-based algorithms [20], [21]. Such approaches are unable to take advantage of the side information, require significantly more data than `DaTaReach`, and provide only probabilistic guarantees of the correctness of the computed reachable sets. Instead, `DaTaReach` provides correct over-approximations of the reachable set at the expense of sometimes being conservative.

II. PRELIMINARIES

We denote an interval by $[a, b] = \{x \in \mathbb{R} | a \leq x \leq b\}$ for some $a, b \in \mathbb{R}$ such that $a \leq b$, the set $\{i, \dots, j\}$ by $\mathbb{N}_{[i,j]}$ for $i, j \in \mathbb{N}$ with $i \leq j$, the 2-norm by $\|\cdot\|_2$, the k^{th} component of a vector x and the (k, j) component of a matrix X by $(x)_k$ and $(X)_{k,j}$, respectively, and the Lipschitz constant of $f : \mathcal{X} \rightarrow \mathbb{R}$ by $L_f = \sup\{L \in \mathbb{R} | |f(x) - f(y)| \leq L\|x -$

$y\|_2, x, y \in \mathcal{X}, x \neq y\}$ for $\mathcal{X} \subseteq \mathbb{R}^n$. A function $f \in \mathcal{C}^k(\mathcal{X})$, referred as f is \mathcal{C}^k , with $k \geq 0$ if f is continuous on $\mathcal{X} \subseteq \mathbb{R}^n$ and all the partial derivatives of order $1, \dots, k$ exist and are continuous on \mathcal{X} , and f is *piecewise- \mathcal{C}^k* with $k \geq 0$ if there exists a partition of \mathcal{X} such that f is \mathcal{C}^k on each set in the partition. Comparisons (e.g., \geq) between matrices or vectors are conducted elementwise.

A. Interval Analysis

We denote the set of intervals on \mathbb{R} by $\mathbb{IR} = \{\mathcal{A} = [\underline{\mathcal{A}}, \overline{\mathcal{A}}] \mid \underline{\mathcal{A}}, \overline{\mathcal{A}} \in \mathbb{R}, \underline{\mathcal{A}} \leq \overline{\mathcal{A}}\}$, the set of n -dimensional interval vectors by \mathbb{IR}^n , and the set of $n \times m$ -dimensional interval matrices by $\mathbb{IR}^{n \times m}$. We carry forward the definitions [23] of arithmetic operations, set inclusion, and intersections of intervals to interval vectors and matrices by applying them componentwise. We denote the absolute value of an interval \mathcal{A} by $|\mathcal{A}| = \max\{|\underline{\mathcal{A}}|, |\overline{\mathcal{A}}|\}$, the infinity norm of $\mathcal{A} \in \mathbb{IR}^n$ by $\|\mathcal{A}\|_\infty = \sup_{i \in \mathbb{N}_{[1, n]}} |(\mathcal{A})_i|$, the Cartesian product between intervals by \otimes , i.e., $\mathcal{A} \otimes \mathcal{B} = [[\underline{\mathcal{A}}, \overline{\mathcal{A}}], [\underline{\mathcal{B}}, \overline{\mathcal{B}}]] \in \mathbb{IR}^2$ for any two intervals $\mathcal{A}, \mathcal{B} \in \mathbb{IR}$. We define \mathcal{A}^n as the Cartesian product of any interval $\mathcal{A} \in \mathbb{IR}$ with itself n times. We use the term interval to specify an interval vector or interval matrix when it is clear from the context.

Given $f : \mathcal{X} \mapsto \mathcal{Y}$ with $\mathcal{X} \subseteq \mathbb{R}^n$ and $\mathcal{Y} \subseteq \mathbb{R}^m$ (or $\mathcal{Y} \subseteq \mathbb{R}^{n \times m}$), we define an *interval extension* of f as an interval-valued function $\mathbf{f} : \mathbb{IR}^n \mapsto \mathbb{IR}^m$ such that

$$\mathbf{f}(\mathcal{A}) \supseteq \mathcal{R}(f, \mathcal{A}) = \{f(x) \mid x \in \mathcal{A}\}, \quad \forall \mathcal{A} \subseteq \mathcal{X}. \quad (1)$$

Thus, given an interval \mathcal{A} , $\mathbf{f}(\mathcal{A})$ is an interval that over-approximates the range of values taken by f over \mathcal{A} . We denote interval-valued vector functions via bold lowercase and interval-valued matrix functions via bold uppercase symbols.

Example 1 (INTERVAL EXTENSION OF 2-NORM). *Consider $f = \|\cdot\|_2$. We compute its interval extension \mathbf{f} via interval extensions of $\alpha = \sqrt{\cdot}$ and $\beta = (\cdot)^2$. For any $\mathcal{A} = [\underline{\mathcal{A}}, \overline{\mathcal{A}}] \in \mathbb{IR}$,*

$$\alpha(\mathcal{A}) = [\sqrt{\underline{\mathcal{A}}}, \sqrt{\overline{\mathcal{A}}}], \quad \text{if } \mathcal{A} \geq 0, \quad (2)$$

$$\beta(\mathcal{A}) = \begin{cases} [0, \max\{\underline{\mathcal{A}}^2, \overline{\mathcal{A}}^2\}], & \text{if } 0 \in \mathcal{A} \\ [\min\{\underline{\mathcal{A}}^2, \overline{\mathcal{A}}^2\}, \max\{\underline{\mathcal{A}}^2, \overline{\mathcal{A}}^2\}], & \text{otherwise.} \end{cases} \quad (3)$$

Using (2) and (3) with interval arithmetic, we have, for any $\mathcal{S} = [\mathcal{S}_1, \dots, \mathcal{S}_n] \in \mathbb{IR}^n$, $\mathbf{f}(\mathcal{S}) = \alpha(\sum_{i=1}^n \beta(\mathcal{S}_i))$.

B. Over-Approximations of the Reachable Set

Consider a nonlinear dynamical system,

$$\dot{x} = h(x, u), \quad (4)$$

where the state $x : \mathbb{R}_+ \mapsto \mathcal{X}$ is a continuous-time signal evolving in $\mathcal{X} \in \mathbb{IR}^n$, the control $u \in \mathbb{U}$ is a signal of time evolving in the control set $\mathcal{U} \in \mathbb{IR}^m$ with $\mathbb{U} = \{v : \mathbb{R}_+ \mapsto \mathcal{U} \mid v \text{ is piecewise-}\mathcal{C}^{D_u}\}$ for $D_u \geq 0$, and the function $h : \mathcal{X} \times \mathcal{U} \mapsto \mathcal{Y}$ is \mathcal{C}^{D_h} for some $D_h \geq 1$ and $\mathcal{Y} \subseteq \mathbb{R}^n$.

Definition 1 (TRAJECTORY OF THE SYSTEM). *Given an initial state $x_i = x(t_i)$ at time t_i and a control signal $u \in \mathbb{U}$, a trajectory of (4) is a continuous function of time $x(\cdot; x_i, u) : [t_i, \infty[\mapsto \mathcal{X}$ that satisfies (4).*

We are interested in the set of states reachable by trajectories of the system when the initial state and the control signal are uncertain quantities. We call such a set the reachable set of the system, and we define it as follows.

Definition 2 (REACHABLE SET). *Given a set $\mathcal{I}_i \subseteq \mathcal{X}$ of states at time t_i and a set $\mathbb{V} \subseteq \mathbb{U}$ of control signals, the reachable set of the dynamics (4) at time $t \geq t_i$ is given by $\mathcal{R}(t, \mathcal{I}_i, \mathbb{V}) = \{z \in \mathcal{X} \mid \exists x_i \in \mathcal{I}_i, \exists v \in \mathbb{V}, z = x(t; x_i, v)\}$.*

Given a set $\mathbb{V} \subseteq \mathbb{U}$ and a set $\mathcal{I}_0 \subseteq \mathcal{X}$ of states at time t_0 , we compute over-approximations of $\mathcal{R}(t, \mathcal{I}_0, \mathbb{V})$ at time $t \geq t_0$ using interval Taylor-based methods [3], [23], [24]. Specifically, we consider a time grid $t_0 < \dots < t_N$ such that for all $v \in \mathbb{V}$, v is \mathcal{C}^{D_u} on each interval $[t_i, t_{i+1}[$. We want to compute sets $\mathcal{R}_i^+ \in \mathbb{IR}^n$ (with $\mathcal{R}_0^+ = \mathcal{I}_0$) such that for all $i \in \mathbb{N}_{[0, N-1]}$, $\mathcal{R}_{i+1} = \mathcal{R}(t_{i+1}, \mathcal{R}_i, \mathbb{V}) \subseteq \mathcal{R}_{i+1}^+$. First, interval arithmetic enables to inductively define $\mathbf{h}^{[d]}$, the interval extensions of the Taylor coefficients $h^{[d]}$ given by

$$h^{[1]} = h, \quad h^{[d+1]} = \frac{1}{d+1} \left(\frac{\partial h^{[d]}}{\partial x} h + \sum_{l=0}^{d-1} \frac{\partial h^{[d]}}{\partial u^{(l)}} u^{(l+1)} \right), \quad (5)$$

where $u^{(l)}$ is the l^{th} derivative of u and $h^{[d]}$ is a function of higher order derivatives of h . Next, we start with $\mathcal{R}_0^+ = \mathcal{I}_0$, and compute the over-approximations $\{\mathcal{R}_i^+\}_{i=1}^N$ iteratively as

$$\begin{aligned} \mathcal{R}_{i+1}^+ = \mathcal{R}_i^+ + \sum_{d=1}^{D-1} (t_{i+1} - t_i)^d (\mathbf{h}^{[d]}(\mathcal{R}_i^+, \mathbf{v}))(t_i) \\ + (t_{i+1} - t_i)^D (\mathbf{h}^{[D]}(\mathcal{S}_i, \mathbf{v}))([t_i, t_{i+1}]), \end{aligned} \quad (6)$$

where $D \leq \min(D_u + 1, D_h)$ is the order of the Taylor expansion, we denote $\mathbf{v}^{(0)}(\mathcal{A})$ by $\mathbf{v}(\mathcal{A})$ and the intervals $\mathbf{v}^{(d)}(\mathcal{A})$ for all $d \in \mathbb{N}_{[0, D_u]}$ are such that $\cup_{v \in \mathbb{V}} \mathcal{R}(v^{(d)}, \mathcal{A}) \subseteq \mathbf{v}^{(d)}(\mathcal{A})$ with $\mathcal{A} \subseteq \mathbb{R}_+$. In other words, the interval $\mathbf{v}^{(d)}(\mathcal{A})$ over-approximates the range of the d^{th} derivative of all functions $v \in \mathbb{V}$ on the interval \mathcal{A} . Note that $\mathbf{v}^{(d)}(\mathcal{A})$ is used in the computation of $\mathbf{h}^{[p]}$ (5) for all $p \in \mathbb{N}_{[d+1, D]}$ with $\mathcal{A} = t_i$ and $\mathcal{A} = [t_i, t_{i+1}]$. Here, the set $\mathcal{S}_i \subseteq \mathcal{X}$ is an *a priori rough enclosure* of $\mathcal{R}(\cdot, \mathcal{R}_i^+, \mathbb{V})$ on $[t_i, t_{i+1}]$ and is a solution of

$$\mathcal{R}_i^+ + [0, t_{i+1} - t_i] \mathcal{R}(h, \mathcal{S}_i \times \mathbf{v}([t_i, t_{i+1}])) \subseteq \mathcal{S}_i. \quad (7)$$

Since it is a hard problem to compute exactly the range $\mathcal{R}(h, \mathcal{S}_i \times \mathbf{v}([t_i, t_{i+1}]))$, we instead compute the a priori rough enclosure \mathcal{S}_i recursively by solving

$$\mathcal{R}_i^+ + [0, t_{i+1} - t_i] \mathbf{h}(\mathcal{S}_i, \mathbf{v}([t_i, t_{i+1}])) \subseteq \mathcal{S}_i, \quad (8)$$

where \mathbf{h} is an interval extension of the known function h . Note that a solution \mathcal{S}_i of (8) is also a solution of (7) by definition of the interval extension \mathbf{h} . The existence of the a priori rough enclosure \mathcal{S}_i is guaranteed for a sufficiently small time step size $t_{i+1} - t_i$. Hence, picking adequately the time step size is primordial in order to find \mathcal{S}_i in a finite number of iterations when solving the fixed-point equation (8).

C. One-Step Optimal Control

Constrained receding-horizon control [5] is a form of constrained control in which, at each sampling time, we obtain the current control input by solving a finite-horizon open-loop optimal control problem, using the current state of the

system as the initial state. Consider a system with dynamics in the form (4), the desired behavior of the system is generally encoded as a *one-step cost function* $c : \mathcal{X} \times \mathcal{U} \times \mathcal{X} \rightarrow \mathbb{R}$ that assigns preferences over the tuple of the current state, the current control input, and the corresponding state at the next time step. Given a constant time step size $\Delta t \geq 0$, we define the next state $x_{p+1} = x(t_{p+1}; x_p, u_p)$ as the value of the trajectory of the system at time $t_{p+1} = t_p + \Delta t$, given the initial state $x_p \in \mathcal{X}$ at time t_p and the constant control signal $u_p \in \mathcal{U}$. Therefore, the finite horizon open-loop optimal control problem at a state $x_i \in \mathcal{X}$ at time t_i is given by

$$\underset{u_i, \dots, u_{i+N-1} \in \mathcal{U}}{\text{minimize}} \quad \sum_{p=i}^{i+N-1} c(x_p, u_p, x_{p+1}), \quad (9)$$

where $N \geq 1$ is the planning horizon of the control problem. Note that even if the cost function c is convex, the optimization problem (9) is generally nonconvex since the next state x_{p+1} is possibly highly nonlinear in the control variable u_p due to the nonlinear dynamics function h .

In the setting where the dynamics of the system are unknown, we cannot compute its trajectories, and $c(x_p, u_p, x_{p+1})$ is therefore unknown. Hence, the optimization problem (9) is similar to the problem of optimizing an a priori unknown function. In such a setting, the inability to accurately predict the next states of the system motivates using the short planning horizon $N = 1$. Thus, at each time t_i , we solve the *one-step optimal control problem* [8], [25], [26] given by

$$\underset{u_i \in \mathcal{U}}{\text{minimize}} \quad c(x_i, u_i, x_{i+1}), \quad (10)$$

where x_i is the state of the system at t_i , x_{i+1} is the unknown next state at time $t_{i+1} = t_i + \Delta t$, and Δt is the constant time step size. The reduction to a one-step optimal control problem greatly improves the tractability of the problem of controlling the unknown dynamical system [25].

III. PROBLEM STATEMENT

In this paper, we consider *control-affine* nonlinear systems,

$$\dot{x} = f(x) + G(x)u, \quad (11)$$

where $f : \mathcal{X} \mapsto \mathbb{R}^n$ and $G : \mathcal{X} \mapsto \mathbb{R}^{n \times m}$ are *unknown* vector-valued and matrix-valued functions, respectively, for $\mathcal{X} \in \mathbb{R}^n$. Note that even though we consider control-affine dynamics, in the general case, we can construct a control-affine model of the system locally and apply the results of the paper.

Assumption 1 (LIPSCHITZ SYSTEM). *f and G are locally Lipschitz-continuous functions. That is, each component of f and G has a finite Lipschitz constant on each subset of \mathbb{R}^n .*

Assumption 1 is common in the frameworks of reachability analysis and receding-horizon control. Since the domain $\mathcal{X} \in \mathbb{R}^n$ is bounded, f and G are globally Lipschitz-continuous on \mathcal{X} . We exploit such global Lipschitz continuity on \mathcal{X} by assuming known upper bounds on the Lipschitz constants. That is, we have access to $L_f \in \mathbb{R}_+^n$ and $L_G \in \mathbb{R}_+^{n \times m}$ with $(L_f)_k = L_{f_k}$ and $(L_G)_{k,l} = L_{G_{k,l}}$ as known upper bounds on the Lipschitz constants of $(f)_k$ and $(G)_{k,l}$ for all $k \in \mathbb{N}_{[1,n]}$ and $l \in \mathbb{N}_{[1,m]}$. We emphasize that the Lipschitz bounds can

be directly estimated from data at the expense of weakening some of the guarantees in this paper.

Let $N \in \mathbb{N}$, $N \geq 1$. Let $\mathcal{T}_N = \{(\tilde{x}_i, \dot{\tilde{x}}_i, u_i)\}_{i=1}^N$ denote a single finite-horizon trajectory containing N noisy samples of the exact state $x_i = x(t_i)$, the derivative $\dot{x}_i = \dot{x}(t_i)$ of the state, and the control signal $u_i = u(t_i)$ from a trajectory of the system. We build on the widely-used *bounded noise assumption* and consider that the following bounds hold:

$$|x(t) - \tilde{x}(t)| \leq \eta, \quad |\dot{x}(t) - \dot{\tilde{x}}(t)| \leq \bar{\eta}, \quad (12)$$

for all $t \in \mathbb{R}_+$ and for some vector values $\eta, \bar{\eta} \in \mathbb{R}_+^n$. Here the absolute value and the comparison are conducted elementwise.

Given the current noisy measurements of the state $\tilde{x}_{N+1} = \tilde{x}(t_{N+1})$ and the trajectory \mathcal{T}_N , we first seek to over-approximate the reachable set of the system.

Problem 1 (REACHABLE SET OVER-APPROXIMATION). *Given a set $\mathbb{V} \subseteq \mathbb{U}$ of admissible control signals, a time step size $\Delta t > 0$, and a maximum number $T > N$ of time steps, compute an over-approximation of the reachable set at time $t_i = t_N + (i - N)\Delta t$ for all $i \in \mathbb{N}_{[N+1,T]}$.*

Next, we seek to control the system by computing approximate solutions to the one-step optimal control problem (10).

Problem 2 (APPROXIMATE ONE-STEP OPTIMAL CONTROL). *Given a single finite-horizon trajectory \mathcal{T}_{i-1} for some $i \in \mathbb{N}$, $i > N$, the current measurement \tilde{x}_i at time t_i , compute, at sampling time t_i , approximate solutions of the one-step optimal control problem (10) and characterize the suboptimality of such solutions with respect to the solutions of the optimal control problem if the dynamics were known.*

Example 2 (UNICYCLE SYSTEM). *Consider, as a running example, the unicycle system with dynamics given by*

$$\dot{p}_x = v \cos(\theta), \quad \dot{p}_y = v \sin(\theta), \quad \dot{\theta} = \omega, \quad (13)$$

where the components of the state $x = [p_x, p_y, \theta]$ represent, respectively, the position in the x plane, the y plane, and the heading of the unicycle. The components of the control $u = [v, \omega]$ represent the speed and the turning rate, respectively. We consider the constraint set $\mathcal{U} = [-3, 3] \times [-\pi, \pi]$. In the control-affine form (11), we have $f = 0$, $(G)_{1,1} = \cos(\theta)$, $(G)_{2,1} = \sin(\theta)$, $(G)_{3,2} = 1$, and $(G)_{k,l} = 0$ otherwise. We assume the dynamics (13) are unknown and are given the loose Lipschitz bounds $L_f = [0.01, 0.01, 0.01]$, $L_{G_{1,1}} = L_{G_{2,1}} = 1.1$, $L_{G_{3,2}} = 0.1$, and $L_{G_{k,l}} = 0$ otherwise. These bounds encode the knowledge about the influence of the current state and control signals on the unicycle dynamics. For example, since p_x and p_y are unchanged when no velocity is applied, we can deduce that $(G)_{1,2} = (G)_{2,2} = 0$ and therefore $L_{G_{1,2}} = L_{G_{2,2}} = 0$. The same reasoning allows to say that $f = 0$, but we do not use such knowledge. Furthermore, we consider the knowledge that the vector field does not depend on its positions. That is, $f(x) = f(\theta)$ and $G(x) = G(\theta)$.

IV. OVER-APPROXIMATIONS OF THE REACHABLE SET OF UNKNOWN DYNAMICAL SYSTEMS

We develop DaTaReach to address Problem 1. It constructs a differential inclusion that contains the unknown vec-

tor field. Then, it utilizes the interval Taylor-based method to over-approximate the reachable set of dynamics described by the constructed differential inclusion. We also provide a bound on the time step size that guarantees that an a priori rough enclosure exists. Henceforth, the bound enables to ensure the correctness and termination of DaTaReach.

A. Differential Inclusion based on Lipschitz Continuity

First, to aid the construction of the differential inclusion, we over-approximate f and G at each data point of \mathcal{T}_N .

Lemma 1 (CONTRACTION VIA DATA). *Given a data point $(\tilde{x}_i, \tilde{\dot{x}}_i, u_i)$, an interval $\mathcal{F}_i \in \mathbb{IR}^n$ such that $f(\tilde{x}_i) \in \mathcal{F}_i$, and an interval $\mathcal{G}_i \in \mathbb{IR}^{n \times m}$ such that $G(\tilde{x}_i) \in \mathcal{G}_i$, the intervals $C_{\mathcal{F}_i}$ and $C_{\mathcal{G}_i}$, defined sequentially for $l = 1, \dots, m$ by*

$$\begin{aligned} (C_{\mathcal{F}_i})_k &= (\mathcal{F}_i)_k \cap ([\tilde{x}_i - \bar{\eta}, \tilde{x}_i + \bar{\eta}] - \mathcal{G}_i u_i)_k, \\ (s_0)_k &= ([\tilde{x}_i - \bar{\eta}, \tilde{x}_i + \bar{\eta}] - C_{\mathcal{F}_i})_k \cap (\mathcal{G}_i u_i)_k, \\ (C_{\mathcal{G}_i})_{k,l} &= \begin{cases} (((s_{l-1})_k - \sum_{p>l} (\mathcal{G}_i)_{k,p}(u_i)_p) \cap ((\mathcal{G}_i)_{k,l} u_i^l)) \frac{1}{u_i^l}, \\ \text{if } u_i^l = (u_i)_l \neq 0 \\ (\mathcal{G}_i)_{k,l}, & \text{otherwise,} \end{cases} \\ (s_l)_k &= ((s_{l-1})_k - (C_{\mathcal{G}_i})_{k,l}(u_i)_l) \cap (\sum_{p>l} (\mathcal{G}_i)_{k,p}(u_i)_p), \end{aligned} \quad (14)$$

for all $k \in \mathbb{N}_{[1,n]}$, are the smallest intervals enclosing $f(\tilde{x}_i)$ and $G(\tilde{x}_i)$, respectively, given only the data point, \mathcal{F}_i , and \mathcal{G}_i .

The proof for Lemma 1 is derived exactly as in our previous work [10] for noiseless data, where we replace the unknown exact derivative \dot{x}_i with the uncertain quantities $\tilde{\dot{x}}_i + [-\bar{\eta}, \bar{\eta}]$ and propagate the result with interval arithmetic. Intuitively, we use the constraint $\dot{\tilde{x}}_i = f(\tilde{x}_i) + G(\tilde{x}_i)u_i$, where $|\dot{\tilde{x}}_i - \tilde{\dot{x}}_i| \leq \bar{\eta}$, to remove from \mathcal{F}_i and \mathcal{G}_i some values of $f(\tilde{x}_i)$ and $G(\tilde{x}_i)$ that do not satisfy the constraint. Note that, in practise, $|\dot{\tilde{x}}_i - \tilde{\dot{x}}_i| \leq \bar{\eta}$ holds since measurements of the state derivative are usually obtained from derivatives of the noisy measurements of the state \tilde{x}_i . We also provide an illustration of Lemma 1 in Appendix A for the unicycle system.

Next, we provide functions to globally over-approximate f and G , via uncertain knowledge of f and G at the data points.

Lemma 2 (OVER-APPROXIMATION OF f AND G). *Given a set $\mathcal{E}_N = \{(\tilde{x}_i, C_{\mathcal{F}_i}, C_{\mathcal{G}_i}) \mid f(\tilde{x}_i) \in C_{\mathcal{F}_i}, G(\tilde{x}_i) \in C_{\mathcal{G}_i}\}_{i=0}^N$ and the bounds L_f and L_G , the functions $\mathbf{f} : \mathcal{X} \rightarrow \mathbb{IR}^n$ and $\mathbf{G} : \mathcal{X} \rightarrow \mathbb{IR}^{n \times m}$, given for all $k \in \mathbb{N}_{[1,n]}$ and $l \in \mathbb{N}_{[1,m]}$ by*

$$\begin{aligned} (\mathbf{f}(x))_k &= \bigcap_{(\tilde{x}_i, C_{\mathcal{F}_i}, \cdot) \in \mathcal{E}_N} (C_{\mathcal{F}_i})_k + L_{f_k} \|x - \tilde{x}_i\|_2 [-1, 1], \\ (\mathbf{G}(x))_{k,l} &= \bigcap_{(\tilde{x}_i, \cdot, C_{\mathcal{G}_i}) \in \mathcal{E}_N} (C_{\mathcal{G}_i})_{k,l} + L_{G_{k,l}} \|x - \tilde{x}_i\|_2 [-1, 1], \end{aligned} \quad (15)$$

are such that $f(x) \in \mathbf{f}(x)$ and $G(x) \in \mathbf{G}(x)$ for all $x \in \mathcal{X}$.

We provide a proof for Lemma 2 in our previous work [10], where we use interval arithmetic with the definition of the upper bounds on the Lipschitz constants to over-approximate the unknown functions f and G . Note that interval extensions of \mathbf{f} and \mathbf{G} can be obtained by replacing occurrences of $\|\cdot\|_2$

Algorithm 1 Optimal over-approximation of the values of f and G at each data point of a finite-horizon trajectory.

Input: Single trajectory \mathcal{T}_N , sufficiently large $M > 0$, upper bounds on the Lipschitz constants, noise bound $\bar{\eta}$.
Output: $\mathcal{E}_N = \{(\tilde{x}_i, C_{\mathcal{F}_i}, C_{\mathcal{G}_i}) \mid f(\tilde{x}_i) \in C_{\mathcal{F}_i}, G(\tilde{x}_i) \in C_{\mathcal{G}_i}\}_{i=0}^N$
1: $\mathcal{A} \leftarrow \mathcal{X}$, $\mathcal{R}^{f\mathcal{A}} \leftarrow [-M, M]^n$, $\mathcal{R}^{G\mathcal{A}} \leftarrow [-M, M]^{n \times m}$
2: Define $\tilde{x}_0 \in \mathcal{A}$, $C_{\mathcal{F}_0} \leftarrow \mathcal{R}^{f\mathcal{A}}$, and $C_{\mathcal{G}_0} \leftarrow \mathcal{R}^{G\mathcal{A}}$
3: **for** $i \in \mathbb{N}_{[1,N]} \wedge (\tilde{x}_i, \tilde{\dot{x}}_i, u_i) \in \mathcal{T}_N$ **do**
4: Compute $\mathcal{F}_i = \mathbf{f}(\tilde{x}_i)$, $\mathcal{G}_i = \mathbf{G}(\tilde{x}_i)$ via (15) and \mathcal{E}_{i-1}
5: Compute $C_{\mathcal{F}_i}, C_{\mathcal{G}_i}$ via (14), $\mathcal{F}_i, \mathcal{G}_i$, and $(\tilde{x}_i, \tilde{\dot{x}}_i, u_i)$
6: **end for**
7: **do**
8: Execute lines 3–6 with \mathcal{E}_N instead of \mathcal{E}_{i-1} on line 4
9: **while** \mathcal{E}_N is not invariant
10: **return** \mathcal{E}_N

with its interval extension given in Example 1. In the rest of the paper, when the input of \mathbf{f} and \mathbf{G} are intervals, we use such interval extensions.

Finally, we develop Algorithm 1 that utilizes the trajectory \mathcal{T}_N to compute the set \mathcal{E}_N required in Lemma 2 to over-approximate f and G . Then, we construct a differential inclusion containing the unknown vector field.

Theorem 1 (DIFFERENTIAL INCLUSION). *Given a trajectory \mathcal{T}_N , the bounds L_f and L_G , and the noise bound $\bar{\eta}$, the dynamics (11) are contained in the differential inclusion*

$$\dot{x} \in \mathbf{f}(x) + \mathbf{G}(x)u, \quad (16)$$

where the functions $\mathbf{f} : \mathcal{X} \rightarrow \mathbb{IR}^n$ and $\mathbf{G} : \mathcal{X} \rightarrow \mathbb{IR}^{n \times m}$ are obtained by (15) with \mathcal{E}_N being the output of Algorithm 1.

A proof of Theorem 1 is provided in our previous work [10], where we show that the output \mathcal{E}_N of Algorithm 1 is such that for all $(\tilde{x}_i, C_{\mathcal{F}_i}, C_{\mathcal{G}_i}) \in \mathcal{E}_N$, $f(\tilde{x}_i) \in C_{\mathcal{F}_i}$ and $G(\tilde{x}_i) \in C_{\mathcal{G}_i}$. Therefore, Lemma 2 enables to conclude.

Figure 2 shows that the differential inclusion holds on the unicycle system using a randomly generated trajectory \mathcal{T}_{15} .

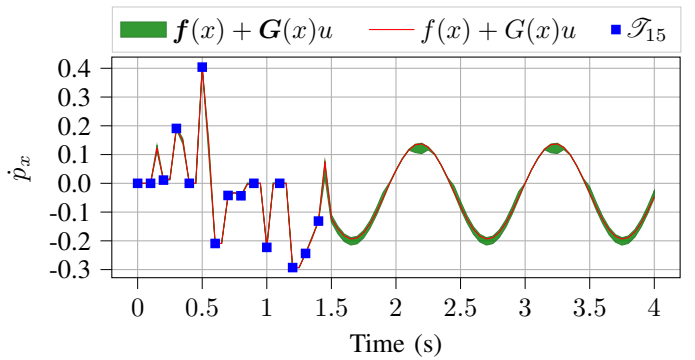


Fig. 2. Time evolution of p_x and its over-approximation $(\mathbf{f}(x) + \mathbf{G}(x)u)_1$ for the unicycle system of Example 2 in a noiseless setting. We generate the trajectory corresponding to $\dot{x}(t)$ using a randomly generated piecewise-constant input $u(t) \in \mathcal{U}$ for $t \leq t_{15} = 1.5$ s, and $u(t) = [1, \cos(6(t - t_{15}))]$ for $t \in [t_{15}, 4]$.

Remark 1 (PERSISTENT EXCITATION). *The quality of the differential inclusion of Theorem 1 depends on how much information on f and G can be obtained from \mathcal{T}_N . Thus,*

if the trajectory is not diverse, it is likely impossible to obtain tight differential inclusions. For example, consider one of the corner cases where $u_i = 0$ for every data point in \mathcal{T}_N . In this case, it is impossible to retrieve any information on G . When the goal is to control the system, control values can be synthesized to diversify the trajectory and obtain tight differential inclusions that help future control decisions. For example, excitation-based control (i.e., control inputs that are either zero or nonzero only on a single axis) can be performed to learn the dynamics.

B. Bound on the Time Step Size for the Existence of an a Priori Rough Enclosure

In this section, we consider the problem of finding an a priori rough enclosure \mathcal{S}_i solution of the fixed-point equation

$$\mathcal{R}_i^+ + [0, \Delta t] \mathcal{R}(h, \mathcal{S}_i \times \mathbf{v}([t_i, t_i + \Delta t])) \subseteq \mathcal{S}_i, \quad (17)$$

where $\mathcal{R}(\cdot, \cdot)$ defined in (1) represents the range of a function over an interval domain, $\mathcal{R}_i^+ \in \mathbb{I}\mathbb{R}^n$ is the set of states at time $t_i \geq 0$, $\Delta t > 0$ is the time step size, $\mathbf{v}([t_i, t_i + \Delta t]) \in \mathbb{I}\mathbb{R}^m$ over-approximates the range of all $u \in \mathbb{V}$ on the interval $[t_i, t_i + \Delta t]$ for a given set $\mathbb{V} \subseteq \mathbb{U}$, and the unknown function h is given by $h(x, u) = f(x) + G(x)u$ for $x \in \mathcal{X}, u \in \mathcal{U}$. Specifically, we give a bound on the time step size Δt such that \mathcal{S}_i exists and provide an analytic expression for \mathcal{S}_i . Recall that \mathcal{S}_i and Δt enable to compute an over-approximation of the reachable set when using interval Taylor-based methods.

Theorem 2 (EXPLICIT A PRIORI ROUGH ENCLOSURE). *Assume that the time step size Δt satisfies*

$$(\sqrt{n}\beta_i)\Delta t < 1, \quad (18)$$

where $\beta_i = \sqrt{\sum_{k=1}^n (L_{f_k} + \sum_{l=1}^m L_{G_{k,l}} |\mathbf{v}([t_i, t_i + \Delta t])|_l)^2}$. Then, the set \mathcal{S}_i given by

$$\mathcal{S}_i = \mathcal{R}_i^+ + \frac{\Delta t \|\mathbf{f}(\mathcal{R}_i^+) + \mathbf{G}(\mathcal{R}_i^+) \mathbf{v}([t_i, t_i + \Delta t])\|_\infty}{1 - \sqrt{n}\Delta t \beta_i} [-1, 1]^n \quad (19)$$

is an a priori rough enclosure solution of the fixed-point equation (17). The functions \mathbf{f} and \mathbf{G} in (19) can be any known interval extensions of the unknown functions f and G .

A proof of Theorem 2 is based on several lemmas and is provided in Appendix A. Intuitively, we use the bounds on the Lipschitz constants to provide explicit bounds on the rate of changes of solutions of the differential equation (11), which provide intuition on how to pick a set \mathcal{S}_i . However, the condition involves β_i which is also dependent on Δt , as $\mathbf{v}([t_i, t_i + \Delta t])$ depends on Δt . The following corollary gives an explicit bound on Δt based on the constraint set \mathcal{U} .

Corollary 1. *Under the notation of Theorem 2, assume that*

$$(\sqrt{n}\beta_\infty)\Delta t < 1, \quad (20)$$

where $\beta_\infty = \sqrt{\sum_{k=1}^n (L_{f_k} + \sum_{l=1}^m L_{G_{k,l}} |\mathcal{U}|_l)^2}$. Then, \mathcal{S}_i given by (19) is a solution of (17).

Proof. This is immediate by observing that $\frac{1}{\beta_\infty} \leq \frac{1}{\beta_i}$ since $\mathbf{v}([t_i, t_i + \Delta t]) \subseteq \mathcal{U}$. Hence, Theorem 2 applies. \square

C. Interval Taylor-Based Method for Differential Inclusions

We compute an over-approximation of the reachable set of the dynamics described by the differential inclusion (16). The obtained over-approximation is also an over-approximation of the reachable set of the unknown system. Theorem 3 provides a closed-form expression for such over-approximating sets.

Theorem 3 (REACHABLE SET OVER-APPROXIMATION). *Given a trajectory \mathcal{T}_N , a set $\mathbb{V} \subseteq \mathbb{U}$ of piecewise- \mathcal{C}^{D_u} control signals for $D_u \geq 1$, the bounds L_f and L_G , the set $\mathcal{R}_i^+ \in \mathbb{I}\mathbb{R}^n$ of states at time t_i , assume that $\Delta t > 0$ satisfies (18). Then, an over-approximation \mathcal{R}_{i+1}^+ of the reachable set at $t_i + \Delta t$ of dynamics described by the differential inclusion $\dot{x} \in \mathbf{f}(x) + \mathbf{G}(x)u$ with $u \in \mathbb{V}$ is given by*

$$\begin{aligned} \mathcal{R}_{i+1}^+ = & \mathcal{R}_i^+ + \left(\mathbf{f}(\mathcal{R}_i^+) + \mathbf{G}(\mathcal{R}_i^+) \mathbf{v}(t_i) \right) \Delta t \\ & + \left(\mathcal{J}_f + \mathcal{J}_G \mathcal{V}_i \right) \left(\mathbf{f}(\mathcal{S}_i) + \mathbf{G}(\mathcal{S}_i) \mathcal{V}_i \right) \frac{\Delta t^2}{2} \\ & + \mathbf{G}(\mathcal{S}_i) \mathcal{V}_i^{(1)} \frac{\Delta t^2}{2}, \end{aligned} \quad (21)$$

where $\mathbf{v}(t_i)$, $\mathcal{V}_i = \mathbf{v}([t_i, t_i + \Delta t])$, and $\mathcal{V}_i^{(1)} = \mathbf{v}^{(1)}([t_i, t_i + \Delta t])$ denote the intervals that contain the range of all the control signals in \mathbb{V} and their first derivative on $[t_i, t_i + \Delta t]$ and at t_i . The interval matrices $\mathcal{J}_f \in \mathbb{I}\mathbb{R}^{n \times n}$ and $\mathcal{J}_G \in \mathbb{I}\mathbb{R}^{n \times m \times n}$ are interval extensions of the Jacobian of f and G , given by

$$\mathcal{J}_f = \begin{bmatrix} L_{f_1}[-1, 1] & \cdots & L_{f_1}[-1, 1] \\ \vdots & \ddots & \vdots \\ L_{f_n}[-1, 1] & \cdots & L_{f_n}[-1, 1] \end{bmatrix}, \quad (22)$$

$$\mathcal{J}_G = \begin{bmatrix} L_{G_{1,1}}[-1, 1]^n & \cdots & L_{G_{1,m}}[-1, 1]^n \\ \vdots & \ddots & \vdots \\ L_{G_{n,1}}[-1, 1]^n & \cdots & L_{G_{n,m}}[-1, 1]^n \end{bmatrix}, \quad (23)$$

and the set \mathcal{S}_i can be either obtained by (19) of Theorem 2 or by solving the fixed-point equation

$$\mathcal{R}_i^+ + [0, \Delta t] \left(\mathbf{f}(\mathcal{S}_i) + \mathbf{G}(\mathcal{S}_i) \mathcal{V}_i \right) \subseteq \mathcal{S}_i. \quad (24)$$

For $k, p \in \mathbb{N}_{[1, n]}$, we define the (k, p) component of $\mathcal{J}_G \mathcal{V}_i$ as

$$(\mathcal{J}_G \mathcal{V}_i)_{k,p} = \sum_{l=1}^m (\mathcal{J}_G)_{k,l,p} (\mathcal{V}_i)_l. \quad (25)$$

A proof of Theorem 3 is provided in our previous work [10], where we use a Taylor expansion (6) of order $D = 2$ with an over-approximation of the unknown Jacobian of f and G obtained from the Lipschitz bounds. Besides, Theorem 2 ensures the existence of an a priori rough enclosure \mathcal{S}_i .

Remark 2. *Theorem 3 implicitly relies on the knowledge of the intervals $\mathbf{v}(t_i)$, $\mathbf{v}([t_i, t_i + \Delta t])$, and $\mathbf{v}^{(1)}([t_i, t_i + \Delta t])$ given solely by the definition of the set \mathbb{V} . For such intervals to exist, all the controls $u \in \mathbb{V}$ must be at least \mathcal{C}^1 on the subdomain $[t_i, t_i + \Delta t]$. Hence, the assumption that the controls in \mathbb{V} are piecewise- \mathcal{C}^{D_u} with $D_u \geq 1$. However, when \mathbb{V} is such that $D_u = 0$, we propose Corollary 2 to over-approximate the reachable set of the system. We will focus on Theorem 3 in the remainder of the paper since, in practise, the control signals of interest are piecewise-constants.*

Algorithm 2 DaTaReach: Over-approximation of the reachable set of unknown smooth systems.

Input: Single trajectory \mathcal{T}_N , upper bounds on the Lipschitz constants, set \mathbb{V} of control signals, time step size Δt satisfying (18) or (20), maximum number of time steps $T > N$, noise bounds $\eta \geq 0$ and $\bar{\eta} \geq 0$.

Optional: Any of the side information 1–4.

Output: Over-approximations $\{\mathcal{R}_i^+\}_{i=N+1}^T$ of the reachable sets at times $t_i = t_N + (i - N)\Delta t$ with $i \in \mathbb{N}_{[N+1, T]}$.

- 1: Define $R_{N+1}^+ \leftarrow \{[\tilde{x}_{N+1} - \eta, \tilde{x}_{N+1} + \eta]\}$
- 2: **for** $i \in \mathbb{N}_{[N+1, T-1]}$ **do**
- 3: Compute $v(t_i)$, \mathcal{V}_i , and $\mathcal{V}_i^{(1)}$ in Theorem 3 from \mathbb{V}
- 4: Compute $f(\mathcal{R}_i^+)$ and $G(\mathcal{R}_i^+)$ via Theorem 1
- 5: Get tighter $f(\mathcal{R}_i^+)$ and $G(\mathcal{R}_i^+)$ if any side information
- 6: Compute \mathcal{S}_i via (19) or (24), $f(\mathcal{R}_i^+)$, and $G(\mathcal{R}_i^+)$
- 7: Compute $f(\mathcal{S}_i)$ and $G(\mathcal{S}_i)$ via Theorem 1
- 8: Get tighter $f(\mathcal{S}_i)$ and $G(\mathcal{S}_i)$ if any side information
- 9: Compute \mathcal{J}_f and \mathcal{J}_G via (22) and (23)
- 10: Get tighter \mathcal{J}_f and \mathcal{J}_G if any side information
- 11: Compute \mathcal{R}_{i+1}^+ via (21)
- 12: **end for**
- 13: **return** $\{\mathcal{R}_i^+\}_{i=N+1}^T$

Corollary 2. Under the notation of Theorem 3, assume that $D_u = 0$. Then, an over-approximation \mathcal{R}_{i+1}^+ of the reachable set of the dynamics described by the differential inclusion $\dot{x} \in f(x) + G(x)u$ with $u \in \mathbb{V}$ is given by

$$\mathcal{R}_{i+1}^+ = \mathcal{R}_i^+ + (f(\mathcal{S}_i) + G(\mathcal{S}_i)\mathcal{V}_i)\Delta t.$$

D. Side Information

As motivated in the introduction, prior knowledge on the underlying unknown dynamics might be available. However, Theorem 3, as it is, does not incorporate any side information. We now show how to incorporate a variety of side information to obtain tighter over-approximations of the reachable sets.

Side information 1 (VECTOR FIELD BOUNDS). We are given $\mathcal{R}^{fA} \in \mathbb{I}\mathbb{R}^n$ and $\mathcal{R}^{GA} \in \mathbb{I}\mathbb{R}^{n \times m}$ as supersets of the range of f and G , respectively, over a given set $A \subseteq \mathcal{X}$.

Given a set $S \subseteq A$, tight extensions of f and G over S can be obtained by the update $f(S) \leftarrow f(S) \cap \mathcal{R}^{fA}$ and $G(S) \leftarrow G(S) \cap \mathcal{R}^{GA}$.

Side information 2 (GRADIENT BOUNDS). We are given bounds on the gradient of some components of f and G . Such side information may include the monotonicity of f and G .

These bounds can be used in equations (22) and (23) to provide tight interval extensions \mathcal{J}_f and \mathcal{J}_G . For example, if the function $(f)_k$ is known to be non-decreasing with respect to the variable x_p on a set $A \subseteq \mathcal{X}$, then we obtain a tighter \mathcal{R}_{i+1}^+ by the update $(\mathcal{J}_f)_{k,p} \leftarrow (\mathcal{J}_f)_{k,p} \cap \mathbb{R}_+$ if $\mathcal{S}_i \subseteq A$.

This side information expresses properties such as decoupling between states. For example, if the state $(x(t))_p$ does not directly affect $(\dot{x}(t))_k$ for some $p, k \in \mathbb{N}_{[1, n]}$ under any control signal in \mathcal{U} , we can obtain a tighter over-approximation of the reachable set by setting to zero the intervals $(\mathcal{J}_f)_{k,p}$

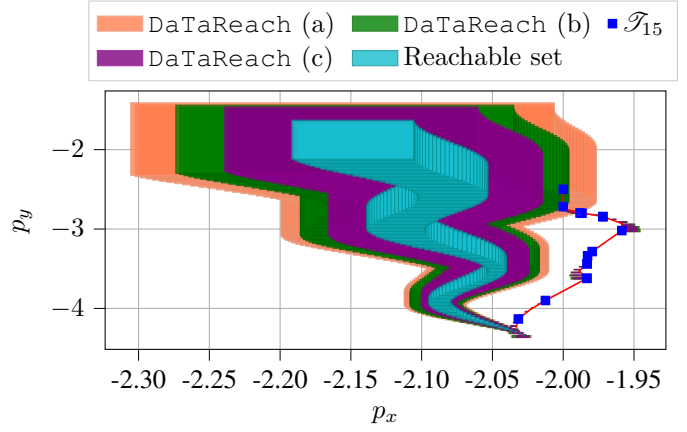


Fig. 3. Over-approximation of the reachable set of the unicycle system in the x - y plane. The trajectory \mathcal{T}_{15} comes from a noise-free trajectory of the system obtained by a randomly generated piecewise-constant control signal $u(t) \in \mathcal{U}$ for $t \leq t_{15} = 1.5$ s. The parameters for DaTaReach were given by $\Delta t = 0.02$, $T = 200$, and $\mathbb{V} = \{t \mapsto [1 + a_1, \cos(6(t - t_{15})) + a_2] \mid a_1 \in [-0.1, 0.1], a_2 \in [-0.01, 0.01]\}$. The case (a) ignores the side information $f(x) = f(\theta)$, case (b) is exactly the setting of Example 2, and case (c) assumes the extra knowledge that $f = 0$ and $(G)_{3,2} = 1$.

and $(\mathcal{J}_G)_{k,l,p}$ for all $l \in \mathbb{N}_{[1, m]}$. For the unicycle system of Example 2, since f and G depends only on the heading θ , the Jacobian terms $(\mathcal{J}_f)_{1,1}$, $(\mathcal{J}_f)_{1,2}$, $(\mathcal{J}_f)_{2,1}$, $(\mathcal{J}_f)_{2,2}$, $(\mathcal{J}_f)_{3,1}$, $(\mathcal{J}_f)_{3,2}$, $(\mathcal{J}_G)_{1,1,1}$, $(\mathcal{J}_G)_{1,1,2}$, $(\mathcal{J}_G)_{2,1,1}$, $(\mathcal{J}_G)_{2,1,2}$, $(\mathcal{J}_G)_{3,2,1}$ and $(\mathcal{J}_G)_{3,2,2}$ must all be set to zero.

Side information 3 (ALGEBRAIC CONSTRAINTS). We are given a differentiable map $g : \mathbb{R}^n \mapsto \mathbb{R}^q$ satisfying the constraint $g(x) \geq 0$ over the state x of the unknown system.

Without loss of generality, we consider that $g : \mathbb{R}^n \mapsto \mathbb{R}$. This side information provides bounds on f , G , \mathcal{J}_f , and \mathcal{J}_G locally. Specifically, if we derivate the function $t \mapsto g(x(t))$, then we obtain a new constraint $w : \mathbb{R}^n \times \mathbb{R}^{n \times m} \times \mathbb{R}^m \times \mathbb{R}^n \mapsto \mathbb{R}$ such that $w(f(x), G(x), u, x) \geq 0$. If we compute the partial derivatives of w with respect to all the states x_i , then we obtain a new constraint $z : \mathbb{R}^{n \times n} \times \mathbb{R}^{n \times m \times n} \times \mathbb{R}^m \times \mathbb{R}^n \mapsto \mathbb{R}^n$ such that $z(\frac{\partial f}{\partial x}(x), \frac{\partial G}{\partial x}(x), u, x) \geq 0$. The obtained constraints w and z can be incorporated in the computation of \mathcal{R}_{i+1}^+ through the framework of contractors [27], [28]. Specifically, given a set A of states, a set \mathcal{V} of control values, and the over-approximating sets $f(A)$, $G(A)$, \mathcal{J}_f , and \mathcal{J}_G , the methods based on contractor programming [27] compute the intervals $C_{f(A)} \subseteq f(A)$, $C_{G(A)} \subseteq G(A)$, $C_{\mathcal{J}_f} \subseteq \mathcal{J}_f$, and $C_{\mathcal{J}_G} \subseteq \mathcal{J}_G$. These intervals are such that for all $f_x \in f(A) \setminus C_{f(A)}$, $G_x \in G(A) \setminus C_{G(A)}$, $p_f \in \mathcal{J}_f \setminus C_{\mathcal{J}_f}$, and $p_G \in \mathcal{J}_G \setminus C_{\mathcal{J}_G}$, there does not exist $x \in A$ and $u \in \mathcal{V}$ such that $w(f_x, G_x, u, x) \geq 0$ and $z(p_f, p_G, u, x) \geq 0$. In other words, it contracts $f(A)$, $G(A)$, \mathcal{J}_f , and \mathcal{J}_G by pruning out some values that do not satisfy the constraints imposed by w and z . In the computation of \mathcal{R}_{i+1}^+ , the set A is either \mathcal{R}_i^+ or \mathcal{S}_i , and the set \mathcal{V} is \mathcal{V}_i .

Side information 4 (PARTIAL DYNAMICS KNOWLEDGE). We are given terms of some components of the vector field of (11). That is, we consider that the dynamics are in the form

$$\dot{x} = f_{\text{kn}}(x) + f_{\text{ukn}}(x) + (G_{\text{kn}}(x) + G_{\text{ukn}}(x))u,$$

where the functions f_{kn} and G_{kn} are known while f_{ukn} and G_{ukn} are unknown. Such side information is usually obtained

from the application of elementary laws of physics.

The new functions \mathbf{f} and \mathbf{G} are given by $\mathbf{f} = \mathbf{f}_{\text{kn}} + \mathbf{f}_{\text{ukn}}$ and $\mathbf{G} = \mathbf{G}_{\text{kn}} + \mathbf{G}_{\text{ukn}}$, where the functions \mathbf{f}_{kn} and \mathbf{f}_{ukn} are interval extensions of known functions f_{kn} and G_{kn} , respectively, and the functions \mathbf{f}_{ukn} and \mathbf{G}_{ukn} are obtained by Theorem 1 using $L_{f_{\text{ukn}}}$, $L_{G_{\text{ukn}}}$, and the new trajectory $\mathcal{F}'_N = \{(x_i, \dot{x}_i - (f_{\text{kn}}(x_i) + G_{\text{kn}}(x_i)u_i), u_i) \mid (x_i, \dot{x}_i, u_i) \in \mathcal{F}_N\}$. Furthermore, the new Jacobian terms in the computation of \mathcal{R}_{i+1}^+ are given by $\mathcal{J}_f = \frac{\partial \mathbf{f}_{\text{kn}}}{\partial \mathbf{x}}(\mathcal{S}_i) + \mathcal{J}_{f_{\text{ukn}}}$ and $(\mathcal{J}_G)_{k,l,p} = \frac{\partial (\mathbf{G}_{\text{kn}})_{k,l}}{\partial \mathbf{x}_p}(\mathcal{S}_i) + (\mathcal{J}_{G_{\text{ukn}}})_{k,l,p}$, respectively.

DaTaReach (Algorithm 2) addresses Problem 1 by combining Theorem 3 and the available side information. We demonstrate the value of additional side information on the unicycle system. Figure 3 shows that, as expected, the over-approximation becomes tighter with richer side information.

V. CONTROL SYNTHESIS FOR UNKNOWN DYNAMICS

We develop DaTaControl to address Problem 2. It computes approximate solutions to the one-step optimal control problem (10) using the over-approximation of the reachable set. Specifically, it replaces the *unknown* next state x_{i+1} in problem (10) with the corresponding over-approximation of the reachable set \mathcal{R}_{i+1}^+ , which may be computed using Theorem 3. Unfortunately, \mathcal{R}_{i+1}^+ is a nonconvex function of the control signal u , the decision variable. Therefore, it constructs an over-approximation of \mathcal{R}_{i+1}^+ , in Theorem 4, that is convex in the given constant control signal u applied in the time interval $[t_i, t_{i+1}]$. The new over-approximation provides tractable approximations of the optimization problem (10).

A. Approximate One-Step Optimal Control

First, Theorem 4 provides a control-affine linearization of the over-approximation \mathcal{R}_{i+1}^+ given by Theorem 3.

Theorem 4. *Under the notation of Theorem 3 and given a constant control signal $u : t \mapsto u_i$ applied between t_i and $t_{i+1} = t_i + \Delta t$ with $u_i \in \mathcal{U}$, the reachable set \mathcal{R}_{i+1} satisfies*

$$\mathcal{R}_{i+1} \subseteq (\mathcal{B}_i + \mathcal{A}_i^+ u_i) \cap (\mathcal{B}_i + \mathcal{A}_i^- u_i), \quad (26)$$

where the intervals \mathcal{A}_i^- , \mathcal{A}_i^+ , and \mathcal{B}_i are given by

$$\begin{aligned} \mathcal{A}_i^- &= \mathbf{G}(\mathcal{R}_i^+) \Delta t + (\mathcal{J}_f \mathbf{G}(\mathcal{S}_i) + \mathcal{J}_G^T (\mathbf{f}(\mathcal{S}_i) + \mathbf{G}(\mathcal{S}_i) \mathcal{U})) \frac{\Delta t^2}{2}, \\ \mathcal{A}_i^+ &= \mathbf{G}(\mathcal{R}_i^+) \Delta t + ((\mathcal{J}_f + \mathcal{J}_G \mathcal{U}) \mathbf{G}(\mathcal{S}_i) + \mathcal{J}_G^T \mathbf{f}(\mathcal{S}_i)) \frac{\Delta t^2}{2}, \\ \mathcal{B}_i &= \mathcal{R}_i^+ + \mathbf{f}(\mathcal{R}_i^+) \Delta t + \mathcal{J}_f \mathbf{f}(\mathcal{S}_i) \frac{\Delta t^2}{2}, \end{aligned} \quad (27)$$

where $(\mathcal{J}_G^T)_{k,p,l} = (\mathcal{J}_G)_{k,l,p}$ for $k, p \in \mathbb{N}_{[1,n]}$ and $l \in \mathbb{N}_{[1,m]}$, and the a priori rough enclosure \mathcal{S}_i is either obtained by (19) with $\mathbf{v}([t_i, t_{i+1}])$ replaced by \mathcal{U} or is a solution of

$$\mathcal{R}_i^+ + [0, \Delta t] \left(\mathbf{f}(\mathcal{S}_i) + \mathbf{G}(\mathcal{S}_i) \mathcal{U} \right) \subseteq \mathcal{S}_i. \quad (28)$$

A proof of Theorem 4 is provided in our previous work [10], where we linearize the quadratic term in u of the closed-form expression (21). Such linearization enables to have

a control-affine over-approximation at time $t_i + \Delta t$. Intuitively, the linearization comes from over-approximating the quadratic term $(\mathcal{J}_G u_i)(\mathbf{G}(\mathcal{S}_i) u_i)$ of Theorem 3 by linear terms $(\mathcal{J}_G \mathcal{U})(\mathbf{G}(\mathcal{S}_i) u_i)$ and $(\mathcal{J}_G u_i)(\mathbf{G}(\mathcal{S}_i) \mathcal{U})$. Thus, we obtain two different over-approximations characterized by \mathcal{A}^+ and \mathcal{A}^- , which are finally intersected to provide a tighter set.

Next, we use the control-affine linearization of Theorem 4 to propose two convex optimization problems which solutions approximate the one-step optimal control problem (10). The first problem, called *optimistic control problem*, is given by

$$\underset{u_i \in \mathcal{U}}{\text{minimize}} \quad \inf_{\substack{x_{i+1} \in \mathcal{X}, \\ x_{i+1} \in (\mathcal{B}_i + \mathcal{A}_i^+ u_i) \cap (\mathcal{B}_i + \mathcal{A}_i^- u_i)}} c(\tilde{x}_i, u_i, x_{i+1}), \quad (29)$$

where the goal is to minimize the best possible cost value over all possible state x_{i+1} in the over-approximation (26). The second problem, called *idealistic control problem*, is an idealistic approximation of (10) given by

$$\underset{u_i \in \mathcal{U}}{\text{minimize}} \quad c(\tilde{x}_i, u_i, b_i^{\text{ide}} + A_i^{\text{ide}} u_i), \quad (30)$$

where the goal is to minimize the cost associated to a specific trajectory $x_{i+1} = b_i^{\text{ide}} + A_i^{\text{ide}} u_i$ in the over-approximation (11), idealistically considered as the unknown next state evolution. Specifically, we choose x_{i+1} according to the middle point between a point $x_{i+1}^+ \in \mathcal{B}_i + \mathcal{A}_i^+ u_i$ and a point $x_{i+1}^- \in \mathcal{B}_i + \mathcal{A}_i^- u_i$. More specifically, given weights $w^+, w^- \in [0, 1]$, we pick b_i^{ide} and A_i^{ide} as follows

$$\begin{aligned} b_i^{\text{ide}} &= \frac{1}{2} \left(w^+ \overline{\mathcal{B}_i} + (1 - w^+) \underline{\mathcal{B}_i} + w^- \overline{\mathcal{B}_i} + (1 - w^-) \underline{\mathcal{B}_i} \right), \\ A_i^{\text{ide}} &= \frac{1}{2} \left(w^+ \overline{\mathcal{A}_i^+} + (1 - w^+) \underline{\mathcal{A}_i^+} + w^- \overline{\mathcal{A}_i^-} + (1 - w^-) \underline{\mathcal{A}_i^-} \right). \end{aligned} \quad (31)$$

Remark 3. *Another possible approximation of the one-step optimal control problem (10) is a pessimistic control problem where, in contrast to the optimistic control problem, the goal is to minimize the worst possible one-step cost value over all possible state x_{i+1} in the over-approximation of the reachable set. Such a pessimistic control problem can be formulated as a semidefinite program using tools from robust optimization.*

Finally, we provide, in Theorem 5, bounds on the suboptimality of the optimistic and idealistic control with respect to the optimal control when the cost function c is quadratic.

Assumption 2 (QUADRATIC ONE-STEP COST). *We consider that the one-step cost function c is a convex quadratic function,*

$$c(x, u, y) = \begin{bmatrix} y \\ u \end{bmatrix}^T \begin{bmatrix} Q & S \\ S^T & R \end{bmatrix} \begin{bmatrix} y \\ u \end{bmatrix} + \begin{bmatrix} q \\ r \end{bmatrix}^T \begin{bmatrix} y \\ u \end{bmatrix}, \quad (32)$$

where $q \in \mathbb{R}^n$, $r \in \mathbb{R}^m$, $Q = Q^T \in \mathbb{R}^{n \times n}$, $R = R^T \in \mathbb{R}^{m \times m}$, and $S \in \mathbb{R}^{n \times m}$.

Theorem 5 (SUBOPTIMALITY BOUND). *Let c_i^* , c_i^{opt} , and c_i^{ide} be the optimal cost of the one-step optimal control problem (10), the optimistic control problem (29), and the idealistic control problem (30), respectively, at the sampling time t_i . The following inequality holds*

$$\begin{aligned} |c_i^* - c_i| &\leq \max \left(\|w(\mathcal{B}_i) + w(\mathcal{A}_i^+) \mathcal{U}\|_2 K(\mathcal{A}_i^+), \right. \\ &\quad \left. \|w(\mathcal{B}_i) + w(\mathcal{A}_i^-) \mathcal{U}\|_2 K(\mathcal{A}_i^-) \right), \end{aligned} \quad (33)$$

Algorithm 3 DaTaControl: Approximate one-step optimal control solution at $t_i \geq t_N$

Input: Single trajectory \mathcal{T}_{i-1} of length $i > N$, ime step size Δt satisfying (20), Lipschitz bounds, noise bounds $\eta \geq 0$ and $\bar{\eta} \geq 0$, one-step cost c , current state $\tilde{x}_i = \tilde{x}(t_i)$.

Optional: Any of the side information 1–4.

Output: Constant control $\hat{u}_i \in \mathcal{U}$ to apply between t_i and $t_i + \Delta t$ that approximates a solution of (10).

- 1: Define $\mathcal{R}_i^+ \leftarrow \{\tilde{x}_i - \eta, \tilde{x}_i + \eta\}$
- 2: Compute $\mathbf{f}(\mathcal{R}_i^+)$ and $\mathbf{G}(\mathcal{R}_i^+)$ via Theorem 1
- 3: Get tighter $\mathbf{f}(\mathcal{R}_i^+)$ and $\mathbf{G}(\mathcal{R}_i^+)$ if any side information
- 4: Compute \mathcal{S}_i via (19) or (28), $\mathbf{f}(\mathcal{R}_i^+)$, and $\mathbf{G}(\mathcal{R}_i^+)$
- 5: Compute $\mathbf{f}(\mathcal{S}_i)$ and $\mathbf{G}(\mathcal{S}_i)$ via Theorem 1
- 6: Get tighter $\mathbf{f}(\mathcal{S}_i)$ and $\mathbf{G}(\mathcal{S}_i)$ if any side information
- 7: Compute \mathcal{J}_f and \mathcal{J}_G via (22) and (23)
- 8: Get tighter \mathcal{J}_f and \mathcal{J}_G if any side information
- 9: Compute \mathcal{B}_i , \mathcal{A}_i^+ , and \mathcal{A}_i^- via (27)
- 10: Compute \hat{u}_i as the solution of either (29) or (30)
- 11: **return** \hat{u}_i

where c_i is either c_i^{opt} or c_i^{ide} , $w(\mathcal{A}) = \bar{\mathcal{A}} - \underline{\mathcal{A}}$ is the width of an interval \mathcal{A} , and $K(\mathcal{A}) \in \mathbb{R}^{n \times m}$, is given by

$$K(\mathcal{A}) = \min (\|2|SU| + q + 2|Q(\mathcal{B}_i + \mathcal{A}U)|\|_2, \|2|SU| + q + 2|Q\mathcal{X}'\|_2).$$

We provide a proof for Theorem 5 in our previous work [10] where we use the local Lipschitz property of the cost function c to characterize its variations over the over-approximation set (26).

Remark 4. The main term of the suboptimality bound in Theorem 5 is directly related to the width of the over-approximation of the reachable set at time $t_i + \Delta t$. Thus, as the over-approximation of the reachable set becomes tighter, the gap with the unknown optimal cost decreases. Specifically, it is straightforward to see that the more data and richer side information are used in the computation of \mathcal{B}_i , \mathcal{A}_i^+ , and \mathcal{A}_i^- , the tighter the widths $w(\mathcal{B}_i)$, $w(\mathcal{A}_i^+)$, and $w(\mathcal{A}_i^-)$ become.

B. Real-Time Approximate Control Synthesis

In this section, we investigate efficient algorithms to compute solutions of the optimistic control problem (29) and the idealistic control problem (30) when the one-step cost function c satisfies Assumption 2. We provide a bound on the number of elementary operations required by DaTaControl to synthesize near-optimal control values. Such a bound makes DaTaControl a good candidate for real-time control of systems with a priori unknown dynamics.

Lemma 3 (APPROXIMATE QUADRATIC PROBLEMS). *Without loss of generality, consider that either $(\mathcal{U})_l \geq 0$ or $(\mathcal{U})_l \leq 0$ for a given $l \in \mathbb{N}_{[1,m]}$. The optimistic control problem (29) can be reformulated as the quadratic programming problem*

$$\underset{u_i \in \mathcal{U}, x_{i+1} \in \mathcal{X}}{\text{minimize}} \quad c(x_i, u_i, x_{i+1}), \quad (34a)$$

$$\text{subject to} \quad \underline{\mathcal{B}}_i + \mathcal{A}_i^{1+} u_i \leq x_{i+1} \leq \bar{\mathcal{B}}_i + \mathcal{A}_i^{s+} u_i, \quad (34b)$$

$$\underline{\mathcal{B}}_i + \mathcal{A}_i^{1-} u_i \leq x_{i+1} \leq \bar{\mathcal{B}}_i + \mathcal{A}_i^{s-} u_i, \quad (34c)$$

and the idealistic control problem (30) as the convex, box-constrained quadratic programming problem

$$\underset{u_i \in \mathcal{U}}{\text{minimize}} \quad \frac{1}{2} u_i^T Q_i u_i + q_i^T u_i + p_i, \quad (35)$$

where Q_i , p_i , and q_i are given by

$$Q_i = 2(\mathcal{A}_i^{\text{ide}})^T Q \mathcal{A}_i^{\text{ide}} + 4(\mathcal{A}_i^{\text{ide}})^T S + 2R, \quad (36)$$

$$q_i = 2(S + Q \mathcal{A}_i^{\text{ide}})^T b_i^{\text{ide}} + (\mathcal{A}_i^{\text{ide}})^T q + r, \quad (37)$$

$$p_i = (b_i^{\text{ide}})^T Q b_i^{\text{ide}} + q^T b_i^{\text{ide}}, \quad (38)$$

the matrices \mathcal{A}_i^{1+} and \mathcal{A}_i^{s+} are such that, for all $k \in \mathbb{N}_{[1,n]}$ and $l \in \mathbb{N}_{[1,m]}$, if $(\mathcal{U})_l \geq 0$, then $(\mathcal{A}_i^{s+})_{k,l} = (\mathcal{A}_i^{1+})_{k,l}$ and $(\mathcal{A}_i^{1+})_{k,l} = (\mathcal{A}_i^+)_{k,l}$, otherwise $(\mathcal{A}_i^{1+})_{k,l} = (\mathcal{A}_i^+)_{k,l}$ and $(\mathcal{A}_i^{s+})_{k,l} = (\mathcal{A}_i^+)_{k,l}$, and the matrices \mathcal{A}_i^{1-} and \mathcal{A}_i^{s-} are obtained in a similar manner with \mathcal{A}_i^+ replaced by \mathcal{A}_i^- .

We provide a proof of Lemma 4 in Appendix A, where interval arithmetic and elementary algebra provide (34), (35).

Remark 5. Lemma 3 assumes that the components of \mathcal{U} are either nonnegative or nonpositive in order to simplify the notation and provide a simple expression for the optimistic control problem. In the general case, \mathcal{U} can be partitioned into subsets where each component is either nonnegative or nonpositive. Then, we solve the optimistic control problem on each partition and return the control value from the partition with the minimum cost. It is important to note that the idealistic control problem (35) does not require any partitioning of \mathcal{U} and can therefore be solved as a one-shot quadratic problem. Besides, if $m_i \leq m$ is the number of components of \mathcal{U} having an indefinite sign, then solving the optimistic control problem (29) requires to solve 2^{m_i} quadratic programming problems similar to (34). As a consequence, the number of problems drastically reduces when the intervals $(\mathcal{U})_l$ are known to be nonnegative or nonpositive a priori.

We seek for algorithms to compute solutions of (34) and (35) while providing bounds on the number of elementary operations to obtain such solutions. The number of elementary operations, referred as flops, is obtained by counting the number of operations $+$, $-$, $*$, $/$, \max , \min , and $\sqrt{\cdot}$ executed by the algorithms. When the cost function c is assumed to be strongly convex, i.e., either Q is positive definite or R is positive definite, the authors of [29] develop an algorithm [29, Algorithm 1] based on applying the fast gradient-projection method on the dual of the optimization problem. The algorithm can be directly used to solve both the optimistic and idealistic control problem. The rate of convergence of such an algorithm is proportional to $1/\sqrt{\epsilon}$, where ϵ is the desired accuracy for optimality. Furthermore, the authors provide a maximum number of flops, cubic polynomial in both n and m , required by the algorithm to terminate. We consider the general case where c might not be strongly convex and develop an algorithm inspired from [30] to solve the idealistic control problem with a linear rate of convergence to a solution, i.e. proportional to $\ln 1/\epsilon$. We deduce a maximum number of flops, quadratic in both n and m and linear in the length of the

Algorithm 4 AdaRES: Accelerated proximal gradient with adaptive restart for optimization problems satisfying the hypothesis of Lemma 4

Input: Desired accuracy ϵ for the optimality, a point $y_0 \in \mathbb{R}^p$, smoothness parameter L , an initial estimation μ_0 of $\mu_{\mathcal{P}}(y_0)$, and a function $\Pi_{\mathcal{P}}$ for projection onto \mathcal{P} .

Output: $\hat{y} \in \mathcal{P}$ such that $l(\hat{y}) - l^* \leq \epsilon$

```

1:  $s \leftarrow -1, t_s \leftarrow 0, y_{s,t_s} \leftarrow y_0, y_{s+1,0} \leftarrow \Pi_{\mathcal{P}}(y_{s,t_s})$ 
2: do
3:    $s \leftarrow s + 1, t \leftarrow 0$ 
4:    $C_s \leftarrow 16\mu_s^{-1} \|y_{s,0} - y_{s-1,t_{s-1}}\|_2^2$ 
5:    $K_s \leftarrow 2\sqrt{\epsilon\mu_s^{-1}} - 1$ 
6:   do
7:      $x_0 \leftarrow y_{s,t}, z_0 \leftarrow x_0, \theta_0 \leftarrow 1$ 
8:     for  $k \in \mathbb{N}_{[0,\dots,K_s-1]}$  do
9:        $x_{k+1} \leftarrow \Pi_{\mathcal{P}}(z_k - \frac{1}{L}\nabla l(z_k))$ 
10:       $\theta_{k+1}$  solves  $\theta_{k+1} = (1 - \theta_{k+1})\theta_k^2$ 
11:       $\beta_{k+1} = \theta_k(1 - \theta_k)/(\theta_k^2 + \theta_{k+1})$ 
12:       $z_{k+1} = x_{k+1} + \beta_{k+1}(x_{k+1} - x_k)$ 
13:    end for
14:     $y_{s,t+1} \leftarrow z_{K_s}, t \leftarrow t + 1$ 
15:    while  $\|\Pi_{\mathcal{P}}(y_{s,t}) - y_{s,t}\|_2^2 > C_s(\theta_{K_s-1}^2\mu_s^{-1})^t$  or  $L^2\|\Pi_{\mathcal{P}}(y_{s,t}) - y_{s,t}\|_2^2 \leq \epsilon\mu_s/8$ 
16:       $t_s \leftarrow t, y_{s+1,0} \leftarrow \Pi_{\mathcal{P}}(y_{s,t_s}), \mu_{s+1} \leftarrow \mu_s/2$ 
17:      while  $L^2\|y_{s+1,0} - y_{s,t_s}\|_2^2 \leq \epsilon\mu_s/8$ 
18:         $\hat{N} \leftarrow 1 + \sum_{k=0}^s (t_s K_s + 1)$ 
19:      return  $\Pi_{\mathcal{P}}(y_{s,t_s})$ 

```

trajectory \mathcal{T}_p , required by DaTaControl to compute near-optimal control values when the idealistic formulation is used.

Lemma 4 ([30], THEOREM 2). Consider the convex optimization problem $\inf_{y \in \mathcal{P}} l(y)$, where $\mathcal{P} \subseteq \mathbb{R}^p$ is a convex set and $l: \mathbb{R}^p \mapsto \mathbb{R}$ is a differentiable convex function. Let $l^* \in \mathbb{R}$ be the optimal cost, \mathcal{P}^* the set of minimizers of the problem, and assume that there exists $L > 0$ and $\mu_{\mathcal{P}}(y_0) > 0$ for any $y_0 \in \mathbb{R}^p$ such that l satisfies the smoothness and local error bound assumption given respectively by

$$l(y) \leq l(w) + \nabla l(w)^T(y - w) + \frac{L}{2}\|y - w\|_2, \quad (39)$$

$$l(z) \geq l^* + \frac{\mu_{\mathcal{P}}(y_0)}{2}\text{dist}^2(z, \mathcal{P}^*), \quad (40)$$

for all $y, w \in \mathbb{R}^p$ and for all $z \in \mathcal{P}$ such that $l(z) \leq l(y_0)$. Then, for any $\mu_0 > 0$ and $\epsilon > 0$, Algorithm 4 returns a solution $\hat{y} \in \mathcal{P}$ such that $l(\hat{y}) - l^* \leq \epsilon$ with a total number of iterations \hat{N} satisfying

$$\begin{aligned} \hat{N} &\leq \left\lceil 2\sqrt{\frac{e}{\mu_0}} - 1 \right\rceil \left\lceil \ln \frac{16(l(y_0) - l^*)}{\epsilon\mu_0} \right\rceil, \text{ if } \mu_0 \leq \mu_{\mathcal{P}}(y_0), \\ \hat{N} &\leq 1 + \frac{2\sqrt{e}}{\sqrt{2}-1} \left(\sqrt{\frac{2}{\mu_{\mathcal{P}}(y_0)}} - \sqrt{\frac{1}{\mu_0}} \right) \left\lceil \ln \frac{2^8(l(y_0) - l^*)}{\epsilon\mu_{\mathcal{P}}(y_0)^2} \right\rceil \\ &\quad + \left\lceil \frac{16\sqrt{2e}}{\sqrt{\mu_{\mathcal{P}}(y_0)}} \right\rceil \left\lceil \ln \frac{2(l(y_0) - l^*)}{\epsilon\mu_{\mathcal{P}}(y_0)} \right\rceil, \text{ otherwise.} \end{aligned} \quad (41)$$

Remark 6. Lemma 4 guarantees a linear rate of convergence of AdaRES without knowing explicitly the parameter $\mu_{\mathcal{P}}(y_0)$.

Specifically, AdaRES achieves this rate of convergence by being able to check at iteration s whether the current estimate μ_s of $\mu_{\mathcal{P}}(y_0)$ is too large and finally by refining such estimate. Therefore, AdaRES can also be used to estimate a lower bound on $\mu_{\mathcal{P}}(y_0)$ and thus providing the guarantees (41). Note that the number of iterations \hat{N} represents also the total number of calls to the projection $\Pi_{\mathcal{P}}(\cdot)$ onto \mathcal{P} .

We exploit Lemma 4 to provide a maximum number of flops required by DaTaControl to terminate.

Theorem 6 (NUMBER OF FLOPS BY DATACONTROL). At each sampling time $t_i \geq t_N$, given a trajectory \mathcal{T}_i of length $i \geq N$, if DaTaControl based on the idealistic control problem (30) uses Algorithm 5 with $\mu_0 \leq \mu_{\mathcal{U}}(v_0)$ and without any side information 1–4, then DaTaControl takes at most N^{ide} flops to compute near-optimal control values with N^{ide} roughly given by

$$\begin{aligned} N^{\text{ide}} &= \left\lceil 2\sqrt{\frac{e}{\mu_0}} - 1 \right\rceil \left\lceil \ln \frac{32c_{\max}}{\epsilon\mu_0} \right\rceil (2m^2 + 16m + 2) \\ &\quad + 2i(8n + 9)(m + 1) + 24n^2m + 8n^2 + 8m^2 \\ &\quad + 18nm + 32n + 2m + 8, \end{aligned} \quad (42)$$

where $c_{\max} = |c(\mathcal{X}, \mathcal{U}, \mathcal{X})|$ is the maximum value in absolute value taken by the cost function over the range \mathcal{X} and \mathcal{U} . Furthermore, if any of the side information 1–4 is provided, then only a maximum of $R(n, m)$ flops will be added to N^{ide} where R is a polynomial of maximum degree 2.

We provide in Appendix A a proof of Theorem 6 where we show that (30) satisfies the assumption of Lemma 4 in order to bound the number of flops required by DaTaControl.

Remark 7. Even though the optimistic control problem (29) satisfies the assumptions of Lemma 4, computing the projection $\Pi_{\mathcal{P}}$ is as hard as solving directly (29). Hence, we choose to focus on the idealistic control problem. Note that μ_0 can be empirically estimated offline by solving the idealistic control problem on many instances of the same problem.

Algorithm 5 Solve the idealistic control problem

Input: Desired accuracy ϵ for optimality, a point $v_0 \in \mathbb{R}^m$, an initial estimation $\mu_0 > 0$ of $\mu_{\mathcal{U}}(v_0)$, parameters $\mathcal{B}_i, \mathcal{A}_i^+, \mathcal{A}_i^-$ from (27), weights $w^+, w^- \in [0, 1]$.

Output: $\hat{u}_i \in \mathcal{U}$ solution of (30)

- 1: Compute $A^{\text{ide}}, b^{\text{ide}}$ via (31), w^+ , and w^-
- 2: Compute Q_i, q_i, p_i via (36), (37), (38), $Q, R, S, q, r, A^{\text{ide}}$, and b^{ide}

- 3: Define for $l \leq m$, $(\Pi_{\mathcal{U}}(v))_l = \begin{cases} (\underline{\mathcal{U}})_l, & \text{if } v_l \leq (\underline{\mathcal{U}})_l \\ (v)_l, & \text{if } (v)_l \in (\underline{\mathcal{U}})_l \\ (\bar{\mathcal{U}})_l, & \text{otherwise} \end{cases}$
 - 4: **return** Output of AdaRES with inputs $\epsilon, L = 1/(\sqrt{m}\|Q_i\|_{\infty}), \Pi_{\mathcal{U}}$, and $\nabla l(z_k) = Q_i z_k + q_i$
-

VI. NUMERICAL SIMULATIONS

We compare DaTaControl with existing data-driven control algorithms CGP-LCB [9], [31], C2Opt [8], DeePC [7],

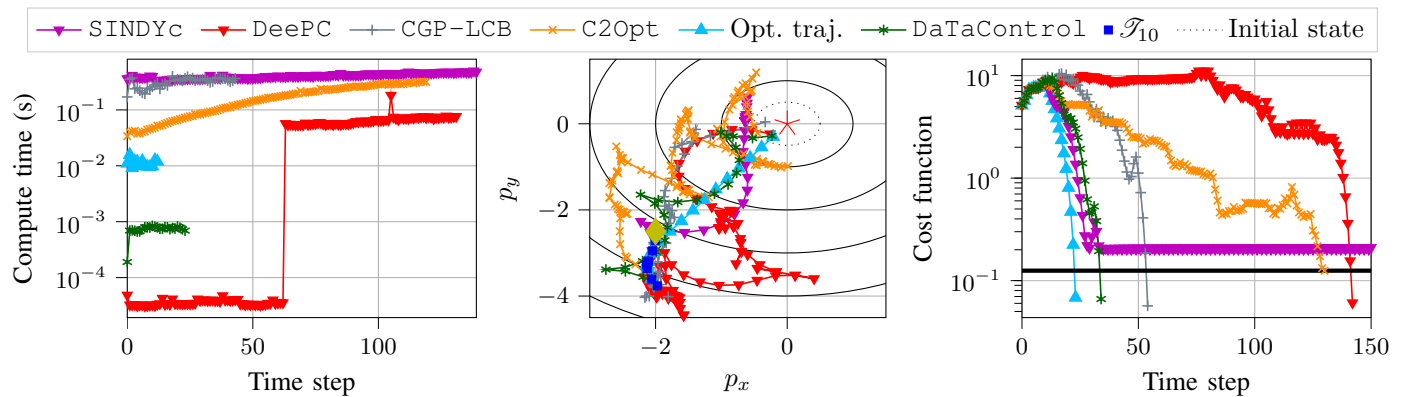


Fig. 4. DaTaControl outperforms state-of-the-art data-driven approaches in terms of computation and optimality of the computed control when applied to a scenario with no noise in the measurements. Opt. traj. corresponds to the one-step optimal control with the known dynamics, and the solid black line on the far right figure represents the optimality threshold for reaching the goal. We emphasize that the extremely low computation time of DeePC at the beginning is due to randomly generated control inputs until the satisfaction of the persistent excitation condition.

and SINDYc [6]. These algorithms also solve the one-step optimal control problem (10) approximately. CGP-LCB assumes that the unknown one-step approximate cost function $C(x_i, u_i) = c(x_i, u_i, x(t_{i+1}; x_i, u_i))$ to minimize is described by a Gaussian process, while C2Opt assumes that C has Lipschitz continuous gradients. On the other hand, SINDYc uses the limited data to perform *sparse identification* of the dynamics over a library of nonlinear functions, which then permits an approximate solution to (10) via numerical solvers. DeePC directly computes control values by solving an N -step model predictive control problem without the system identification step. The experiments on the unicycle system, a quadrotor, and an aircraft system demonstrate that DaTaControl outperforms these algorithms both on the computation time at each sampling time, the robustness to noise, and the suboptimality of the control. We provide the code for reproducibility of the results of this paper at the link: <https://github.com/wuwushrek/datacontrolreach.git>.

A. The Unicycle System

We consider the problem of driving the unicycle of Example 2 to the origin. We encode this control objective using the one-step cost function $c(x_i, u_i, x_{i+1}) = 0.5\|x_{i+1}\|_2^2$. We chose the time step size $\Delta t = 0.1s$ and generated a random initial trajectory \mathcal{T}_{10} starting from the state $x_0 = [-2, -2.5, \pi/2]$ and such that the unicycle goes away from the target. Note that the time step size $\Delta t = 0.1s$ is within the limit enforced by (20) as $1/(\sqrt{n}\beta_{\text{inf}}) \approx 0.123$. Hence, we used the a priori rough enclosure given by Theorem 2.

We applied DaTaControl with the side information discussed in Example 2. We use DaTaControl with Algorithm 5 as the baseline to solve the idealistic control problem. The weights w^+ and w^- were chosen randomly, and $\mu_0 = 1.0$. We used the default parameters in GPYOpt [31] when implementing CGP-LCB. We chose the Lipschitz constant for the gradient $L = 10$ and trade-off hyperparameter $\alpha = 1/2$ for C2Opt [8]. For SINDYc, we considered monomials (up to degree 6), sines and cosines of the state, and the products of these functions with the velocity v and the turning rate ω as the library functions. To perform sparse identification, we swept the regularization parameter [6] $\lambda \in \{10^p : p \in \mathbb{N}_{[-6,5]}\}$

and rounded-down the coefficients smaller than 10^{-3} to zero. For DeePC, we use the data-driven model predictive control scheme (8) in [7] with the slack variable and the regularization in the cost to account for the nonlinearities in the unicycle dynamics. After extensive tuning, we use the parameters $\lambda_y = 10^4$, $\lambda_g = 1$, $T_{\text{ini}} = 8$, and the horizon control $N = 10$. We note that finding working hyperparameters for nonlinear dynamics was a hard task and the hyperparameters did not work for different initializations of the unicycle. We relaxed the target state to the 0.1-sublevel set of the one-step cost function as the stopping criteria for the algorithms.

Figure 4 shows the trajectories and the evolutions of the cost functions for the different algorithms under noiseless measurements. DaTaControl performs significantly better than GPYOpt, SINDYc, DeePC, and C2Opt. It reaches the origin in fewer time steps and significantly lower computation time. Instead, SINDYc failed as it gets stuck close to the origin and is unable to further improve its one-step cost function.

Additionally, we compare DaTaControl with the data-driven baselines under the same conditions as the previous experiment but with different levels of noise in the measurements of the state and the state derivatives. Figure 5 shows that DaTaControl has improved robustness to noise.

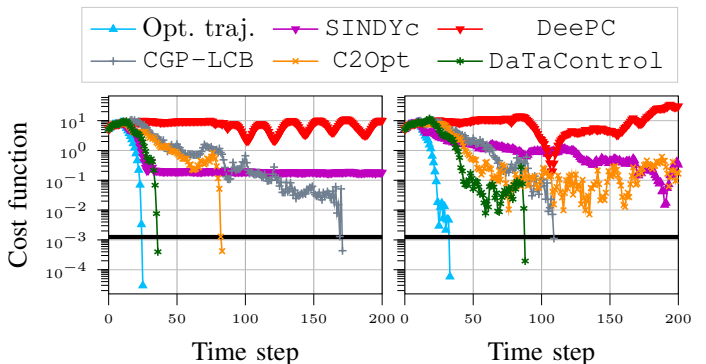


Fig. 5. DaTaControl outperforms the baselines in terms of optimality and robustness when applied to scenario with different levels of noise in the measurements. On the left figure, the noise on the measurements of the state and the state derivative is from a uniform distribution with bounds $\eta = [0.01, 0.01, 0.01]$ and $\bar{\eta} = [0.1, 0.1, 0.1]$, respectively. On the right figure, we have the bound on the noise $\eta = [0.1, 0.1, 0.05]$ and $\bar{\eta} = [1, 1, 0.5]$.

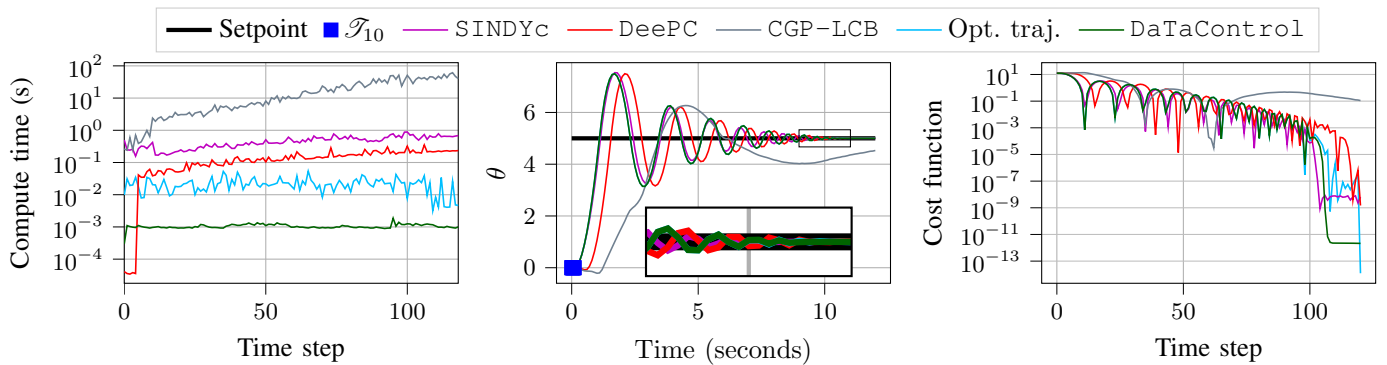


Fig. 6. DaTaControl outperforms the baselines in terms of computation time and optimality of the control even on a linear system in a noise-free scenario.

B. Damaged Aircraft System

In this section, we consider the problem of controlling an aircraft with nonlinearities introduced as damage in the pitch.

$$\begin{aligned}\dot{x}_1 &= -0.021x_1 + 0.122x_2 - 0.322x_3 + 0.01u_1 + u_2, \\ \dot{x}_2 &= -0.209x_1 - 0.53x_2 + 2.21x_3 - 0.064u_1 - 0.044u_2, \\ \dot{x}_3 &= 0.017x_1 + 0.01 \cos(x_1)x_1 - 0.164x_2 + 0.15 \sin(x_1)x_2, \\ &\quad - 0.421x_3 - 0.378u_1 + 0.544u_2 + 0.5 \sin(x_2)u_2, \\ \dot{x}_4 &= x_3, \quad \dot{x}_5 = -x_2 + 2.21x_4.\end{aligned}$$

The model above is based on the linearized dynamics of an undamaged Boeing 747 in landing configuration at sea level [32], [33], where the damages are introduced via nonlinear factors in the dynamics [13]. The state of the aircraft is given by $x = [w_l, w_v, q, \theta, h]$ and the control input by $u = [\delta_e, \delta_t]$. w_l is the deviation from the steady-state longitudinal speed, w_v the downwards-pointing vertical speed, q the pitch rate, θ the pitch angle and h the altitude. The control inputs δ_e and δ_t are the deviations from the elevator and thrust values.

The introduced damages emulate damages to the horizontal stabilizer of the aircraft, which result in the pitch rate behaving more erratically. We seek to control the pitch angle to a setpoint $\theta^{\text{sp}} = 5$ while assuming that the dynamics are unknown. We encode this control objective using the cost function $c(x_i, u_i, x_{i+1}) = 0.5((x_{i+1})_4 - \theta^{\text{sp}})^2$. We chose the time step size $\Delta t = 0.01s$ and generated a random initial trajectory \mathcal{T}_{10} starting from the state $x_0 = [0, 0, 0, 0, 100.0]$. We used DaTaControl, CGP-LCB with the same hyperparameters as for the unicycle dynamics. For SINDYc, we considered monomials (up to degree 2), sines and cosines of the state. We did not investigate C2Opt due to the inability to compute the gradient of C and the fact that it is a strong assumption to assume measurements of such gradient. For DaTaControl, we consider the elementary side information $\dot{\theta} = q$. Furthermore, we consider that $G(x) = G(q, w_v)$ and $f(x) = f(w_l, w_v, q, \theta)$ as the longitudinal speed and the altitude do not directly affect the derivative of the states. We considered the loose Lipschitz bounds $L_f = [0.4, 3, 4, 1, 3]$, $L_{G_{3,2}} = 0.5$, and $L_{G_{k,l}} = 0.01$ otherwise.

We first consider a linear version of the aircraft dynamics where we remove the nonlinear terms $\sin(x_1)$, $\cos(x_1)$, and $\sin(x_2)$ introduced for the damage. The objective here is to show that DaTaControl is more performant on linear systems than state-of-the-art approaches based on behavioral

systems theory foundation, e.g., DeePC. For linear systems, DeePC does not use slack variables and considers the parameters $\lambda_y = 0$, $\lambda_g = 0$ and $T_{\text{ini}} = 8$ as instructed in [7]. Figure 6 shows the near-optimality of DaTaControl, while CGP-LCB fails to stabilize the pitch in a noise-free scenario.

Then, we consider the nonlinear aircraft dynamics above with bounded noise on the measurements on the state and the state derivatives. In this scenario, we use the parameters $\lambda_y = 10^3$, $\lambda_g = 10$ to account for nonlinearities when using DeePC. Figure 7 shows that DeePC is less performant on the nonlinear dynamics, while SINDYc and DeePC fail when there is noise in the measurements. We did not compare with CGP-LCB due to its extremely high computation time for solving the control problem at each time step.

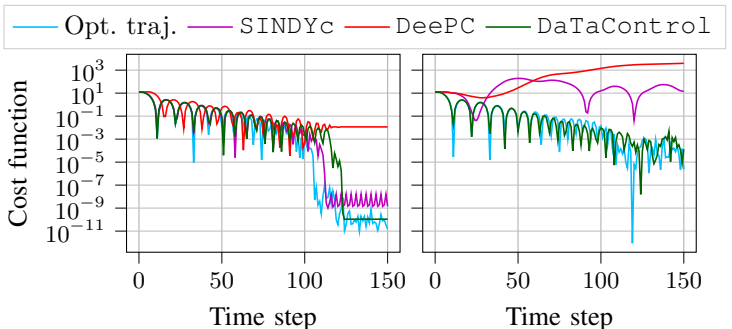


Fig. 7. DaTaControl outperforms the baselines in terms of optimality and robustness when applied to scenarios with different levels of noise in the measurements of the nonlinear aircraft dynamics. On the left figure, we consider a scenario with noise-free measurements. On the right figure, the noise is from a uniform distribution with the bound $\eta = \bar{\eta} = [0.01, 0.01, 0.01, 0.01, 0.1]$.

C. Quadrotor System

Consider a quadrotor with control-affine dynamics [34]

$$\begin{aligned}\dot{p}_x &= v_x, & \dot{v}_x &= -\frac{1}{m}C_D^v v_x - \frac{T_1}{m} \sin\phi - \frac{T_2}{m} \sin\phi, \\ \dot{p}_y &= v_y, & \dot{v}_y &= -\frac{1}{m}(mg + C_D^v v_y) + \frac{T_1}{m} \cos\phi + \frac{T_2}{m} \cos\phi, \\ \dot{\phi} &= \omega, & \dot{\omega} &= -\frac{1}{2I_{yy}}C_D^\phi \omega - \frac{l}{2I_{yy}}T_1 + \frac{l}{2I_{yy}}T_2,\end{aligned}$$

where the components of the state $x = [p_x, v_x, p_y, v_y, \phi, \omega]$ represent, respectively, the horizontal position, horizontal velocity, vertical position, vertical velocity, pitch angle, and pitch rate. The components of the control $u = [T_1, T_2]$ represent the thrust exerted on either end of the quadrotor. We chose the

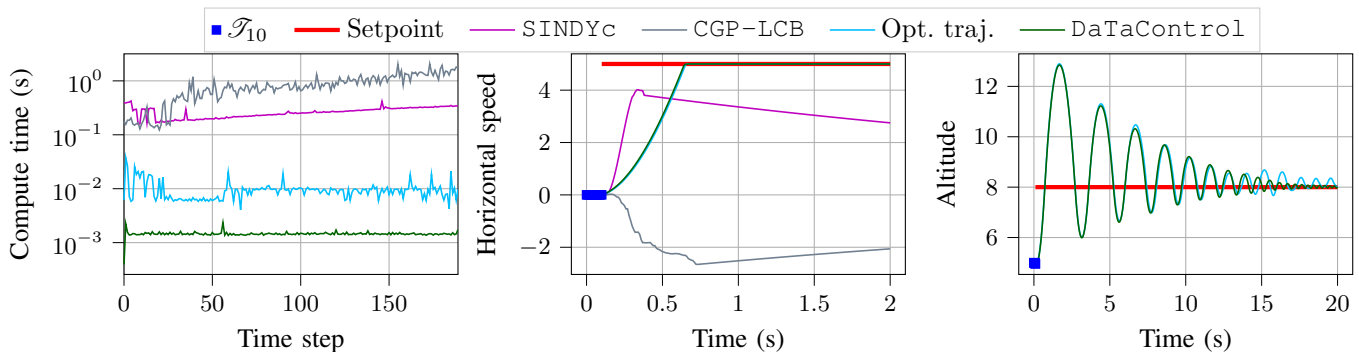


Fig. 8. CGP-LCB and SINDYc fail in the task of controlling the horizontal speed v_x of the quadrotor whereas DaTaControl reaches both setpoints.

constraint set $\mathcal{U} = [0, 18.4] \times [0, 18.4]$. The constants of the dynamics are given by $C_D^v = 0.25$, $C_D^\phi = 0.02255$, $g = 9.81$, $m = 1.25$, $l = 0.5$, and $I_{yy} = 0.03$.

We assume that the dynamics of the quadrotor are unknown and consider the problems of controlling v_x to a setpoint $v_x^{\text{sp}} = 5$ and p_y to a setpoint $p_y^{\text{sp}} = 8$. We encode these control objectives, respectively, using the cost functions $c_1(x_i, u_i, x_{i+1}) = 0.5((x_{i+1})_2 - v_x^{\text{sp}})^2$ and $c_2(x_i, u_i, x_{i+1}) = 0.5((x_{i+1})_3 - p_y^{\text{sp}})^2$. We chose the time step size $\Delta t = 0.01s$ and generated a random initial trajectory \mathcal{T}_{10} starting from the state $x_0 = [0, 0, 5, 0, 0, 0]$. We consider the same hyperparameters for Algorithm 5 as in the unicycle case. We applied DaTaControl with the side information $\dot{p}_x = v_x$, $\dot{p}_y = v_y$, and $\dot{\phi} = \omega$. Such extra knowledge is obtained from elementary laws of physics. Furthermore, DaTaControl considered the loose Lipschitz bounds $L_{f_2} = L_{f_4} = 0.3$, $L_{f_6} = 0.9$, $L_{G_{6,1}} = L_{G_{6,2}} = 0.01$, $L_{G_{2,1}} = L_{G_{2,2}} = L_{G_{4,1}} = L_{G_{4,2}} = L_{G_{6,1}} = L_{G_{6,2}} = 0.9$, and uses the side information $G(x) = G(\phi)$ and $f(x) = f(v_x, v_y, w)$. We used the default parameters of GPYOpt when implementing CGP-LCB. For SINDYc, we considered monomials (up to degree 1), sines and cosines of the state, and the products of these functions with the T_1 and T_2 as the library functions. We did not investigate C2Opt due to the inability to compute the gradient of C .

Figure 8 shows the near-optimality of DaTaControl while CGP-LCB and SINDYc fail to reach the setpoints. Furthermore, the figure demonstrates that DaTaControl can achieve near-optimal, near-real-time control of the vertical position and horizontal speed. We justify the suboptimality of Opt. traj. by the fact that the one-step optimal control problem is highly nonlinear, which makes possible the synthesis of local optimum solutions by the numerical solvers.

VII. CONCLUSION

We developed two data-driven algorithms, DaTaReach and DaTaControl, for on-the-fly over-approximation of the reachable set and constrained near-optimal, real-time control of systems with unknown dynamics. These algorithms are suitable for scenarios with severely limited noisy data and can take advantage of various side information on the underlying a priori unknown dynamics. The numerical experiments demonstrate the efficacy of the algorithms over existing approaches both in terms of the suboptimality of the control, robustness to noise, and the computation time to synthesize control values.

REFERENCES

- [1] M. Ahmadi, A. Israel, and U. Topcu, "Safe controller synthesis for data-driven differential inclusions," *IEEE Trans. Autom. Control*, 2020.
- [2] M. Berz and K. Makino, "Verified integration of odes and flows using differential algebraic methods on high-order Taylor models," *Reliable computing*, vol. 4, no. 4, pp. 361–369, 1998.
- [3] N. S. Nedialkov, K. R. Jackson, and G. F. Corliss, "Validated solutions of initial value problems for ordinary differential equations," *Applied Mathematics and Computation*, vol. 105, no. 1, pp. 21–68, 1999.
- [4] E. Goubault and S. Putot, "Forward inner-approximated reachability of non-linear continuous systems," in *Proceedings of the 20th International Conference on Hybrid Systems: Computation and Control*, ser. HSCC '17. New York, NY, USA: Association for Computing Machinery, 2017, p. 1–10. [Online]. Available: <https://doi.org/10.1145/3049797.3049811>
- [5] D. Q. Mayne, J. B. Rawlings, C. V. Rao, and P. O. Scokaert, "Constrained model predictive control: Stability and optimality," *Automatica*, vol. 36, no. 6, pp. 789–814, 2000.
- [6] E. Kaiser, J. N. Kutz, and S. L. Brunton, "Sparse identification of nonlinear dynamics for model predictive control in the low-data limit," *Proc. of the Royal Society A*, vol. 474, no. 2219, p. 20180335, 2018.
- [7] J. Coulson, J. Lygeros, and F. Dörfler, "Data-enabled predictive control: In the shallows of the deepc," *2019 18th European Control Conference (ECC)*, pp. 307–312, 2019.
- [8] A. P. Vinod, A. Israel, and U. Topcu, "Convexified contextual optimization for on-the-fly control of smooth systems," in *2020 American Control Conference (ACC)*, 2020, pp. 2004–2011.
- [9] A. Krause and C. S. Ong, "Contextual gaussian process bandit optimization," in *Advances in NIPS.*, 2011, pp. 2447–2455.
- [10] F. Djeumou, A. P. Vinod, E. Goubault, S. Putot, and U. Topcu, "On-the-fly control of unknown smooth systems from limited data," *arXiv preprint arXiv:2009.12733*, 2020.
- [11] M. Korda and I. Mezić, "Linear predictors for nonlinear dynamical systems: Koopman operator meets model predictive control," *Automatica*, vol. 93, pp. 149–160, 2018.
- [12] J. L. Proctor, S. L. Brunton, and J. N. Kutz, "Dynamic mode decomposition with control," *SIAM Journal on Applied Dynamical Systems*, vol. 15, no. 1, pp. 142–161, 2016.
- [13] M. Ornik, S. P. Carr, A. Israel, and U. Topcu, "Control-oriented learning on the fly," *IEEE Trans. Autom. Control*, 2019.
- [14] J. Berberich, J. Köhler, M. A. Müller, and F. Allgöwer, "Data-driven model predictive control with stability and robustness guarantees," *IEEE Transactions on Automatic Control*, vol. 66, no. 4, pp. 1702–1717, 2020.
- [15] J. Berberich, C. W. Scherer, and F. Allgöwer, "Combining prior knowledge and data for robust controller design," *arXiv preprint arXiv:2009.05253*, 2020.
- [16] I. Markovsky and F. Dörfler, "Data-driven dynamic interpolation and approximation," *Vrije Univ. Brussel, Tech. Rep.*, 2021.
- [17] H. J. van Waarde, M. K. Camlibel, and M. Mesbahi, "From noisy data to feedback controllers: non-conservative design via a matrix s-lemma," *IEEE Transactions on Automatic Control*, 2020.
- [18] H. J. Van Waarde, J. Eising, H. L. Trentelman, and M. K. Camlibel, "Data informativity: a new perspective on data-driven analysis and control," *IEEE Transactions on Automatic Control*, vol. 65, no. 11, pp. 4753–4768, 2020.
- [19] J. Willems, I. Markovsky, P. Rapisarda, and B. De Moor, "A note on persistency of excitation," in *2004 43rd IEEE Conference on Decision and Control (CDC) (IEEE Cat. No.04CH37601)*, vol. 3, 2004, pp. 2630–2631 Vol.3.

- [20] A. Devonport and M. Arcak, "Data-driven reachable set computation using adaptive gaussian process classification and monte carlo methods," in *2020 American Control Conference (ACC)*. IEEE, 2020, pp. 2629–2634.
- [21] S. Haesaert, P. M. Van den Hof, and A. Abate, "Data-driven and model-based verification via bayesian identification and reachability analysis," *Automatica*, vol. 79, pp. 115–126, 2017.
- [22] A. Chakrabarty, A. Raghunathan, S. Di Cairano, and C. Danielson, "Data-driven estimation of backward reachable and invariant sets for unmodeled systems via active learning," in *2018 IEEE Conference on Decision and Control (CDC)*. IEEE, 2018, pp. 372–377.
- [23] R. E. Moore, *Interval analysis*. Prentice-Hall Englewood Cliffs, 1966, vol. 4.
- [24] E. Goubault and S. Putot, "Inner and outer reachability for the verification of control systems," in *Proc. Hybrid Syst.: Comput. & Ctrl.*, 2019, pp. 11–22.
- [25] J. Kocijan, R. Murray-Smith, C. E. Rasmussen, and A. Girard, "Gaussian process model based predictive control," in *Proceedings of the 2004 American control conference*, vol. 3. IEEE, 2004, pp. 2214–2219.
- [26] B.-G. Park, J.-W. Lee, and W. H. Kwon, "Robust one-step receding horizon control for constrained systems," *International Journal of Robust and Nonlinear Control: IFAC-Affiliated Journal*, vol. 9, no. 7, pp. 381–395, 1999.
- [27] G. Chabert and L. Jaulin, "Contractor programming," *Artificial Intelligence*, vol. 173, no. 11, pp. 1079–1100, 2009. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0004370209000381>
- [28] F. Benhamou, F. Goualard, L. Granvilliers, and J.-F. Puget, "Revising hull and box consistency," in *Proceedings of the 1999 International Conference on Logic Programming*. USA: Massachusetts Institute of Technology, 1999, p. 230–244.
- [29] P. Patrinos and A. Bemporad, "An accelerated dual gradient-projection algorithm for embedded linear model predictive control," *IEEE Transactions on Automatic Control*, vol. 59, no. 1, pp. 18–33, 2013.
- [30] O. Fercoq and Z. Qu, "Adaptive restart of accelerated gradient methods under local quadratic growth condition," *IMA Journal of Numerical Analysis*, vol. 39, no. 4, pp. 2069–2095, 2019.
- [31] The GPyOpt authors, "GPyOpt: A Bayesian optimization framework in Python," <http://github.com/SheffieldML/GPyOpt>, 2016.
- [32] A. E. Bryson, *Control of spacecraft and aircraft*. Princeton university press Princeton, New Jersey, 1993.
- [33] J. Che and D. Chen, "Automatic landing control using h/sub/spl inf/control and stable inversion," in *Proceedings of the 40th IEEE Conference on Decision and Control (Cat. No. 01CH37228)*, vol. 1. IEEE, 2001, pp. 241–246.
- [34] M. Chen, S. L. Herbert, M. S. Vashishtha, S. Bansal, and C. J. Tomlin, "Decomposition of reachable sets and tubes for a class of nonlinear systems," *IEEE Transactions on Automatic Control*, vol. 63, no. 11, pp. 3675–3688, Nov 2018.
- [35] G. Teschl, *Ordinary differential equations and dynamical systems*. American Mathematical Soc., 2012, vol. 140.
- [36] S. Bubeck, "Convex optimization: Algorithms and complexity," *arXiv preprint arXiv:1405.4980*, 2014.
- [37] Z.-Q. Luo and P. Tseng, "On the linear convergence of descent methods for convex essentially smooth minimization," *SIAM Journal on Control and Optimization*, vol. 30, no. 2, pp. 408–425, 1992.

APPENDIX A

In this appendix, we provide the proofs for all lemmas and theorems presented in this paper.

PROOF OF THEOREM 2

First, we provide a bound on the maximum variation of the function h given upper bounds on the Lipschitz constants of f and G .

Lemma 5 (MAXIMUM VARIATION OF h). *Let L_f and L_G be upper bounds on the Lipschitz constants of f and G . For all $x_1, x_2 \in \mathcal{X}$ and $u_1 \in \mathcal{U}$, we have that*

$$\|h(x_1, u_1) - h(x_2, u_1)\|_2 \leq \beta(u_1) \|x_1 - x_2\|_2, \quad (43)$$

where $\beta(u_1) = \sqrt{\sum_{k=1}^n (L_{f_k} + \sum_{l=1}^m L_{G_{k,l}} |(u_1)_l|)^2}$.

Proof. This is a direct application of the definition of the Lipschitz bounds of f_k and $G_{k,l}$. Specifically, we have that

$$\begin{aligned} & |(h(x_1, u_1))_k - (h(x_2, u_1))_k| \\ &= |(f(x_1) - f(x_2))_k + \sum_{l=1}^m ((G(x_1) - G(x_2))_{k,l} (u_1)_l)| \\ &\leq (L_{f_k} + \sum_{l=1}^m L_{G_{k,l}} |(u_1)_l|) \|x_1 - x_2\|_2, \end{aligned} \quad (44)$$

for all $k \in \mathbb{N}_{[1,n]}$. The last inequality is obtained by the definitions of L_{f_k} and $L_{G_{k,l}}$. Hence, we obtain (43). \square

Lemma 6 (GRÖNWALL'S INEQUALITY [35, LEMMA 2.7]). *Let Ψ, α, γ be real-valued functions on $[t_0, T]$ with $t_0, T \in \mathbb{R}$. Suppose Ψ satisfies for all $t \in [t_0, T]$ the inequality*

$$\Psi(t) \leq \alpha(t) + \int_{t_0}^t \gamma(s) \Psi(s) ds, \quad (45)$$

with $\gamma(s) \geq 0$ for all $s \in [t_0, T]$. Then, for all $t \in [t_0, T]$,

$$\Psi(t) \leq \alpha(t) + \int_{t_0}^t \alpha(s) \gamma(s) \exp\left(\int_s^t \gamma(r) dr\right) ds. \quad (46)$$

Next, we combine the results of Lemma 5 and Lemma 6 to provide a bound on the variations of points in a trajectory.

Lemma 7 (VARIATION OF TRAJECTORIES OF (11)). *Let x be a continuous-time signal satisfying (11) for a given control signal $u \in \mathbb{V}$. For all $t \in [t_i, t_i + \Delta t]$, we have that*

$$\|x(t) - x(t_i)\|_2 \leq \|\alpha_i\|_2 \beta_i^{-1} (e^{\beta_i \Delta t} - 1), \quad (47)$$

where the parameters $\beta_i \in \mathbb{R}_+$ and $\alpha_i \in \mathbb{R}^n$ are given by

$$\beta_i = \sqrt{\sum_{k=1}^n \left(L_{f_k} + \sum_{l=1}^m L_{G_{k,l}} |(\mathbf{v}([t_i, t_i + \Delta t]))_l| \right)^2}, \quad (48)$$

$$\alpha_i = f(x(t_i)) + |G(x(t_i))| |\mathbf{v}([t_i, t_i + \Delta t])|. \quad (49)$$

Proof. For all $t \in \mathcal{T}_i = [t_i, t_i + \Delta t]$, we have that

$$\begin{aligned} & \|x(t) - x(t_i)\|_2 \\ &= \left\| \int_{t_i}^t h(x(s), u(s)) ds \right\|_2 \end{aligned} \quad (50)$$

$$\begin{aligned} &\leq \int_{t_i}^t \|h(x(s), u(s)) - h(x(t_i), u(s))\|_2 ds \\ &\quad + \int_{t_i}^t \|h(x(t_i), u(s))\|_2 ds \end{aligned} \quad (51)$$

$$\leq \int_{t_i}^t \beta(u(s)) \|x(s) - x(t_i)\|_2 ds + \int_{t_i}^t \|h(x(t_i), u(s))\|_2 ds \quad (52)$$

$$\leq \|\alpha_i\|_2 (t - t_i) + \int_{t_i}^t \beta_i \|x(s) - x(t_i)\|_2 ds. \quad (53)$$

We obtain (50) since x is a solution of (11). The passage from (50) to (51) results from applying the triangle inequality. We use Lemma 5 with $x_1 = x(s)$, $x_2 = x(t_i)$, and $u_1 = u(s)$ to obtain (52). We obtain (53) by showing that the inequalities $\beta(u(s)) \leq \beta_i$ and $\|h(x(t_i), u(s))\|_2 \leq \|\alpha_i\|_2$ hold for all

$s \in \mathcal{T}_i$. Specifically, the former inequality holds by upper-bounding $|(u(s))_l|$ in $\beta(u(s))$ with $|\mathbf{v}([t_i, t_i + \Delta t])_l| \geq \sup_{s \in \mathcal{T}_i} |(u(s))_l|$. The latter inequality holds since

$$\begin{aligned} & |(h(x(t_i), u(s)))_k| \\ & \leq |(f(x(t_i)))_k| + \sum_{l=1}^m |(G(x(t_i)))_{k,l}| \sup_{s \in \mathcal{T}_i} |(u(s))_l| \leq |(\alpha_i)_k|. \end{aligned}$$

The inequality (53) satisfies the condition (46) of Lemma 6 with $\Psi(t) = \|x(t) - x(t_i)\|_2$, $\alpha(t) = \|\alpha_i\|_2(t - t_i)$, and $\gamma(t) = \beta_i$. Thus, we have that

$$\begin{aligned} \|x(t) - x(t_i)\| & \leq \|\alpha_i\|_2(t - t_i) \\ & \quad + \beta_i \|\alpha_i\|_2 \int_{t_i}^t (s - t_i) e^{\beta_i(t-s)} ds. \end{aligned} \quad (54)$$

By integration by parts, we have that

$$\begin{aligned} \int_{t_i}^t (s - t_i) e^{\beta_i(t-s)} ds & = -\frac{t - t_i}{\beta_i} + \frac{1}{\beta_i} \int_{t_i}^t e^{\beta_i(t-s)} ds \\ & = -\frac{t - t_i}{\beta_i} + \frac{e^{\beta_i(t-t_i)} - 1}{\beta_i^2}. \end{aligned} \quad (55)$$

Finally, by combining (55) and (54), we obtain (47). \square

Finally, we provide the proof of Theorem 2 below.

The expression (19) is derived by scaling adequately the bound (47) from Lemma 7. Specifically, we seek for an a priori rough enclosure \mathcal{S}_i such that

$$\mathcal{S}_i = \mathcal{R}_i^+ + \mu \frac{\bar{\alpha}_i}{\beta_i} (e^{\beta_i \Delta t} - 1) [-1, 1]^n, \quad (56)$$

where $\mu > 0$ is a parameter to find in order for \mathcal{S}_i to satisfy the fixed-point equation (17) and $\bar{\alpha}_i$ is given by

$$\bar{\alpha}_i = \sup_{x(t_i) \in \mathcal{R}_i^+} \|f(x(t_i)) + |G(x(t_i))| |\mathbf{v}([t_i, t_i + \Delta t])\|_2.$$

We over-approximate the set $\mathcal{R}(h, \mathcal{S}_i \times \mathbf{v}([t_i, t_i + \Delta t]))$ as a function of Δt , $\beta_{t_i, t_i + \Delta t}$, and $\mathbf{h} = \mathbf{f} + \mathbf{G}u$. For all $x_i \in \mathcal{R}_i^+$, $u_i \in \mathbf{v}([t_i, t_i + \Delta t])$, and $s_i \in \mathcal{S}_i$, we have that

$$\|h(s_i, u_i) - h(x_i, u_i)\| \leq \beta_i \|s_i - x_i\| \quad (57)$$

$$\leq \sqrt{n} \mu \bar{\alpha}_i (e^{\beta_i \Delta t} - 1). \quad (58)$$

The inequality (57) comes from Lemma 5 with $\beta(u_i)$ upper-bounded by β_i and the definition of \mathcal{S}_i (56) provides an upper bound on $\|s_i - x_i\|$ that yields (58). Additionally, the inequality (58) implies that

$$\begin{aligned} & \mathcal{R}(h, \mathcal{S}_i \times \mathbf{v}([t_i, t_i + \Delta t])) \\ & \subseteq \mathbf{h}(\mathcal{R}_i^+, \mathbf{v}([t_i, t_i + \Delta t])) + \sqrt{n} \mu \bar{\alpha}_i (e^{\beta_i \Delta t} - 1) [-1, 1]^n. \end{aligned}$$

Hence, \mathcal{S}_i from (56) solves the fixed-point equation (17) if

$$\begin{aligned} & [0, \Delta t] \left(\mathbf{h}(\mathcal{R}_i^+, \mathbf{v}([t_i, t_i + \Delta t])) + \sqrt{n} \mu \bar{\alpha}_i (e^{\beta_i \Delta t} - 1) [-1, 1]^n \right) \\ & \subseteq \mu \frac{\bar{\alpha}_i}{\beta_i} (e^{\beta_i \Delta t} - 1) [-1, 1]^n. \end{aligned} \quad (59)$$

For notation brevity, let $c_1 = \bar{\alpha}_i (e^{\beta_i \Delta t} - 1)$. Observe that $[0, \Delta t] [\underline{\mathcal{A}}, \bar{\mathcal{A}}] = [\min(0, \underline{\mathcal{A}}), \max(0, \bar{\mathcal{A}})]$. We use the observation to find $\mu > 0$ such that the inclusion (59) holds. That is, the inequalities

$$\begin{aligned} & \Delta t \left(\overline{\mathbf{h}(\mathcal{R}_i^+, \mathbf{v}([t_i, t_i + \Delta t]))}_k + \sqrt{n} \mu c_1 \right) \leq \frac{\mu c_1}{\beta_i} \\ & \iff \left(\frac{1}{\Delta t \beta_i} - \sqrt{n} \right) \mu \geq \frac{1}{c_1} \overline{\mathbf{h}(\mathcal{R}_i^+, \mathbf{v}([t_i, t_i + \Delta t]))}_k \end{aligned}$$

and

$$\begin{aligned} & \Delta t \left(\underline{\mathbf{h}(\mathcal{R}_i^+, \mathbf{v}([t_i, t_i + \Delta t]))}_k - \sqrt{n} \mu c_1 \right) \geq -\frac{\mu c_1}{\beta_i} \\ & \iff \left(\sqrt{n} - \frac{1}{\Delta t \beta_i} \right) \mu \geq \frac{1}{c_1} \underline{\mathbf{h}(\mathcal{R}_i^+, \mathbf{v}([t_i, t_i + \Delta t]))}_k \end{aligned}$$

hold for all $k \in \mathbb{N}_{[1, n]}$. Therefore, for a step size Δt satisfying (18), μ given by

$$\mu = \frac{\|\mathbf{f}(\mathcal{R}_i^+) + \mathbf{G}(\mathcal{R}_i^+) \mathbf{v}([t_i, t_i + \Delta t])\|_\infty}{c_1 \left(\frac{1}{\Delta t \beta_i} - \sqrt{n} \right)}$$

satisfies the above inequalities. Thus, the inclusion (59) holds and the set \mathcal{S}_i is solution of the fixed-point equation (17). By replacing μ in \mathcal{S}_i given by (56), we obtain (19). \square

PROOF OF LEMMA 3

We use the interval arithmetic to characterize exactly the belonging relations $x_{i+1} \in \mathcal{B}_i + \mathcal{A}_i^+ u_i$ and $x_{i+1} \in \mathcal{B}_i + \mathcal{A}_i^- u_i$ in term of linear constraints. Specifically, since $(\mathcal{A}_i^+ u_i)_k = \sum_{l=1}^m (\mathcal{A}_i^+)_{k,l} (u_i)_l$, interval arithmetic provides upper and lower bound on each term $(\mathcal{A}_i^+)_{k,l} (u_i)_l$ as follows

$$\begin{cases} (\mathcal{A}_i^+)_{k,l} (u_i)_l = [(\underline{\mathcal{A}_i^+})_{k,l} (u_i)_l, \overline{(\mathcal{A}_i^+)}_{k,l} (u_i)_l], & \text{if } (u_i)_l \geq 0 \\ (\mathcal{A}_i^+)_{k,l} (u_i)_l = [(\overline{\mathcal{A}_i^+})_{k,l} (u_i)_l, \underline{(\mathcal{A}_i^+)}_{k,l} (u_i)_l], & \text{otherwise.} \end{cases}$$

We can deduce that $\overline{\mathcal{A}_i^+ u_i} = \mathcal{A}_i^{s+} u_i$ and $\underline{\mathcal{A}_i^+ u_i} = \mathcal{A}_i^{l+} u_i$.

Similarly, it is easy to prove that $\overline{\mathcal{A}_i^- u_i} = \mathcal{A}_i^{s-} u_i$ and $\underline{\mathcal{A}_i^- u_i} = \mathcal{A}_i^{l-} u_i$. As a consequence, the optimistic control problem (29) can be immediately reformulated as the convex quadratic programming problem (34).

Additionally, we have that

$$\begin{aligned} & c(x_i, u_i, b_i^{\text{ide}} + A_i^{\text{ide}} u_i) \\ & = u_i^T \left((A_i^{\text{ide}})^T Q A_i^{\text{ide}} + 2(A_i^{\text{ide}})^T S + R \right) u_i \\ & \quad + \left(2(b_i^{\text{ide}})^T (S + Q A_i^{\text{ide}}) + q^T A_i^{\text{ide}} + r^T \right) u_i \\ & \quad + (b_i^{\text{ide}})^T Q b_i^{\text{ide}} + q^T b_i^{\text{ide}} \\ & = 0.5 u_i^T Q_i u_i + q_i^T u_i + p_i. \end{aligned}$$

Hence we obtain the cost function of the idealistic control problem (35). Therefore, it remains to prove that the matrix Q_i is positive semidefinite. Let prove that $u_i^T Q_i u_i \geq 0$ for all $u_i \in \mathbb{R}^m$. Since by assumption, the cost function is convex, then the matrix implied in the quadratic term is positive semidefinite. Hence, we can deduce that

$$\begin{bmatrix} A_i^{\text{ide}} u_i \\ u_i \end{bmatrix}^T \begin{bmatrix} Q & S \\ S^T & R \end{bmatrix} \begin{bmatrix} A_i^{\text{ide}} u_i \\ u_i \end{bmatrix} \geq 0,$$

for all $u_i \in \mathbb{R}^m$. Expanding the expression above immediately provides that $u_i^T Q_i u_i \geq 0$ for all $u_i \in \mathbb{R}^m$. \square

PROOF OF THEOREM 6

First, using the notation of Lemma 4, we have that $l(y) = 0.5y^T Q_i y + q_i^T y + p_i$ and $\mathcal{P} = \mathcal{U}$ for the idealistic control problem. Thus, l satisfies condition (39) with any parameter $L \geq \lambda_{\max}(Q_i)$ where $\lambda_{\max}(Q_i)$ is the maximum eigen value of Q_i . Recall that since we have

$$\begin{aligned} \|\nabla l(x) - \nabla l(y)\|_2^2 &= \|Q_i x - Q_i y\|_2^2 = (x - y)^T Q_i^T Q_i (x - y) \\ &\leq \lambda_{\max}^2(Q_i) \|x - y\|_2^2, \end{aligned}$$

then Lemma 3.4 in [36] implies that the condition (39) holds. Since $\lambda_{\max}(Q_i) \leq \sqrt{m} \|Q_i\|_\infty$, we pick $L = \sqrt{m} \|Q_i\|_\infty$. The local error bound assumption holds due to Theorem 2.1 in [37] where the authors prove that for a function h satisfying $h(x) = g(Ax) + c^T x$ on a convex polyhedra \mathcal{P} with g strongly convex, there exists $\mu_{\mathcal{P}}(x_0)$ for any x_0 such that the condition (40) holds. As a consequence, since we assume that $\mu_0 \leq \mu_{\mathcal{P}}(x_0)$, Lemma 4 provides that the number of iterations \hat{N} required by AdaRES is upper bounded as follows

$$\hat{N} \leq \left\lceil 2\sqrt{\frac{e}{\mu_0}} - 1 \right\rceil \left\lceil \ln \frac{16(2c_{\max})}{\epsilon\mu_0} \right\rceil = N_{\max}.$$

As a consequence, since the computation of $\Pi_{\mathcal{P}}(\cdot)$ is simple and requires at most $3m$ comparisons, and ∇l requires $2m^2 + m$ flops, we can deduce that the maximum number of flops required by AdaRES to terminate is roughly given by $N_{\max}(2m^2 + 16m + 12)$. Such a number is obtained by considering the worst case where each line of Algorithm 4 is executed N_{\max} times even though in reality, only lines 9–12 of AdaRES will be executed maximum N_{\max} times.

Finally, since computing \mathbf{f} and \mathbf{G} requires $N(8n + 9)$ and $N(8n + 9)m$ respectively, we can deduce that, without any side information 1–4, lines 1–9 of DaTaControl require at most $2N(8n + 9)(m + 1) + 16n^2m + 8n^2 + 2m^2 + 6nm + 24n + 8$. Specifically, it is easy to check that \mathcal{S}_i using Theorem 2 can be computed using $6mn + 6n + 4$ flops. The matrices \mathcal{B}_i , \mathcal{A}_i^+ and \mathcal{A}_i^- can be computed using $4n^2 + 6n$ flops for \mathcal{B}_i and $8n^2m + 3nm + m^2$ flops for each \mathcal{A}_i^+ and \mathcal{A}_i^- . Hence, since computing $\mathcal{A}_i^{\text{ide}}$ and \mathcal{B}^{ide} requires $8n + 8nm$ flops, we can compute Q_i and q_i using $6n^2m + 6m^2$ and $(2n^2 + 4n + 2)m$ flops, respectively. Thus, after adding the number of flops for AdaRES, lines 1–9 of DaTaControl, $\mathcal{A}_i^{\text{ide}}$ and \mathcal{B}^{ide} , and Q_i and q_i , we obtain the bound N^{ide} . \square

ILLUSTRATION FOR LEMMA 1

Consider the unicycle system of Example 2, and assume that $(G)_{1,2}$ is unknown. Given the data point (x_i, \dot{x}_i, u_i) where $x_i = [0, 0, \pi/2]$, $u_i = [1, 0.1]$ and $\dot{x}_i = [0, 1, 0.1]$, and the over-approximating ranges $(\mathcal{F}_i)_1 = [-0.01, 1]$, $(\mathcal{G}_i)_{1,1} = [-0.05, 0.05]$, and $(\mathcal{G}_i)_{1,2} = [-0.1, 1]$ of $(f(x_i))_1 = 0$, $(G(x_i))_{1,1} = 0$, and $(G(x_i))_{1,2} = 0$. By Lemma 1, we have

$$\begin{aligned} (C_{\mathcal{F}_i})_1 &= [-0.01, 1] \cap (0 - [-0.05, 0.05] - 0.1[-0.1, 1]) \\ &= [-0.01, 1] \cap [-0.15, 0.06] = [-0.01, 0.06]. \end{aligned}$$

Hence, $(s_0)_1 = [-0.06, 0.01] \cap [-0.06, 0.15] = [-0.06, 0.01]$ and the interval $(C_{\mathcal{G}_i})_{1,1}$ is given by

$$\begin{aligned} (C_{\mathcal{G}_i})_{1,1} &= (([-0.06, 0.01] - 0.1[-0.1, 1]) \cap [-0.05, 0.05]) \\ &= [-0.16, 0.02] \cap [-0.05, 0.05] = [-0.05, 0.02]. \end{aligned}$$

Moreover, since

$$\begin{aligned} (s_1)_1 &= ([-0.06, 0.01] - [-0.05, 0.02]) \cap (0.1[-0.1, 1]) \\ &= [-0.08, 0.06] \cap [-0.01, 0.1] = [-0.01, 0.06], \end{aligned}$$

the interval $(C_{\mathcal{G}_i})_{1,2}$ is given by

$$(C_{\mathcal{G}_i})_{1,2} = 10([-0.01, 0.06] \cap (0.1[-0.1, 1])) = [-0.1, 0.6].$$

Therefore, Lemma 1 provides the contractions $(C_{\mathcal{F}})_1 \subset (\mathcal{F}_i)_1$, $(C_{\mathcal{G}_i})_{1,1} \subset (\mathcal{G}_i)_{1,1}$, and $(C_{\mathcal{G}_i})_{1,2} \subset (\mathcal{G}_i)_{1,2}$. \square