

Efficient Norm-Based Reachable Sets via Iterative Dynamic Programming

Akash Harapanahalli and Samuel Coogan¹

Abstract—In this work, we present a numerical optimal control framework for reachable set computation using *normotopes*, a new set representation as a norm ball with a shaping matrix. In reachable set computations, we expect to continuously vary the shape matrix along trajectories. Incorporating the shape dynamics as an input, we build a *controlled embedding system* using a linear differential inclusion bounding the dynamics of the system, where a single forward simulation of this embedding system always provides an overapproximating reachable set of the system, no matter the choice of *hypercontrol*. By iteratively solving a linear quadratic approximation of the nonlinear optimal hypercontrol problem, we synthesize less conservative final reachable sets, providing a natural tradeoff between runtime and accuracy. Terminating our algorithm at any point always returns a valid overapproximating reachable set.

I. INTRODUCTION

One way to assess the safe operation of a complex nonlinear system under uncertainty is to verify its forward reachable set. For nonlinear systems, the reachable set is often too complex to determine in closed form—instead, various algorithmic methods have been developed to compute guaranteed overapproximations. To combat conservativeness in these algorithmic strategies, different set representations have been developed, which have different tradeoffs between reachable set accuracy and computational efficiency. To name a few, there are zonotopes [1], constrained zonotopes [2], polynomial zonotopes [3], hybrid zonotopes [4], ellipsotopes [5], and Taylor models [6]. For linear systems, any generalized star set [7] can be propagated using only $n + 1$ simulations. Beyond these generator-based approaches, there are many other frameworks for reachable set computation. For instance, level set methods use the Hamilton-Jacobi equations [8], [9], [10] to represent the reachable set as a level set of a solution to a partial differential equation.

From a dual perspective, one can evolve halfspaces in linear systems using the adjoint equation [11]. This dual formulation was recently re-explored in [12], where they consider general parametric set representations defined by the intersection of constraints, instead of ones generated by constrained combinations. As shown in [12], the parameters defining the constraints can be arbitrarily updated as long as the offset term compensates accordingly. This observation is the basic backbone of this work—we directly optimize over

the dynamics of these parameters to attempt to minimize the final volume of our overapproximating reachable set.

Computing reachable sets using norm balls is not a novel approach on its own, and there are many works applying results from contraction theory to compute such overapproximations. For instance, [13], [14] use bounds on logarithmic norms to bloat or shrink norm balls along nominal trajectories. The works [15], [16] use interval analysis to overapproximate the Jacobian matrix to automatically bound the logarithmic norm. More recently, [17] uses a more accurate interval matrix to bound the logarithmic norm by comparing specifically to the nominal trajectory. The advantage of these works is in the simplicity of computing the logarithmic norm, which is often known in closed form—for example, as a simple eigenvalue maximization in the ℓ_2 case, or a maximum row/column sum computation in the ℓ_∞/ℓ_1 cases [18], [19]. However, to our knowledge, existing works fix the norm along a real portion of the trajectory, using semi-definite programming to find a single shaping matrix along that segment. In this work, instead of successively choosing fixed norms for segments of the trajectory using semi-definite programming, we consider the shaping matrix as a continuous curve we can dynamically control, and use iterative dynamic programming along the nominal trajectory to find smoothly varying shaping matrices.

Contributions: In this work, we reformulate the reachable set computation problem as an optimal control problem. First, we propose a simple constraint-based set representation called a *normotope*, which is the sublevel set of a norm with a specified shaping matrix, center, and radius. We build a *controlled embedding system* for the original system using interval-based linear differential inclusions and logarithmic norms, with the observation that we may smoothly vary the shape matrix of the normotope as long as the offset term compensates accordingly—we call this the *hypercontrol*.

Next, we pose the following question: given an initial set for the system, can we find a hypercontrol mapping through the embedding space such that final reachable set has minimal volume? To address this problem, we define an optimal control problem to synthesize hypercontrol policies resulting in small terminal set volume. We show how dynamic programming algorithms like iLQR can trade extra computations to iteratively optimize the accuracy of the reachable set estimate with respect to the hypercontrol. With two case studies, we show how (i) just a few cheap iterations can dramatically improve the reachable set accuracy, and (ii) that the algorithm can learn nontrivial policies which accumulate extra uncertainty earlier in the trajectory to shrink

*This work was supported in part by the National Science Foundation under awards #2219755, #2333488, and #2440387, and by the Air Force Office of Scientific Research under Grant FA9550-23-1-0303.

¹Akash Harapanahalli and Samuel Coogan are with the School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA, 30318, USA. {aharapan,sam.coogan}@gatech.edu

the final overapproximating reachable set.

II. BACKGROUND AND MATHEMATICAL PRELIMINARIES

A. Notations

Let $\|\cdot\|$ denote a norm. Let I denote the identity matrix of appropriate dimension and $\mathbf{1}$ denote the vector of all ones of appropriate dimension. For linear operator $A : \mathbb{R}^{n_x} \rightarrow \mathbb{R}^{n_y}$ between normed vector spaces, we set $\|A\|_{x \rightarrow y} = \sup_{x \in \mathbb{R}^{n_x} : \|x\|_{\mathbb{R}^{n_x}} = 1} \|Ax\|_{\mathbb{R}^{n_y}}$ to be the operator norm of A . For linear operator $A : \mathbb{R}^{n_x} \rightarrow \mathbb{R}^{n_x}$, define the induced logarithmic norm $\mu(A) = \limsup_{h \searrow 0} \frac{\|I + hA\|_{x \rightarrow x} - 1}{h}$. $GL(n)$ is the set of invertible $n \times n$ matrices. Let $\text{co } \mathcal{S}$ denote the convex hull of $\mathcal{S}$.

B. Nonlinear Systems and Reachable Sets

We consider nonlinear dynamical systems of the form

$$\dot{x} = f(t, x, w), \quad (1)$$

where $x \in \mathbb{R}^{n_x}$ is the state of the system, $w \in \mathbb{R}^{n_w}$ is a disturbance input to the system, and $f : \mathbb{R}_{\geq 0} \times \mathbb{R}^{n_x} \times \mathbb{R}^{n_w} \rightarrow \mathbb{R}^{n_x}$ is a C^1 smooth function.

Under these assumptions, the system (1) has a unique trajectory starting from any initial condition $x_0 \in \mathbb{R}^{n_x}$ at t_0 , under essentially bounded disturbance map $\mathbf{w} : [t_0, \infty) \rightarrow \mathbb{R}^{n_w}$ defined for some neighborhood of t_0 . Let $\phi_f(t; t_0, x_0, \mathbf{w})$ denote this trajectory.

Suppose we are given an initial set $\mathcal{X}_0 \subseteq \mathbb{R}^{n_x}$ at initial time t_0 , and a compact disturbance set $\mathcal{W} \subset \mathbb{R}^{n_w}$. Then we can define the (forward) *reachable set* of the system as

$$\begin{aligned} \mathcal{R}_f(t; t_0, \mathcal{X}_0, \mathcal{W}) \\ := \{\phi_f(t; t_0, x_0, \mathbf{w}) : x_0 \in \mathcal{X}_0, \mathbf{w} : [t_0, t_f] \rightarrow \mathcal{W}\}. \end{aligned}$$

The reachable set is generally intractable to find in closed-form, both analytically and computationally, as it requires an infinite number of forward simulations of the system. The goal of this work and many others in the literature [20] is to efficiently compute a set *overapproximation* $\overline{\mathcal{R}}_f(t; t_0, \mathcal{X}_0, \mathcal{W}) \supseteq \mathcal{R}_f(t; t_0, \mathcal{X}_0, \mathcal{W})$.

An overapproximation of the true reachable set can verify several safety specifications for the system—for instance, reach-avoid problems can be posed as

$$\underbrace{\overline{\mathcal{R}}(t_f; t_0, \mathcal{X}_0, \mathcal{W}) \subseteq \mathcal{G}}_{\text{reach } \mathcal{G} \text{ at } t = t_f}, \underbrace{\overline{\mathcal{R}}(t; t_0, \mathcal{X}_0, \mathcal{W}) \subseteq \mathcal{A}^c}_{\text{avoid } \mathcal{A} \text{ for every } t} \forall t \in [t_0, t_f].$$

In the coming sections, we propose a new set representation called a normotope, build a controlled embedding system whose trajectory provides a normotope overapproximating the true reachable set, and optimize over the choice of hypercontrol mappings in this embedding space.

III. NORMOTOPES: A SIMPLE CONSTRAINT-BASED SET REPRESENTATION

In this section, we introduce the normotope, some basic properties, and a simple recipe to compute the normotope reachable set of a linear system using the adjoint dynamics.

A. Definition and Examples

Definition 1. Given a norm $\|\cdot\|$, a *normotope* is the set

$$\llbracket \hat{x}, \alpha, y \rrbracket := \{x \in \mathbb{R}^{n_x} : \|\alpha(x - \hat{x})\| \leq y\},$$

where $\hat{x} \in \mathbb{R}^{n_x}$ is the *center*, $\alpha \in GL(n_x)$ is the square invertible $n_x \times n_x$ *shape matrix*, and $y > 0$ is the *offset*.

Normotopes are a special case of a *parametope*, the general constraint-based set representation introduced in [12], and are dual to basic ellipsotopes [5] (letting $z = \frac{\alpha}{y}(x - \hat{x})$, $\llbracket \hat{x}, \alpha, y \rrbracket = \{y\alpha^{-1}z + \hat{x} : \|z\| \leq 1\}$, which is a basic ellipsotope with center $\hat{x}$ and generator matrix $y\alpha^{-1}$).

Example 1 (ℓ_2 normotopes). The normotope

$$\begin{aligned} \llbracket \hat{x}, \alpha, y \rrbracket_2 &= \{x \in \mathbb{R}^{n_x} : \|\alpha(x - \hat{x})\|_2 \leq y\} \\ &= \{x \in \mathbb{R}^{n_x} : (x - \hat{x})^T \alpha^T \alpha (x - \hat{x}) \leq y^2\} \end{aligned}$$

is equivalent to an ellipsoid centered at $\hat{x}$, with weighting matrix $\alpha^T \alpha$ and radius y^2 . $\alpha \in GL(n_x)$ implies $\alpha^T \alpha \succ 0$.

Example 2 (ℓ_∞ normotopes). The normotope

$$\begin{aligned} \llbracket \hat{x}, \alpha, y \rrbracket_\infty &= \{x \in \mathbb{R}^{n_x} : \|\alpha(x - \hat{x})\|_\infty \leq y\} \\ &= \{x \in \mathbb{R}^{n_x} : -y\mathbf{1} \leq \alpha(x - \hat{x}) \leq y\mathbf{1}\} \\ &= \{x \in \mathbb{R}^{n_x} : \alpha\hat{x} - y\mathbf{1} \leq \alpha x \leq \alpha\hat{x} + y\mathbf{1}\} \end{aligned}$$

is equivalent to the H-rep polytope given by $\{x : Hx \leq b\}$, $H = \begin{bmatrix} -\alpha \\ \alpha \end{bmatrix}$, $b = \begin{bmatrix} \alpha\hat{x} + y\mathbf{1} \\ -\alpha\hat{x} + y\mathbf{1} \end{bmatrix}$, with $2n_x$ faces and 2^{n_x} vertices.

Example 3 (ℓ_1 normotopes). The normotope

$$\begin{aligned} \llbracket \hat{x}, \alpha, y \rrbracket_1 &= \{x \in \mathbb{R}^{n_x} : \|\alpha(x - \hat{x})\|_1 \leq y\} \\ &= \{x \in \mathbb{R}^{n_x} : -y\mathbf{1} \leq S\alpha(x - \hat{x}) \leq y\mathbf{1}\} \\ &= \{x \in \mathbb{R}^{n_x} : S\alpha\hat{x} - y\mathbf{1} \leq S\alpha x \leq S\alpha\hat{x} + y\mathbf{1}\} \end{aligned}$$

is equivalent to the H-rep polytope given by $\{x : Hx \leq b\}$, $H = \begin{bmatrix} S\alpha \\ -S\alpha \end{bmatrix}$, $b = \begin{bmatrix} S\alpha\hat{x} + y\mathbf{1} \\ -S\alpha\hat{x} + y\mathbf{1} \end{bmatrix}$, with 2^{n_x} faces and $2n_x$ vertices, where $S \in \mathbb{R}^{2^{n_x} \times n_x}$ is the matrix consisting of all 2^{n_x} possible choices of signs, i.e., the k -th row $S_k = (s_1^k, \dots, s_n^k)$ where $s_j^k \in \{+1, -1\}$,

Remark 1 (Nomenclature). We choose the name “normotope” over “norm-ball” to emphasize how we expect the shape matrix α to vary along reachable set trajectories.

B. The LTV Case

In the undisturbed linear time-varying case (LTV) with a normotope initial set, the adjoint equation provides the true reachable set of the system as a normotope with fixed offset.

Proposition 1. For the LTV system

$$\dot{x}(t) = A(t)x(t),$$

the true reachable set is given by

$$\mathcal{R}_f(t; t_0, \llbracket \hat{x}_0, \alpha_0, y_0 \rrbracket) = \llbracket \hat{x}(t), \alpha(t), y_0 \rrbracket,$$

where $t \mapsto \hat{x}(t), \alpha(t)$ satisfy the following ODEs,

$$\dot{\hat{x}}(t) = A(t)\hat{x}(t), \quad \dot{\alpha}(t) = -\alpha(t)A(t).$$

with initial conditions $\dot{x}(t_0) = \dot{x}_0$, $\alpha(t_0) = \alpha_0$.

Proof. We first recall how the flow map is given by $\phi_f(t; t_0, x_0) = \Phi(t; t_0)x_0$, where Φ is the state transition matrix satisfying the matrix ODE

$$\dot{\Phi}(t; t_0) = A(t)\Phi(t; t_0), \quad \Phi(t_0; t_0) = \mathbf{I},$$

with the property $\Phi^{-1}(t; t_0) = \Phi(t_0; t)$. Thus, the true reachable set is given by

$$\begin{aligned} \mathcal{R}(t; t_0, [\dot{x}_0, \alpha_0, y_0]) &= \{\Phi(t; t_0)x : \|\alpha_0(x - \dot{x}_0)\| \leq y_0\} \\ &= \{z : \|\alpha_0(\Phi(t_0; t)z - \dot{x}_0)\| \leq y_0\} \\ &= \{z : \|\alpha_0\Phi(t_0; t)(z - \Phi(t; t_0)\dot{x}_0)\| \leq y_0\} \\ &= [\Phi(t; t_0)\dot{x}, \alpha_0\Phi(t_0; t), y_0] = [\hat{x}(t), \alpha(t), y_0], \end{aligned}$$

where $\alpha(t)$ satisfies the following,

$$\begin{aligned} \dot{\alpha}(t) &= \alpha_0 \frac{d}{dt} \Phi(t; t_0)^{-1} = -\alpha_0 \Phi(t; t_0)^{-1} \dot{\Phi}(t; t_0) \Phi(t; t_0)^{-1} \\ &= -\alpha_0 \Phi(t_0; t) A(t) \Phi(t; t_0) \Phi(t; t_0)^{-1} = -\alpha(t) A(t), \end{aligned}$$

and $\hat{x}(t)$ satisfies $\dot{\hat{x}}(t) = A(t)\hat{x}(t)$. $\square$

We can interpret the ODE $\dot{\alpha}(t) = -\alpha(t)A(t)$ as each row α_j^T evolving according to the adjoint dynamics $\dot{\alpha}_j(t) = -A(t)^T \alpha_j(t)$.

In the nonlinear systems case, the true reachable set is generally not itself a normotope. In the next section, we demonstrate how a similar embedding system approach can obtain guaranteed overapproximating normotope reachable sets by adding an appropriate dynamic compensation term to the offset y using logarithmic and induced matrix norms.

IV. NORMOTOPE EMBEDDINGS

In Proposition 1, we showed that the true reachable set in the linear time-varying case was given by a single forward simulation of another ODE (what we will refer to as an *embedding system*), by evolving the center $\hat{x}$ using the system dynamics and the rows of the shape matrix α using the adjoint dynamics. In this section, for nonlinear systems, we demonstrate how a similar *controlled embedding* approach can be used to bound the behavior of any nonlinear dynamics.

A. Dynamics Abstraction: Linear Differential Inclusions

A linear differential inclusion (LDI) [21, p.51] encompassing the error dynamics of the system (1) is an inclusion where for every $x \in \mathcal{X}$ and $w \in \mathcal{W}$, for prespecified $\hat{x} \in \mathcal{X} \subseteq \mathbb{R}^{n_x}$ and $\hat{w} \in \mathcal{W} \subseteq \mathbb{R}^{n_w}$,

$$f(t, x, w) - f(t, \hat{x}, \hat{w}) \in \mathcal{M}^x(t)(x - \hat{x}) + \mathcal{M}^w(t)(w - \hat{w}), \quad (2)$$

where $\mathcal{M}^x(t) \subseteq \mathbb{R}^{n_x \times n_x}$ and $\mathcal{M}^w(t) \subseteq \mathbb{R}^{n_x \times n_w}$ are sets of matrices. There are many ways to obtain matrix sets satisfying (2). For example, if $\overline{\text{co}}\{\frac{\partial f}{\partial x}(t, x, w)\} \subseteq \mathcal{M}^x(t)$ and $\overline{\text{co}}\{\frac{\partial f}{\partial w}(t, x, w)\} \subseteq \mathcal{M}^w(t)$ for every $(x, w) \in \mathcal{X} \times \mathcal{W}$ (where $\overline{\text{co}}$ is the closed convex hull), then (2) holds for every $(x, w) \in \mathcal{X} \times \mathcal{W}$ [17, Prop. 1], [21, p.55].

We briefly discuss an algorithmic method from [17, Cor. 1] to obtain the matrices $\mathcal{M}^x$ and $\mathcal{M}^w$ using interval analysis

on the first partial derivatives of f . Suppose we are bounding a C^1 function $g : \mathbb{R}^n \rightarrow \mathbb{R}^m$, and we are given an interval set $Z_1 \times \dots \times Z_n \subseteq \mathbb{R}^n$. Then the following implication holds, for fixed $\hat{z} \in Z_1 \times \dots \times Z_n$,

$$\begin{aligned} \frac{\partial g_i}{\partial x_j}(Z_1, \dots, Z_j, \hat{z}_{j+1}, \dots, \hat{z}_n) &\subseteq [\mathcal{M}]_{ij} \\ \implies g(z) - g(\hat{z}) &\in [\mathcal{M}](z - \hat{z}), \end{aligned}$$

for every $z \in Z_1 \times \dots \times Z_n$ [17, Cor. 1], where the RHS is evaluated using interval matrix multiplication [22]. We can apply this procedure to the map $f(t, \cdot, \cdot) : \mathbb{R}^{n_x} \times \mathbb{R}^{n_w} \rightarrow \mathbb{R}^{n_x}$ by defining an equivalent map $\hat{f}_t : \mathbb{R}^{n_x + n_w} \rightarrow \mathbb{R}^{n_x}$ to obtain the bound

$$f(t, x, w) - f(t, \hat{x}, \hat{w}) \in [\mathcal{M}(t)] \begin{bmatrix} x - \hat{x} \\ w - \hat{w} \end{bmatrix},$$

and finally, we can split $[\mathcal{M}(t)] \subseteq \mathbb{R}^{n_x \times (n_x + n_w)}$ into two interval matrices $[\mathcal{M}^x(t)] \subseteq \mathbb{R}^{n_x \times n_x}$ and $[\mathcal{M}^w(t)] \subseteq \mathbb{R}^{n_x \times n_w}$.

We simplify the expression by replacing the interval matrices $[\mathcal{M}^x(t)]$ and $[\mathcal{M}^w(t)]$ with the convex hull of a finite set of corners. For example, if the interval matrix $[\mathcal{M}^x(t)]$ has 4 non-singleton entries, then there are $2^4 = 16$ corners $M_i^x(t)$ to consider. Thus, we obtain a bound of the following form,

$$\begin{aligned} f(t, x, w) - f(t, \hat{x}, \hat{w}) &\in \\ &\text{co}\{M_i^x(t)\}_i(x - \hat{x}) + \text{co}\{M_j^w(t)\}_j(w - \hat{w}). \end{aligned}$$

As we will see in Theorem 1 below, these finite sets will help us build a simple embedding system by evaluating logarithmic and induced matrix norms over each corner.

B. Controlled Embedding System

In Theorem 1, we apply build a particularly well structured embedding system for the special case of a normotope set with a linear differential inclusion bounding the error dynamics to the center of the normotope. To do so, we introduce a *hypercontrol* map $U(t, \hat{x}, \alpha, y)$, which dynamically controls the evolution of the shape matrix of the normotope.

Theorem 1 (Controlled normotope embedding). *Let $\|\cdot\|$ be a differentiable norm, the ℓ_1 -norm, or the ℓ_∞ -norm. Let μ be the induced logarithmic norm. Define the following controlled embedding system,*

$$\dot{\hat{x}}(t) = f(t, \hat{x}, \hat{w}), \quad (3a)$$

$$\dot{\alpha}(t) = U(t, \hat{x}, \alpha, y), \quad (3b)$$

$$\begin{aligned} \dot{y}(t) &= \left(\max_i \mu(U(t, \hat{x}, \alpha, y)\alpha^{-1} + \alpha M_i^x(t)\alpha^{-1}) \right) y \\ &\quad + \max_j \|\alpha M_j^w(t)\|_{w \rightarrow x}, \end{aligned} \quad (3c)$$

where $f(t, x, w) - f(t, \hat{x}, \hat{w}) \in \text{co}\{M_i^x(t)\}(x - \hat{x}) + \text{co}\{M_j^w(t)\}(w - \hat{w})$ for every $(x, w) \in \partial[\hat{x}, \alpha, y] \times \mathcal{W}$. Then for every $t \geq 0$,

$$\mathcal{R}_f(t, [\dot{x}_0, \alpha_0, y_0], \mathcal{W}) \subseteq [\hat{x}(t), \alpha(t), y(t)],$$

where $t \mapsto (\hat{x}(t), \alpha(t), y(t))$ is the trajectory of the system (3) under any locally Lipschitz map U .

Proof. ((i) $\|\cdot\|$ is differentiable). For $(x, w) \in \partial[\hat{x}, \alpha, y] \times \mathcal{W}$, there is $M^x \in \text{co}\{M_i^x\}_i$ and $M^w \in \text{co}\{M_j^w\}_j$ such that

$$\begin{aligned} \xi(t, x, w) &:= \partial_\alpha g[U] + \partial_x g[M^x(x - \hat{x}) + M^w(w - \hat{w})] \\ &= \mathcal{L}_{\alpha(x - \hat{x})} \|\cdot\| [U(x - \hat{x}) + \alpha(M^x(x - \hat{x}) + M^w(w - \hat{w}))] \\ &= \lim_{h \rightarrow 0} \frac{\|\alpha(x - \hat{x}) + hU(x - \hat{x}) + h\alpha M^x(x - \hat{x}) + h\alpha M^w(w - \hat{w})\| - \|\alpha(x - \hat{x})\|}{h} \\ &\leq \limsup_{h \searrow 0} \frac{\|(I + h(U\alpha^{-1} + \alpha M^x \alpha^{-1}))\alpha(x - \hat{x})\| - \|\alpha(x - \hat{x})\|}{h} \\ &\quad + \|\alpha M^w(w - \hat{w})\| \\ &\leq \limsup_{h \searrow 0} \frac{\|I + h(U\alpha^{-1} + \alpha M^x \alpha^{-1})\|_{x \rightarrow x} - 1}{h} \|\alpha(x - \hat{x})\| \\ &\quad + \|\alpha M^w(w - \hat{w})\| \\ &= \mu(U\alpha^{-1} + \alpha M^x \alpha^{-1})y + \|\alpha M^w\|_{\text{op}} \\ &\leq \max_i \mu(U\alpha^{-1} + \alpha M_i^x \alpha^{-1})y + \max_j \|\alpha M_j^w\|_{w \rightarrow x}. \end{aligned}$$

where $\mathcal{L}_z \|\cdot\| [v]$ denotes the directional derivative of $\|\cdot\|$ at z in the direction v , and the final inequality holds since μ and $\|\cdot\|_{w \rightarrow x}$ are convex [23, Lemma 2.3] with arguments linear in M^x and M^w . [12, Cor. 1] concludes the proof.

((ii) $\|\cdot\| = \|\cdot\|_\infty$) We first note that the normotope $\{\|\alpha(x - \hat{x})\|_\infty \leq y\}$ is equivalent to the symmetric H-rep polytope $\{-\hat{y} \leq \alpha(x - \hat{x}) \leq \hat{y}\}$ where $\hat{y} = y\mathbf{1} \in \mathbb{R}^{n_x}$. Then for any x satisfying $-\hat{y} \leq \alpha(x - \hat{x}) \leq \hat{y}$ and $|\alpha_k(x - \hat{x})| = y$ for some k (where α_k is the k -th row of α),

$$\begin{aligned} \xi_k(t, x, w) &:= \dot{\alpha}_k(x - \hat{x}) + \alpha_k(f(t, x, w) - f(t, \hat{x}, \hat{w})) \\ &= \dot{\alpha}_k(x - \hat{x}) + \alpha_k(M^x(x - \hat{x}) + M^w(w - \hat{w})) \\ &= (\dot{\alpha}_k \alpha^{-1} + \alpha_k M^x \alpha^{-1})\alpha(x - \hat{x}) + \alpha_k M^w(w - \hat{w}) \\ &\leq ((\dot{\alpha}_k \alpha^{-1} + \alpha_k M^x \alpha^{-1})_k \\ &\quad + \sum_{j \neq k} |(\dot{\alpha}_k \alpha^{-1} + \alpha_k M^x \alpha^{-1})_j|)y + \alpha_k M^w(w - \hat{w}) \\ &\leq \mu_\infty(U\alpha^{-1} + \alpha M^x \alpha^{-1})y + \|\alpha M^w\|_{w \rightarrow x} \\ &\leq \max_i \mu_\infty(U\alpha^{-1} + \alpha M_i^x \alpha^{-1})y + \max_j \|\alpha M_j^w\|_{w \rightarrow x} \end{aligned}$$

by definition of the μ_∞ log norm [23, Table 2.1], and concluding with the same logic as part (i). The resulting H-polytope embedding recovers the same trajectory as (3) with the equivalence $\hat{y} = y\mathbf{1}$. [12, Cor. 1] concludes the proof.

((iii) $\|\cdot\| = \|\cdot\|_1$) We first note that

$$\|x\|_1 = \sum_i |x_i| = \max_{s_j \in S} (s_j^T x) = \|Sx\|_\infty,$$

where $S \in \mathbb{R}^{2^{n_x} \times n_x}$ is the matrix consisting of all 2^{n_x} possible choices of signs, i.e., the k -th row $S_k = (s_1^k, \dots, s_n^k)$ where $s_j^k \in \{+1, -1\}$. Thus, the normotope $\{\|\alpha(x - \hat{x})\|_1 \leq y\}$ is equivalent to the symmetric H-rep polytope given by $\{x : -\hat{y} \leq S\alpha(x - \hat{x}) \leq \hat{y}\}$, where $\hat{y} = y\mathbf{1}$. Let S^+ be a left inverse of S ($S^+ S = \mathbf{I}$). Then for any x satisfying $-\hat{y} \leq S\alpha(x - \hat{x}) \leq \hat{y}$ and $|S_k \alpha(x - \hat{x})| = y$ for some k (where α_k is the k -th row of α),

$$\begin{aligned} \xi_k(t, x, w) &= (S_k \dot{\alpha})(x - \hat{x}) \\ &\quad + S_k \alpha(M^x(x - \hat{x}) + M^w(w - \hat{w})) \\ &= (S_k \dot{\alpha} \alpha^{-1} S^+ + S_k \alpha M^x \alpha^{-1} S^+) S \alpha(x - \hat{x}) \\ &\quad + S^k \alpha M^w(w - \hat{w}) \\ &\leq \mu_\infty(S(U\alpha^{-1} + \alpha M^x \alpha^{-1}) S^+) y + \|S \alpha M^w\|_{\infty \rightarrow w}, \end{aligned}$$

concluding with the logic of part (ii). Finally, for any A ,

$$\begin{aligned} \|SAS^+\|_\infty &= \sup_{z: \|z\|_\infty=1} \|SAS^+ z\|_\infty = \sup_{x: \|Sx\|_\infty=1} \|SAx\|_\infty \\ &= \sup_{x: \|x\|_1} \|Ax\|_1 = \|A\|_1, \end{aligned}$$

thus, plugging into their definitions, we see that $\mu_\infty(SAS^+) = \mu_1(A)$ and $\|SA\|_{w \rightarrow \infty} = \|A\|_{w \rightarrow 1}$. [12, Cor. 1] concludes the proof. $\square$

In practice, we first overapproximate the normotope $[\hat{x}(t), \alpha(t), y(t)]$ with an interval subset, then apply the methodology from the previous section to obtain the matrices $\{M_i^x(t)\}_i$ and $\{M_j^w(t)\}_j$. Finally, we note that the dynamics (3c) are not the only possible choice of embedding dynamics; in fact, any expression satisfying the hypotheses of [12, Cor. 1] would constitute a valid embedding. However, when the dynamics are abstracted using a linear differential inclusion as (2), the expression (3c) is a natural choice with direct connections to contraction analysis [24], [19], [23].

Remark 2 (Connection to contraction theory). The $U = 0$ case of Theorem 1 recovers the fixed norm results from [13], [16], [17] for the norm with fixed shape matrix $\|x\|_\alpha = \|\alpha x\|$. The introduction of the hypercontrol U allows us to continuously vary the shape matrix along the trajectory of the normotope embedding system.

V. AN OPTIMAL CONTROL PERSPECTIVE TOWARDS REACHABLE SET COMPUTATION

In this section, we synthesize the policy U from (3) using an optimal control formulation. For clarity of presentation, we omit the disturbance input in this section and consider the following dynamics, for C^1 smooth $f: \mathbb{R} \times \mathbb{R}^{n_x} \rightarrow \mathbb{R}^{n_x}$,

$$\dot{x} = f(t, x),$$

but we note that all results in this section can easily be generalized to the case with a compact disturbance.

As seen in Proposition 1, for LTV systems the best choice of embedding dynamics is $\dot{\alpha} = -\alpha A(t)$. We choose hypercontrol policies of the form

$$U(t, \hat{x}, \alpha, y) = -\alpha \frac{\partial f}{\partial x}(\hat{x}) + U_{\text{ff}}(t),$$

where U_{ff} is a feed-forward input to the embedding system. As discussed in [25], closing the loop with the adjoint of the linearization cancels the first order expansion of the dynamics and generally provides reasonable results when the offset is small and the system is close to linear. In the rest of this section, using the controlled embedding system from the previous section, we formulate an optimal control problem and use iterative dynamic programming to optimize the accuracy of our reachable set overapproximations.

A. Cost Function Design

For simplicity, suppose we are given an initial set $[\hat{x}_0, \alpha_0, y_0]$ at $t = t_0$, and we are interested in verifying a goal specification at $t = t_f$. For a goal specification, we are purely interested in the size of the reachable set at the final

Algorithm 1 Reach-iLQR

Input: initial set $X_0 = \llbracket \hat{x}_0, \alpha_0, y_0 \rrbracket$, time interval $[t_0, t_f]$ **Parameters:** step size $\gamma > 0$, regularizer $R \succ 0$, volume threshold $J_{\max} \in \mathbb{R}$

- 1: $U_{\text{ff}}^{(0)}(t) \leftarrow 0$, $d(t) \leftarrow 0$, $K(t) \leftarrow 0$
 - 2: $i \leftarrow 1$
 - 3: **repeat**
 - 4: Forward simulate under the updated control $U_{\text{ff}}^{(i)}$ from (9) to obtain $X^{(i)}(t)$ until $J > J_{\max}$ or $t = t_f$.
 - 5: Compute terminal conditions (7) using automatic differentiation on terminal cost J from (4).
 - 6: Backward simulate Ricatti equations (6) using automatic differentiation to obtain $d(t)$, $K(t)$ from (8).
 - 7: $i \leftarrow i + 1$
 - 8: **until** termination criterion reached
-

time, and therefore, we design an optimal control problem with a pure terminal cost. The following Lemma recalls a standard result whereby maximizing the log-determinant of the shape matrix minimizes the volume of the normotope set.

Lemma 1. *For any norm $\|\cdot\|$, a minimizer of the cost*

$$J(\hat{x}, \alpha, y) = -\log \det(\alpha^T \alpha / y^2) \quad (4)$$

minimizes the volume of the normotope set $\llbracket \hat{x}, \alpha, y \rrbracket$.

Proof. The normotope set

$$\begin{aligned} \llbracket \hat{x}, \alpha, y \rrbracket &= \{x : \|\alpha(x - \hat{x})\| \leq y\} = \{x : \|\frac{\alpha}{y}(x - \hat{x})\| \leq 1\} \\ &= \{(\frac{\alpha}{y})^{-1}z + \hat{x} : \|z\| \leq 1\} \end{aligned}$$

is an affine image of the norm ball $\mathcal{B}_1(0)$. Thus, the volume

$$\text{vol}(\llbracket \hat{x}, \alpha, y \rrbracket) = |\det((\frac{\alpha}{y})^{-1})| \text{vol}(\mathcal{B}_1(0))$$

is inversely proportional to $|\det(\frac{\alpha}{y})|$. Since $\log((\cdot)^2)$ is monotone on $\mathbb{R}_{>0}$, we can maximize $\log(|\det(\frac{\alpha}{y})|^2) = \log(\det(\frac{\alpha^T}{y}) \det(\frac{\alpha}{y})) = \log \det(\frac{\alpha^T \alpha}{y^2})$. $\square$

We implement normotopes and the terminal cost using JAX for Python, taking care to respect the stringent requirements of JAX, which provides us several crucial capabilities. Normotopes are natively implemented as a Pytree node class, allowing us to use them as a valid JAX type for operations such as automatic differentiation. For the terminal cost (4), we use `jnp.linalg.cholesky` instead of composing `log` and `det` for numerical stability.

B. Open-Loop Hypercontrol Synthesis: Iterative Linear Quadratic Regulator

In this section, we numerically optimize the following

optimal control problem, given an initial set $\llbracket \hat{x}_0, \alpha_0, y_0 \rrbracket$:

$$\underset{U_{\text{ff}}: [t_0, t_f] \rightarrow \mathbb{R}^{n_x \times n_x}}{\text{minimize}} \quad J(\hat{x}(t_f), \alpha(t_f), y(t_f)) \quad (5)$$

$$\text{s.t.} \quad f(t, x) - f(t, \hat{x}) \in \text{co}\{M_i^x(t)\}(x - \hat{x}),$$

$$\dot{\hat{x}}(t) = f(t, \hat{x}),$$

$$\dot{\alpha}(t) = -\alpha \frac{\partial f}{\partial x}(t, \hat{x}) + U_{\text{ff}}(t)$$

$$\dot{y}(t) = \left(\max_i \mu(U_{\text{ff}}(t) \alpha^{-1} + \alpha(M_i^x(t) - \frac{\partial f}{\partial x}(t, \hat{x})) \alpha^{-1}) \right) y$$

$$(\hat{x}(t_0), \alpha(t_0), y(t_0)) = (\hat{x}_0, \alpha_0, y_0)$$

For the rest of this section, we write $X = \text{vec}(\hat{x}_0, \alpha_0, y_0) \in \mathbb{R}^{n_x} = \mathbb{R}^{n_x + n_x^2 + 1}$, and we abbreviate the embedding dynamics (3) as $\dot{X} = F(t, X, U)$.

Next, we describe Algorithm 1, which uses continuous-time iterative linear quadratic regulation (iLQR) [26] to improve the feedforward term U_{ff} . Suppose we are at timestep i , given $U_{\text{ff}}^{(i)}(t)$ and the corresponding embedding system trajectory $X^{(i)}(t)$ from forward simulating the embedding dynamics (3).

1) *Backward Pass:* Since we have no running cost, the backwards pass simplifies to the following ODEs,

$$-\dot{\sigma}(t) = -\frac{1}{2} Q_U^T(t) R^{-1} Q_U(t), \quad (6a)$$

$$-\dot{s}(t) = F_X^T(t) s(t) - Q_{UX}^T(t) R^{-1} Q_U(t), \quad (6b)$$

$$-\dot{S}(t) = Q_{XX}(t) - Q_{UX}^T(t) R^{-1} Q_{UX}(t), \quad (6c)$$

where $R \succ 0$ is some regularization matrix,

$$Q_U(t) = F_U^T(t) s(t),$$

$$Q_{XX}(t) = F_X^T(t) S(t) + S(t) F_X(t),$$

$$Q_{UX}(t) = F_U^T(t) S(t),$$

and the following

$$F_U^T(t) = \frac{\partial F}{\partial U}(t, X^{(i)}(t), U_{\text{ff}}^{(i)}(t)),$$

$$F_X^T(t) = \frac{\partial F}{\partial X}(t, X^{(i)}(t), U_{\text{ff}}^{(i)}(t)),$$

are computed using automatic differentiation in JAX. The ODEs are initialized by automatically differentiating the terminal cost J with respect to the state X as

$$\sigma(t_f) = J(X(t_f)), \quad (7a)$$

$$s(t_f) = \frac{\partial J}{\partial X}(X(t_f))^T, \quad (7b)$$

$$S(t_f) = \frac{\partial^2 J}{\partial X^2}(X(t_f)). \quad (7c)$$

Finally, we obtain curves $d(t)$ and $K(t)$ as

$$d(t) = R^{-1} Q_U(t), \quad K(t) = R^{-1} Q_{UX}(t). \quad (8)$$

The key result from iLQR is that the variational system (first order expansion of the dynamics)

$$\delta \dot{X} = F_X^T(t) \delta X(t) + F_U^T(t) \delta U(t)$$

and the second order expansion of the value function

$$V(x, t) \approx \sigma(t) + s(t)^T \delta X(t) + \frac{1}{2} \delta X(t)^T S(t) \delta X(t)$$

satisfy the Hamilton-Jacobi-Bellman PDE with feedback controller $\delta U(t) = -d(t) - K(t)\delta X(t)$ (for the full derivation, please see [26]).

2) *Forward Pass*: From the backward pass, we obtain the curves $d(t)$ and $K(t)$. We then perform a forward sweep by updating U_{ff} as follows (where i has been incremented),

$$U_{\text{ff}}^{(i)}(t) = U_{\text{ff}}^{(i-1)}(t) - \gamma d(t) - K(t)(X^{(i)}(t) - X^{(i-1)}(t)), \quad (9)$$

where $\gamma > 0$ is the step size, $X^{(i-1)}$ is the previous embedding trajectory under $U_{\text{ff}}^{(i-1)}$, and $X^{(i)}$ is the new embedding trajectory under $U_{\text{ff}}^{(i)}$. Theoretically, when γ is small enough, the displacement $\delta X = X^{(i)} - X^{(i-1)}$ along the trajectory is small enough where the first order dynamics expansion and second order value expansion remain accurate.

While the forward pass always produces a valid reachable set overapproximation, it is often possible that the reachable set blows up before the prescribed final time t_f either due to overconservativeness in the interval analysis generating the linear differential inclusion, and a poor choice of shaping matrix trajectory $\alpha(t)$. To ensure the terminal conditions (7) are well defined, we introduce a threshold J_{max} , where we stop the forward simulation if $J(X(t)) > J_{\text{max}}$. When this happens, we proceed as if the final time t_f was the time instance before the threshold was violated; in practice, we find that iterates will eventually tend towards policies which reach the final time. The result is an algorithm where each iteration provides a valid overapproximate reachable tube, tending towards those with smaller terminal set volumes.

Remark 3 (Comparison to the literature). Previous approaches [13], [16], [17] break the time interval $[t_0, t_f]$ into segments, solve intermediate semidefinite programs to optimize over the potential choices of norms along each individual segment, and ensure the reachable set containment by overapproximating the norm ball at the end of the previous segment with the new norm ball at the start of the next segment. Algorithm 1 instead provides a continuous formulation where the hypercontrol smoothly varies the shaping matrix along the nominal trajectory, both avoiding successive norm overapproximations and allowing us to backpropagate the terminal set volume all the way to the initial timestep.

VI. EXAMPLES

In the following examples, we demonstrate how Algorithm 1 can (i) be used to quickly modify the reachable set to obtain drastically improve estimates in a short amount of time, and (ii) learn nontrivial open-loop policies for more difficult benchmarks. For all ODE simulations, we use Euler integration with a step size of 0.01.¹

¹The code is available at https://github.com/gtfactslab/Harapanahalli_ACC2026.

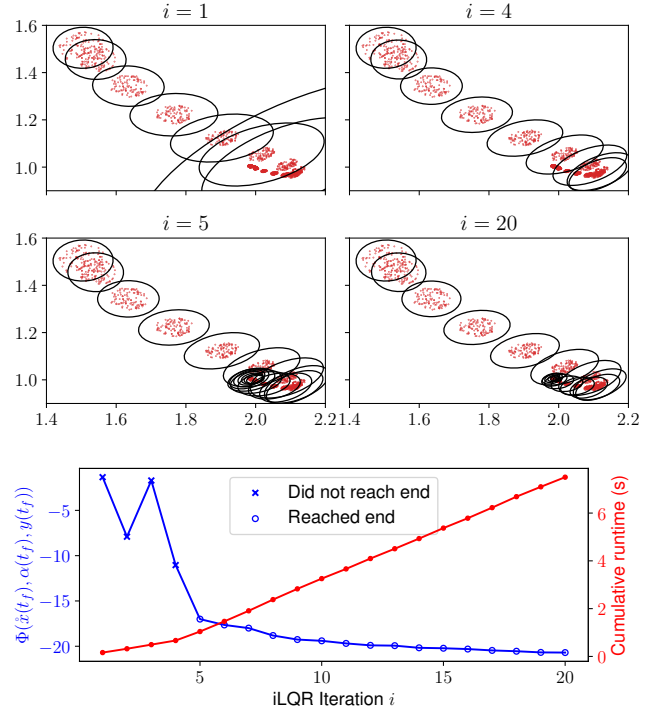


Fig. 1. **Top**: Overapproximating reachable sets for the robot arm for $t \in [0, 10]$ are plotted for various iteration numbers of Algorithm 1, with Monte Carlo samples in red. **Bottom**: The terminal cost J is plotted as a function of the iLQR iteration number in blue, while the cumulative runtime is in red. After 5 iterations of iLQR, the overapproximate reachable set reaches the final timestep. Iterating further, we can trade runtime for less overconservatism in the final reachable set.

A. Robot Arm

Consider the dynamics of a robot arm from [27], [15]

$$\begin{aligned} \dot{q}_1 &= z_1, \quad \dot{q}_2 = z_2, \\ \dot{z}_1 &= \frac{1}{mq_2^2 + ML^2/3}(-2mq_2z_1z_2 - k_{d1}z_1 + k_{p1}(u_1 - q_1)), \\ \dot{z}_2 &= q_2z_1^2 + \frac{1}{m}(-k_{d2}z_2 + k_{p2}(u_2 - q_2)), \end{aligned}$$

with $u_1 = 2$, $u_2 = 1$, $m = M = 1$, $L = \sqrt{3}$, $k_{p1} = 2$, $k_{p2} = 1$, $k_{d1} = 2$, $k_{d2} = 1$. We use the initial set $\llbracket \dot{x}_0, \alpha_0, y_0 \rrbracket_2$, where $\alpha_0 = P^{1/2}/0.1$ with P and $\dot{x}_0$ from [15], and $y_0 = 1$. We note that this initial set has a radius 2.5 times larger than the largest radius considered in the comparison example from [17], and thus a volume that is 39 times larger. We run Algorithm 1 with $t_f = 10$, $\gamma = 1$, $R = 20$, and $J_{\text{max}} = -0.1$ for 20 iterations, taking a total of 7.5 seconds, and the results are shown in Figure 1.

B. Van der Pol Oscillator

Consider the Van der Pol oscillator written in usual state space coordinates as

$$\dot{x}_1 = x_2, \quad \dot{x}_2 = -x_1 + \eta(1 - x_1^2)x_2,$$

with nonlinearity $\eta = 1$. We consider the initial set $\llbracket \dot{x}_0, \alpha_0, y_0 \rrbracket_2$ for $\dot{x}_0 = [-2 \ 0]^T$, $\alpha_0 = 80I$, and $y_0 = 1$.

We split the iLQR iterations into two phases. We first start with a small regularization of $R = 0.5I$ as an exploratory

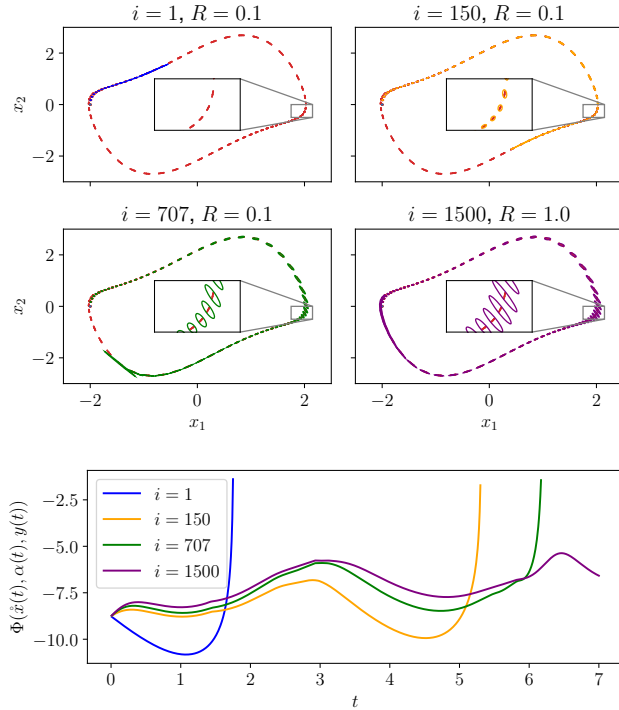


Fig. 2. **Top:** Overapproximating reachable sets for the Van der Pol oscillator for $t \in [0, 7]$ are shown for various iteration numbers of Algorithm 1, with Monte Carlo samples in red. **Bottom:** The function $J(X(t))$ is plotted as a function of time for the same iteration numbers. While the pure adjoint dynamics ($i = 1$) does well for small horizons, the error quickly accumulates over longer horizons and cannot reach the goal of $t_f = 7$ s. On the other hand, after 3000 iterations of iLQR, the algorithm qualitatively learns to accumulate extra error in earlier timesteps so it can reach and minimize the final timestep.

phase, keeping track of the best reachable set computation in all the iterates (the one that reaches the furthest before violation $J > J_{\max} = -1.75$, with the smallest volume). After 750 iterations, we revert to the best iterate from the first phase ($i = 707$) and use a larger regularizer $R = 5I$ for another 750 iterations to help reach the end. All 1500 iterations took a total of 23.8 seconds to execute.

VII. CONCLUSION

In this work, we presented an optimal control framework for synthesizing guaranteed overapproximate reachable sets with small terminal volume using iterative dynamic programming. Our framework builds off the controlled parametric embedding approach described in [12], constructing a special case for *normotope* sets using logarithmic and induced matrix norms. In future work, we plan to use techniques from reinforcement learning to learn closed-loop policies using offline training to mimic the output of Algorithm 1, which would be instant for online reachable set computations.

REFERENCES

- [1] M. Althoff, “An Introduction to CORA 2015,” in *Proc. of the 1st and 2nd Workshop on Applied Verification for Continuous and Hybrid Systems*, pp. 120–151, EasyChair, Dec. 2015.
- [2] J. K. Scott, D. M. Raimondo, G. R. Marzaglia, and R. D. Braatz, “Constrained zonotopes: A new tool for set-based estimation and fault detection,” *Automatica*, vol. 69, pp. 126–136, 2016.

- [3] N. Kochdumper and M. Althoff, “Sparse polynomial zonotopes: A novel set representation for reachability analysis,” *IEEE Transactions on Automatic Control*, vol. 66, no. 9, pp. 4043–4058, 2020.
- [4] T. J. Bird, H. C. Pangborn, N. Jain, and J. P. Koeln, “Hybrid zonotopes: A new set representation for reachability analysis of mixed logical dynamical systems,” *Automatica*, vol. 154, p. 111107, 2023.
- [5] S. Kousik, A. Dai, and G. X. Gao, “Ellipsotopes: Uniting ellipsoids and zonotopes for reachability analysis and fault detection,” *IEEE Transactions on Automatic Control*, vol. 68, no. 6, pp. 3440–3452, 2022.
- [6] X. Chen, *Reachability analysis of non-linear hybrid systems using Taylor models*. PhD Thesis, Fachgruppe Informatik, RWTH Aachen University, 2015.
- [7] P. S. Duggirala and M. Viswanathan, “Parsimonious, simulation based verification of linear systems,” in *International conference on computer aided verification*, pp. 477–494, Springer, 2016.
- [8] I. Mitchell and C. J. Tomlin, “Level set methods for computation in hybrid systems,” in *Hybrid Systems: Computation and Control (HSCC)*, pp. 310–323, 2000.
- [9] S. Bansal, M. Chen, S. Herbert, and C. J. Tomlin, “Hamilton-Jacobi reachability: A brief overview and recent advances,” in *56th Conference on Decision and Control (CDC)*, pp. 2242–2253, 2017.
- [10] S. Bansal and C. J. Tomlin, “Deepreach: A deep learning approach to high-dimensional reachability,” in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1817–1824, IEEE, 2021.
- [11] P. Varaiya, “Reach set computation using optimal control,” in *Verification of Digital and Hybrid Systems*, pp. 323–331, Springer, 2000.
- [12] A. Harapanahalli and S. Coogan, “Parametric reachable sets via controlled dynamical embeddings,” *arXiv preprint arXiv:2504.06955*, 2025.
- [13] J. Maidens and M. Arcak, “Reachability analysis of nonlinear systems using matrix measures,” *IEEE Transactions on Automatic Control*, vol. 60, no. 1, pp. 265–270, 2014.
- [14] M. Arcak and J. Maidens, “Simulation-based reachability analysis for nonlinear systems using componentwise contraction properties,” *Principles of Modeling: Essays Dedicated to Edward A. Lee on the Occasion of His 60th Birthday*, pp. 61–76, 2018.
- [15] C. Fan, J. Kapinski, X. Jin, and S. Mitra, “Locally optimal reach set over-approximation for nonlinear systems,” in *Proceedings of the 13th International Conference on Embedded Software*, pp. 1–10, 2016.
- [16] C. Fan, J. Kapinski, X. Jin, and S. Mitra, “Simulation-driven reachability using matrix measures,” *ACM Transactions on Embedded Computing Systems (TECS)*, vol. 17, no. 1, pp. 1–28, 2017.
- [17] A. Harapanahalli and S. Coogan, “A linear differential inclusion for contraction analysis to known trajectories,” *arXiv preprint arXiv:2411.11587*, 2024.
- [18] C. Desoer and M. Vidyasagar, *Feedback systems: Input-output properties*. Society for Industrial and Applied Mathematics, 1975.
- [19] A. Davydov, S. Jafarpour, and F. Bullo, “Non-Euclidean contraction theory for robust nonlinear stability,” *IEEE Transactions on Automatic Control*, vol. 67, no. 12, pp. 6667–6681, 2022.
- [20] M. Althoff, G. Frehse, and A. Girard, “Set propagation techniques for reachability analysis,” *Annual Review of Control, Robotics, and Autonomous Systems*, vol. 4, no. 1, pp. 369–395, 2021.
- [21] S. Boyd, L. E. Ghaoui, E. Feron, and V. Balakrishnan, *Linear matrix inequalities in system and control theory*. SIAM, 1994.
- [22] L. Jaulin, M. Kieffer, O. Didrit, and E. Walter, *Applied interval analysis*. Springer, 1st edition, ed., 2001.
- [23] F. Bullo, *Contraction theory for dynamical systems*. Kindle Direct Publishing, 1.1 ed., 2023.
- [24] E. D. Sontag, “Contractive systems with inputs,” in *Perspectives in Mathematical System Theory, Control, and Signal Processing: A Festschrift in Honor of Yutaka Yamamoto on the Occasion of his 60th Birthday*, pp. 217–228, Springer, 2010.
- [25] A. Harapanahalli, S. Jafarpour, and S. Coogan, “immrax: A parallelizable and differentiable toolbox for interval analysis and mixed monotone reachability in JAX,” *IFAC-PapersOnLine*, vol. 58, no. 11, pp. 75–80, 2024.
- [26] J. Lieskovský, J. Bušek, and T. Vyhlídal, “Continuous-time iterative linear-quadratic regulator,” *arXiv preprint arXiv:2505.15525*, 2025.
- [27] D. Angeli, E. Sontag, and Y. Wang, “A characterization of integral input-to-state stability,” *IEEE Transactions on Automatic Control*, vol. 45, no. 6, pp. 1082–1097, 2000.