Computing Robust Control Invariant Sets of Nonlinear Systems Using Polynomial Controller Synthesis

Lukas Schäfer and Matthias Althoff

Abstract— **Deploying nonlinear sampled-data systems in safety-critical applications requires us to ensure robust constraint satisfaction for an infinite time horizon. To maximize the region of safe operation, we aim to compute a robust control invariant set with maximum volume. In this work, we propose an iterative optimization-based algorithm that computes a sequence of candidate invariant sets, which is volume-wise monotonically increasing. By leveraging polynomialization-based techniques from reachability analysis and controller synthesis, our approach outperforms linearization-based approaches, especially for higher-dimensional systems. We show that the computational complexity of each iteration of our algorithm is polynomial in the state dimension and demonstrate its broad applicability using several examples from the literature with up to 10 dimensions.**

I. Introduction

Deploying controlled systems, such as vehicles, robots, and drones, in safety-critical applications is very challenging, since these systems exhibit nonlinear dynamics and hard constraints on the state must be satisfied with limited control effort despite disturbances. We address this issue by computing robust control invariant (RCI) sets satisfying provided constraints. RCI sets have many applications in robust control: they serve as terminal sets in robust model predictive control [1], [2] or safeguard learning-based controllers [3], [4]. The following literature overview categorizes the approaches based on the chosen set representation (please see [5] for a more exhaustive literature overview).

Polytopic Sets: The computation of polytopic (robust control) invariant sets of nonlinear systems has been considered in [6]–[10]. The works in [6]–[8] abstract the discrete-time nonlinear system by a convex difference inclusion [11], [12]. However, these approaches rely on the vertex representation of polytopes, which enables a flexible design of the (robust control) invariant set, but restricts the applicability to low-dimensional systems. In the case of polynomial dynamics, the computation of polytopic (robust control) invariant sets can be encoded as a polynomial program that can be relaxed into a sequence of linear programs [9], [13]. However, these approaches have only been applied to low-dimensional systems and it is not clear how the relaxation can be automated.

Level Sets: Level sets have been used to represent invariant sets computed by methods such as Hamilton-Jacobi reachability analysis [14]–[16], occupation measures [17]–[19], and control-barrier functions [20]–[22]. For all these approaches, the task of computing an invariant set is encoded as a (sequence of) semidefinite program(s). However, a) the invariance-enforcing controller must be designed prior to computing the set [15]–[17], b) only controlled but not perturbed systems are considered [18], or c) the size of the semidefinite program compromises the scalability of the approach [18], [23].

Ellipsoidal Sets: Usually, algorithms that use ellipsoids to represent the RCI set scale better to higher-dimensional systems but yield more conservative results: In [24], [25], the nonlinear system is abstracted by a polytopic linear difference inclusion and, in [26], a control invariant set is computed for the linearized system before verifying invariance for the nonlinear system by solving a non-convex program.

Grid Cells: The control invariant set is represented as the union of grid cells in [27], [28]. While these approaches enable to approximate the maximal control invariant set arbitrarily close in theory, gridding the state space restricts their applicability to low-dimensional systems.

Previous Work: In our previous work [5], we proposed the first algorithm for computing RCI sets of nonlinear systems using zonotopes as a set representation. By combining scalable reachability analysis with (successive) convexification, we computed RCI sets of nonlinear systems with up to 20 dimensions in only a couple of minutes. However, we kept the center of the RCI set fixed and abstracted the nonlinear dynamics to a linear difference inclusion.

Contribution: In this paper, we address the aforementioned sources of conservatism in [5] to obtain larger RCI sets of perturbed nonlinear sampled-data systems. In particular, we

- use zonotopes as an efficient set representation of the RCI set and consider its center as an optimization variable;
- leverage the polynomialization approach for reachability analysis from [29] and polynomial controller synthesis [30] to obtain less conservative results;
- design a sequence of polynomial programs where the resulting sequence of candidate RCI sets is volume-wise monotonically increasing; and
- demonstrate the broad applicability and scalability of our approach using various examples from the literature

The authors are with the TUM School of Computation, Information, and Technology, Technical University of Munich, Boltzmannstr. 3, 85748 Garching b. München, Germany. Email: {lukas.schaefer, althoff}@tum.de

with up to 10 dimensions.

Organization: In Sec. II, we provide the problem statement, introduce the required set representations, present conditions for zonotope containment, and algorithms for reachability analysis of nonlinear systems. We introduce and analyze our novel approach for computing RCI sets in Sec. III. The discussion and evaluation of our algorithm are provided in Sec. IV and Sec. V, respectively.

Notation: The sets of natural numbers with and without zero are denoted by $\mathbb{N}_0$ and $\mathbb{N}$, respectively. For two vectors $y, v \in \mathbb{R}^o$, we define $y^v = \prod_{i=1}^{o} y_i^{v_i}$. The vector full of ones and zeros of appropriate dimension is denoted by $\mathbf{1}$ and $\mathbf{0}$, respectively. Given a matrix $A \in \mathbb{R}^{m \times n}$, we use $A_{(j)}$ to denote the j-th column of A and $A_{(\mathcal{J})}$, where $\mathcal{J} = \{j_1, \dots, j_m\} \subset \mathbb{N}$, is used for $[A_{(j_1)}, \dots, A_{(j_m)}]$. The absolute value $|A|$ as well as equalities and inequalities between vectors and matrices are applied elementwise. If A is a square matrix, $\det(A)$ refers to its determinant. For $a \in \mathbb{R}^n$, the operator $\operatorname{diag}(a)$ returns a diagonal matrix with the elements of a on the main diagonal. Given two sets $\mathcal{A}, \mathcal{B} \subset \mathbb{R}^n$, $\mathcal{A} \oplus \mathcal{B} = \{a + b : a \in \mathcal{A}, b \in \mathcal{B}\}$ denotes their Minkowski addition.

II. Preliminaries

A. Problem Statement

We consider perturbed continuous-time nonlinear systems of the form

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

where $x(t) \in \mathbb{R}^{n_x}$ is the system state, $u(t) \in \mathbb{R}^{n_u}$ is the control input, and $w(t) \in \mathbb{R}^{n_w}$ is the unknown disturbance at time $t \in \mathbb{R}_{\geq 0}$. The nonlinear function f is assumed to be sufficiently smooth and the input as well as disturbance trajectories $u(\cdot)$ and $w(\cdot)$, respectively, are assumed to be piecewise continuous. Moreover, $w(\cdot)$ is confined to the set of disturbances $\mathcal{W} \subset \mathbb{R}^{n_w}$, i.e., $\forall t$: $w(t) \in \mathcal{W}$, which we denote by $w(\cdot) \in \mathcal{W}$. We assume that $\mathcal{W}$ is compact and contains the origin. The solution of (1) at time $t \in \mathbb{R}_{\geq 0}$ with initial state $x(0) = x_0$, input $u(\cdot)$, and disturbance $w(\cdot)$ is denoted by $\chi(t, x_0, u(\cdot), w(\cdot))$.

In controller synthesis for cyber-physical systems, one usually encounters the setting of sampled-data systems, i.e., a physical plant evolving in continuous time is controlled by a digital controller [3]. Measurements are obtained at discrete points in time $t_k = k\Delta t$ with $\Delta t \in \mathbb{R}_{>0}$, $k \in \mathbb{N}_0$, and the actuators provide a piecewise constant control input

$$u(t) = u_{\text{ctrl}}(x(t_k)), \quad \forall t \in [t_k, t_{k+1}[, \tag{2}$$

where u_{ctrl} denotes a given sampled-data control law. Next, we define the closed-loop reachable set.

Definition 1 (One-step Reachable Set): For the system in (1), a set of initial states $\mathcal{X}_0 \subset \mathbb{R}^{n_x}$, a sampled-data controller u_{ctrl}, and a set of disturbances $\mathcal{W}$, the reachable set $\mathcal{R}(\Delta t, \mathcal{X}_0, u_{\text{ctrl}})$ after one time step is the set of states reachable from $\mathcal{X}_0$ at time Δt:

$$\begin{aligned}\mathcal{R}(\Delta t, \mathcal{X}_0, u_{\text{ctrl}}) = \{\chi(\Delta t, x_0, u_{\text{ctrl}}(x_0), w(\cdot)) :\\ x_0 \in \mathcal{X}_0, w(\cdot) \in \mathcal{W}\}.\end{aligned} \tag{3}$$

The reachable set over the time interval $[0, \Delta t]$ is the union of reachable sets $\mathcal{R}(t, \mathcal{X}_0, u_{\text{ctrl}})$, $\forall t \in [0, \Delta t]$:

$$\mathcal{R}([0, \Delta t], \mathcal{X}_0, u_{\text{ctrl}}) = \bigcup_{t \in [0, \Delta t]} \mathcal{R}(t, \mathcal{X}_0, u_{\text{ctrl}}). \tag{4}$$

Since the exact computation of $\mathcal{R}(t, \mathcal{X}_0, u_{\text{ctrl}})$ for general nonlinear systems is impossible [31], we compute over-approximations, i.e., $\widehat{\mathcal{R}}(t, \mathcal{X}_0, u_{\text{ctrl}}) \supseteq \mathcal{R}(t, \mathcal{X}_0, u_{\text{ctrl}})$ to ensure safety [32].

The state and the control input are constrained by

$$x(\cdot) \in \mathcal{X}, \tag{5a}$$

$$u(\cdot) \in \mathcal{U}, \tag{5b}$$

where $\mathcal{X} = \{x \in \mathbb{R}^{n_x} : H_{\mathcal{X}} x \leq h_{\mathcal{X}}\}$ is a polytope and $\mathcal{U}$ is assumed to be representable as a zonotope (Sec. II-B). As in our previous work [5], we aim to compute an RCI set $\hat{\mathcal{S}}_{\text{RCI}} \subset \mathbb{R}^{n_x}$ with maximum volume and its associated safety-preserving controller $\hat{u}_{\text{RCI}}$ around a steady state $(x_{\text{eq}}, u_{\text{eq}})$ for nonlinear sampled-data systems by solving

$$\begin{aligned}&\boldsymbol{P}_{\text{RCI}}:\\ &\left(\hat{\mathcal{S}}_{\text{RCI}}, \hat{u}_{\text{RCI}}\right) = \underset{\mathcal{S}, u_{\mathcal{S}}}{\arg\max} \quad \text{VOLUME}(\mathcal{S}) &\text{(6a)}\\ &\qquad\text{such that}\\ &\qquad\widehat{\mathcal{R}}(\Delta t, \mathcal{S}, u_{\mathcal{S}}) \subseteq \mathcal{S}, &\text{(6b)}\\ &\qquad\widehat{\mathcal{R}}([0, \Delta t], \mathcal{S}, u_{\mathcal{S}}) \subseteq \mathcal{X}, &\text{(6c)}\\ &\qquad\forall x_0 \in \mathcal{S} : u_{\mathcal{S}}(x_0) \in \mathcal{U}. &\text{(6d)}\end{aligned}$$

The condition in (6b) ensures invariance of $\hat{\mathcal{S}}_{\text{RCI}}$ at the sampling instants; moreover, satisfaction of the state and input constraints in between sampling times is enforced by the constraints in (6c) and (6d), respectively. Since this argument can be repeated for any future point in time, robust satisfaction of (5a) and (5b) for every $x_0 \in \hat{\mathcal{S}}_{\text{RCI}}$ follows.

B. Set Representations

Zonotopes are a popular set representation for reachability analysis and controller synthesis [33].

Definition 2 (Zonotope): A zonotope $\mathcal{Z} \subset \mathbb{R}^{n_z}$ is given by

$$\mathcal{Z} = \{z \in \mathbb{R}^{n_z} : z = c + G\lambda, |\lambda| \leq \mathbf{1}\}$$

where $c \in \mathbb{R}^{n_z}$ is the center and $G \in \mathbb{R}^{n_z \times \eta_{\mathcal{Z}}}$ is the generator matrix with $\eta_{\mathcal{Z}}$ denoting the number of generators of $\mathcal{Z}$. We use the shorthand $\mathcal{Z} = \langle c, G \rangle$.

In addition, we introduce polynomial zonotopes [29] [34, Ch. 3.2], which are beneficial for reachability analysis of nonlinear systems. Unless stated otherwise, we use polynomial zonotopes to compute reachable sets in this work.

Definition 3 (Polynomial Zonotope): A polynomial zonotope $\mathcal{PZ} \subset \mathbb{R}^{n_z}$ is given by

$$\mathcal{PZ} = \Bigg\{ z \in \mathbb{R}^{n_z} : z = c + \sum_{i=1}^{\eta_\mathrm{D}} g_{(i)} \lambda^{e_{(i)}} + G_\mathrm{I}\delta, \\ |\lambda| \leq \mathbf{1}, |\delta| \leq \mathbf{1} \Bigg\}$$

where $c \in \mathbb{R}^{n_z}$ is the center, $G = \left[g_{(1)} \cdots g_{(\eta_\mathrm{D})}\right] \in \mathbb{R}^{n_z \times \eta_\mathrm{D}}$ is the matrix of dependent generators, $G_\mathrm{I} \in \mathbb{R}^{n_z \times \eta_\mathrm{I}}$ is the matrix of independent generators, and $E = \left[e_{(1)} \ldots e_{(\eta_\mathrm{D})}\right] \in \mathbb{R}^{o \times \eta_\mathrm{D}}$ is the exponent matrix. The operator $\mathrm{Z}\left(\mathcal{PZ}\right)$ returns a zonotope that encloses a polynomial zonotope $\mathcal{PZ}$ [34, Proposition 3.1.14].

C. Zonotope Containment

To formulate the set containment conditions in (6b)-(6d), we recall two encodings of the zonotope containment problem. Consider the zonotopes $\mathcal{Z}_1 = \langle c_1, G_1 \rangle \subset \mathbb{R}^{n_z}$ and $\mathcal{Z}_2 = \langle c_2, G_2 \mathrm{diag}\left(\alpha\right) \rangle \subset \mathbb{R}^{n_z}$ where $\alpha \in \mathbb{R}_{>0}^{\eta_{\mathcal{Z}_2}}$. $\mathcal{Z}_1$ is contained in $\mathcal{Z}_2$, i.e., $\mathcal{Z}_1 \subseteq \mathcal{Z}_2$ if there exist $\Gamma \in \mathbb{R}^{\eta_{\mathcal{Z}_2} \times \eta_{\mathcal{Z}_1}}$, $\gamma \in \mathbb{R}^{\eta_{\mathcal{Z}_2}}$ such that [35, Lemma 2]

$$G_1 = G_2 \Gamma, \tag{7a}$$

$$c_2 - c_1 = G_2 \gamma, \tag{7b}$$

$$\left|\left[\Gamma \ \gamma\right]\right| \mathbf{1} \leq \alpha. \tag{7c}$$

Given a polytope $\mathcal{P} = \{z \in \mathbb{R}^{n_z} : Hz \leq h\}$,

$$Hc_1 + |HG_1| \, \mathbf{1} \leq h, \tag{8}$$

is necessary and sufficient for $\mathcal{Z}_1 \subseteq \mathcal{P}$ [36, Theorem 2].

D. Reachability Analysis of Nonlinear Systems

In this section, we provide a brief overview of the reachability algorithm in [29]. First, the nonlinear dynamics in (1) are abstracted by a Taylor series of order κ:

$$\dot{x}(t) \ \in \ f_\mathrm{lin}(x(t), u(t), w(t)) + \underbrace{\xi(x(t), u(t), w(t)) \oplus \mathcal{L}(t)}_{=:\,\Xi(t)}, \tag{9}$$

where $f_\mathrm{lin}(x(t), u(t), w(t))$ is the first-order approximation of (1), $\xi(x(t), u(t), w(t))$ contains all higher-order terms up to order κ, and $\mathcal{L}(t)$ denotes the Lagrange remainder. We expand the system dynamics about $x = x_\mathrm{eq}$, $u = u_\mathrm{eq}$, and $w = \mathbf{0}$ since we compute an RCI set around the steady state $(x_\mathrm{eq}, u_\mathrm{eq})$. Using the uncertainty set $\Xi(t)$, which encloses $\xi(x(t), u(t), w(t))$ and $\mathcal{L}(t)$, we can apply the superposition principle of linear systems to compute reachable sets: First, we compute the reachable sets of the linearized dynamics $f_\mathrm{lin}(x(t), u(t), w(t))$ at the time point Δt and for the time-interval $[0, \Delta t]$. To obtain $\widehat{\mathcal{R}}\left(\Delta t, \mathcal{X}_0, u_\mathrm{ctrl}\right)$ and $\widehat{\mathcal{R}}\left([0, \Delta t], \mathcal{X}_0, u_\mathrm{ctrl}\right)$, we add the reachable set due to the set of uncertainties $\Xi([0, \Delta t])$, which encloses $\Xi(t)$ for $t \in [0, \Delta t]$.

For the presentation of our approach, we use a symmetric Lagrange remainder

$$\mathcal{L}\left([0, \Delta t]\right) = \left\langle \mathbf{0}, \mathrm{diag}\left(\ell\left(\widehat{\mathcal{R}}\left([0, \Delta t], \mathcal{X}_0, u_\mathrm{ctrl}\right)\right)\right)\right\rangle,$$

where $\ell\left(\widehat{\mathcal{R}}\left([0, \Delta t], \mathcal{X}_0, u_\mathrm{ctrl}\right)\right) \in \mathbb{R}_{\geq 0}^{n_x}$ bounds the Lagrange remainder, see e.g. [34, Eq. (4.13)] for $\kappa = 2$. Moreover, we assume that the discretization of $f_\mathrm{lin}(x(t), u(t), w(t))$ with sampling time Δt is stabilizable.

III. Computation of RCI Sets

In general, we obtain a set of non-differentiable constraint functions if we execute the reachability algorithm in Sec. II-D to evaluate the conditions in (6b) and (6c). Inspired by the successive convexification algorithm in our previous work [5], we therefore propose to iteratively approximate $\boldsymbol{P}_\mathrm{RCI}$.

We start by introducing the parameterized candidate RCI set $\mathcal{S}$ and the candidate invariance-enforcing controller $u_\mathcal{S}$. Afterwards, we present the steps conducted in the i-th iteration of our algorithm: Based on the reachability algorithm in Sec. II-D, we first compute parameterized approximations $\widetilde{\mathcal{R}}^{(i)}\left(\Delta t, \mathcal{S}, u_\mathcal{S}\right)$ and $\widetilde{\mathcal{R}}^{(i)}\left([0, \Delta t], \mathcal{S}, u_\mathcal{S}\right)$ of the reachable sets that enable a differentiable encoding of the conditions in (6b) and (6c) (Sec. III-B). Afterwards, we solve a polynomial approximation $\widetilde{\boldsymbol{P}}^{(i)}$ of $\boldsymbol{P}_\mathrm{RCI}$ to obtain the updated candidate RCI set $\mathcal{S}^{(i)}$ and candidate invariance-enforcing controller $u_\mathcal{S}^{(i)}$ (Sec. III-C). In Sec. III-D, we show that repeating these two steps yields a sequence of candidate RCI sets $\mathcal{S}^{(\cdot)}$ that is volume-wise monotonically increasing.

Since we use approximations of the reachable sets for optimization, we cannot guarantee that any of the iterates $\mathcal{S}^{(i)}$, $u_\mathcal{S}^{(i)}$ satisfies the conditions in (6b)-(6d). In the last step of our algorithm, we therefore compute over-approximations of the reachable sets $\widehat{\mathcal{R}}\left(\Delta t, \mathcal{S}^*, u_\mathcal{S}^*\right)$ and $\widehat{\mathcal{R}}\left([0, \Delta t], \mathcal{S}^*, u_\mathcal{S}^*\right)$ to verify safety of the converged solution $\mathcal{S}^*$, $u_\mathcal{S}^*$ (see [5, Sec. III] for more details). We use $\mathcal{S}_\mathrm{RCI}$ and u_RCI to denote the verified solution of our algorithm.

A. RCI Set and Controller Parameterization

We adopt the generator scaling framework introduced in [37] to obtain an efficient parameterization of the reachable sets: Given a non-degenerate initial guess $\mathcal{S}^{(0)} = \left\langle x_\mathrm{eq}, G_\mathcal{S}^{(0)} \right\rangle$ for $\hat{\mathcal{S}}_\mathrm{RCI}$, we keep the orientation of its η_RCI generators fixed and introduce a vector of scaling factors $s \in \mathbb{R}_{\geq 0}^{\eta_\mathrm{RCI}}$ as an optimization variable. Using the initial step $G_\mathcal{S}^{(1)} = G_\mathcal{S}^{(0)} \mathrm{diag}\left(s^{(1)}\right)$, the generator matrix of $\mathcal{S}^{(i)}$ is defined recursively as

$$G_\mathcal{S}^{(i)} = G_\mathcal{S}^{(i-1)} \mathrm{diag}\left(s^{(i)}\right). \tag{10}$$

To increase flexibility, we introduce the translation $\Delta c_\mathcal{S} \in \mathbb{R}^{n_x}$ of the center of $\mathcal{S}$ as an optimization variable. The center of $\mathcal{S}^{(i)}$ is obtained using the recursion

$$c_\mathcal{S}^{(i)} = c_\mathcal{S}^{(i-1)} + \Delta c_\mathcal{S}^{(i)} \tag{11}$$

with the initial step $c_\mathcal{S}^{(1)} = x_\mathrm{eq} + \Delta c_\mathcal{S}^{(1)}$. By combining (10) and (11), the parameterized candidate RCI set in $\widetilde{\boldsymbol{P}}^{(i)}$ follows as

$$\mathcal{S}\left(\Delta c_\mathcal{S}, s\right) = \left\langle c_\mathcal{S}^{(i-1)} + \Delta c_\mathcal{S}, G_\mathcal{S}^{(i-1)} \mathrm{diag}\left(s\right) \right\rangle = \left\langle c_\mathcal{S}, G_\mathcal{S} \right\rangle. \tag{12}$$

As we will see in the next subsection, the evolution of any $x_0 \in \mathcal{S}\left(\Delta c_{\mathcal{S}}, s\right)$ can be expressed as a polynomial combination of the center and generators of $\mathcal{S}\left(\Delta c_{\mathcal{S}}, s\right)$. Thus, we can compute a control input for x_0 by finding control inputs for the center and generators of $\mathcal{S}\left(\Delta c_{\mathcal{S}}, s\right)$ and interpolating between them using $\lambda_{\mathcal{S}}$, where $x_0 = c_{\mathcal{S}} + G_{\mathcal{S}}\lambda_{\mathcal{S}}$ with $|\lambda_{\mathcal{S}}| \leq \mathbf{1}$. Since $\mathcal{U}$ is assumed to be a zonotope, i.e.,

$$\mathcal{U} = \left\{u \in \mathbb{R}^{n_u} : u = c_{\mathcal{U}} + G_{\mathcal{U}}\lambda_{\mathcal{U}}, |\lambda_{\mathcal{U}}| \leq \mathbf{1}\right\}, \tag{13}$$

the computation of the control inputs for the center and generators of $\mathcal{S}\left(\Delta c_{\mathcal{S}}, s\right)$ can be equivalently expressed as the computation of the corresponding factor $\lambda_{\mathcal{U}}$. We use the parameterization of $\lambda_{\mathcal{U}}$ proposed in [30]: Given a user-defined matrix of exponents $O = \left[o_{(1)} \dots o_{(M)}\right] \in \mathbb{N}_0^{\eta_{\mathrm{RCI}} \times M}$ and the matrix of controller parameters $P = \left[p_{(1)} \dots p_{(M)}\right] \in \mathbb{R}^{n_u \times M}$, we define

$$\lambda_{\mathcal{U}}(\lambda_{\mathcal{S}}, P) = p_{(1)} + \sum_{k=2}^{M} p_{(k)} \lambda_{\mathcal{S}}^{o_{(k)}}, \tag{14}$$

i.e., $\lambda_{\mathcal{U}}(\lambda_{\mathcal{S}}, P)$ is a polynomial in $\lambda_{\mathcal{S}}$ and the optimization variables P. Please note that we define $o_{(1)} = \mathbf{0}$ to model offsets via $p_{(1)}$. Since we impose no other restrictions regarding the choice of the exponent matrix O, the user can add arbitrary desired monomials to $\lambda_{\mathcal{U}}(\lambda_{\mathcal{S}}, P)$. Combining (13) with (14), the parameterized candidate invariance-enforcing control input $u_{\mathcal{S}}(P)$ given the state $x_0 \in \mathcal{S}\left(\Delta c_{\mathcal{S}}, s\right)$ is

$$u_{\mathcal{S}}(P) = c_{\mathcal{U}} + G_{\mathcal{U}}\lambda_{\mathcal{U}}(\lambda_{\mathcal{S}}, P). \tag{15}$$

For the remainder of this work, we write $\mathcal{S}$ and $u_{\mathcal{S}}$ instead of $\mathcal{S}\left(\Delta c_{\mathcal{S}}, s\right)$ and $u_{\mathcal{S}}(P)$, respectively, where convenient.

B. Parameterized Reachability Analysis

We now define the parameterized approximations $\widetilde{\mathcal{R}}^{(i)}\left(\Delta t, \mathcal{S}, u_{\mathcal{S}}\right)$ and $\widetilde{\mathcal{R}}^{(i)}\left([0, \Delta t], \mathcal{S}, u_{\mathcal{S}}\right)$. To enable a differentiable encoding of the conditions in (6b) and (6c), we evaluate the uncertainty set $\Xi(t)$ in (9) only for the initial state x_0 since $\Xi([0,0]) \approx \Xi([0,\Delta t])$ for small time steps as explained in [29]. This approximation entails $\mathcal{L}\left([0,0]\right) = \left\langle \mathbf{0}, \operatorname{diag}\left(\ell\left(\mathcal{S}\right)\right)\right\rangle \approx \mathcal{L}\left([0,\Delta t]\right)$. To avoid the costly evaluation of $\ell\left(\mathcal{S}\left(\Delta c_{\mathcal{S}}, s\right)\right)$ and its derivatives while solving $\widetilde{\boldsymbol{P}}^{(i)}$, we approximate the Lagrange remainder by

$$\Psi^{(i)}\left(\mathcal{S}\right) = \left\langle \mathbf{0}, \operatorname{diag}\left(\psi\right)\right\rangle, \tag{16}$$

where $\psi \in \mathbb{R}_{\geq 0}^{n_x}$ and $\psi_j \geq \tilde{\psi}_j^{(i)}\left(\mathcal{S}\right), j \in \{1, \dots, n_x\}$. The interchangeable function $\tilde{\psi}_j^{(i)}\left(\mathcal{S}\right)$ is a polynomial, i.e., it is twice continuously differentiable, and we assume that $\tilde{\psi}_j^{(i)}\left(\mathcal{S}\right) \approx \ell_j\left(\widehat{\mathcal{R}}\left([0,\Delta t], \mathcal{S}, u_{\mathcal{S}}\right)\right)$. We provide a simple example for $\tilde{\psi}_j^{(i)}\left(\mathcal{S}\right)$ in Sec. III-D.

By plugging $\Xi([0,0]) \approx \Xi([0,\Delta t])$ and (16) into the reachability algorithm in Sec. II-D, we obtain the polynomial zonotope

$$\begin{aligned}\widetilde{\mathcal{R}}^{(i)}\left(\Delta t, \mathcal{S}, u_{\mathcal{S}}\right) = \Bigg\{ & x \in \mathbb{R}^{n_x} : x = c_{\mathcal{R}}\left(\Delta c_{\mathcal{S}}, P\right) \\ & + \sum_{l=1}^{\eta_{\mathrm{D}}} g_{\mathcal{R},(l)}\left(\Delta c_{\mathcal{S}}, P, \psi\right) s^{e_{(l)}} \lambda_{\mathcal{S}}^{e_{(l)}} \\ & + G_{\mathrm{I}}\delta, \ \ |\lambda_{\mathcal{S}}| \leq \mathbf{1}, |\delta| \leq \mathbf{1} \Bigg\},\end{aligned} \tag{17}$$

where $c_{\mathcal{R}}\left(\Delta c_{\mathcal{S}}, P\right)$ and $g_{\mathcal{R},(l)}\left(\Delta c_{\mathcal{S}}, P, \psi\right)$ are vector-valued polynomials. For simplicity, we use the heuristic

$$\widetilde{\mathcal{R}}^{(i)}\left([0,\Delta t], \mathcal{S}, u_{\mathcal{S}}\right) = \mathcal{S} \approx \widehat{\mathcal{R}}\left([0,\Delta t], \mathcal{S}, u_{\mathcal{S}}\right) \tag{18}$$

to approximate the time-interval reachable set.

C. Polynomial Optimization Problem

As stated in Sec. II-A, our goal is to compute an RCI set with maximum volume. According to [38, Corollary 3.4], the cost function $\textsc{volume}\left(\mathcal{S}\right)$ thus is

$$\textsc{volume}\left(\mathcal{S}\right) = 2^{n_x} \sum_{j=1}^{n_{\mathrm{comb}}} \left|\det\left(G_{(\mathcal{J}(j))}^{(i-1)}\right)\right| \prod_{l \in \mathcal{J}(j)} s_l,$$

where $\mathcal{J}(j)$ denotes one of the n_{comb} possible n_x-membered subsets of $\{1, \dots, \eta_{\mathrm{RCI}}\}$.

Due to the parameterization of the invariance-enforcing controller in Sec. III-A, we use the condition $|\lambda_{\mathcal{U}}(\lambda_{\mathcal{S}}, P)| \leq \mathbf{1}$ from (13) to encode the input constraint in (6d). To check this condition for every $x_0 \in \mathcal{S}$, we introduce the set $\Lambda_{\mathcal{U}}(P)$:

$$\Lambda_{\mathcal{U}}(P) = \left\{\lambda_{\mathcal{U}}(\lambda_{\mathcal{S}}, P) : |\lambda_{\mathcal{S}}| \leq \mathbf{1}\right\},$$

which is representable as a polynomial zonotope [30].

Using the proposed approximations in (17) and (18) of the exact reachable sets in (3) and (4), respectively, we can now state the polynomial program $\widetilde{\boldsymbol{P}}^{(i)}$:

$\widetilde{\boldsymbol{P}}^{(i)}$:

$$\left(\mathcal{S}^{(i)}, u_{\mathcal{S}}^{(i)}\right) = \underset{\Delta c_{\mathcal{S}}, s, P, \psi}{\arg\max} \quad \textsc{volume}\left(\mathcal{S}\left(\Delta c_{\mathcal{S}}, s\right)\right) \tag{19a}$$

such that

$$\mathrm{Z}\left(\widetilde{\mathcal{R}}^{(i)}\left(\Delta t, \mathcal{S}\left(\Delta c_{\mathcal{S}}, s\right), u_{\mathcal{S}}(P)\right)\right) \subseteq \mathcal{S}\left(\Delta c_{\mathcal{S}}, s\right), \tag{19b}$$

$$\mathcal{S}\left(\Delta c_{\mathcal{S}}, s\right) \subseteq \mathcal{X}, \tag{19c}$$

$$\mathrm{Z}\left(\Lambda_{\mathcal{U}}(P)\right) \subseteq [-\mathbf{1}, \mathbf{1}], \tag{19d}$$

$$\forall j \in \{1, \dots, n_x\} : \psi_j \geq \tilde{\psi}_j^{(i)}\left(\mathcal{S}\right), \tag{19e}$$

where we compute the zonotope enclosure of $\widetilde{\mathcal{R}}^{(i)}\left(\Delta t, \mathcal{S}, u_{\mathcal{S}}\right)$ and $\Lambda_{\mathcal{U}}(P)$ in (19b) and (19d), respectively, to use the conditions for zonotope containment in (7) and (8). Since $\mathcal{S}$ is representable as a zonotope and both $\widetilde{\mathcal{R}}^{(i)}\left(\Delta t, \mathcal{S}, u_{\mathcal{S}}\right)$ and $\Lambda_{\mathcal{U}}(P)$ are representable as polynomial zonotopes, the cost function and all constraint functions in $\widetilde{\boldsymbol{P}}^{(i)}$ can be rewritten as polynomials.

D. Iterative Procedure

In Sec. III-B, we proposed to approximate the Lagrange remainder using a set of polynomial functions. While this enables us to solve $\widetilde{\boldsymbol{P}}^{(i)}$ efficiently in practice, finding functions $\tilde{\psi}_j^{(i)}(\mathcal{S})$ that tightly approximate the Lagrange remainder $\hat{\mathcal{L}}([0,\Delta t])$ can be challenging, especially for high-dimensional systems. On the other hand, using rather simple functions $\tilde{\psi}_j^{(i)}(\mathcal{S})$ can lead to conservative solutions. Therefore, we propose an iterative procedure, which realizes a local refinement of $\tilde{\psi}_j^{(i)}(\mathcal{S})$. To ensure that every additional iteration improves the quality of our solution, i.e., the resulting sequence of candidate RCI sets $\mathcal{S}^{(\cdot)}$ is volume-wise monotonically increasing, we require that $\tilde{\psi}_j^{(i)}(\mathcal{S})$ meets the following requirements, which are adapted from [5, Def. 4]:

$$\forall i \in \mathbb{N}: \ \tilde{\psi}_j^{(i+1)}\left(\mathcal{S}^{(i)}\right) \leq \tilde{\psi}_j^{(i)}\left(\mathcal{S}^{(i)}\right) \tag{20a}$$

$$\forall \mathcal{S} \subseteq \mathcal{X}: \ \ell_j(\mathcal{S}) \leq \tilde{\psi}_j^{(i)}(\mathcal{S}). \tag{20b}$$

As we will show subsequently in Theorem 1, the condition in (20a) ensures that the solution of $\widetilde{\boldsymbol{P}}^{(i)}$ is a feasible solution of $\widetilde{\boldsymbol{P}}^{(i+1)}$. Moreover, we impose the condition in (20b) to ensure that $\tilde{\psi}_j^{(i)}(\mathcal{S})$ approximates the Lagrange remainder bound $\ell_j\left(\widehat{\mathcal{R}}([0,\Delta t],\mathcal{S},u_{\mathcal{S}})\right)$ reasonably well. Let us now provide a simple example for a suitable $\tilde{\psi}_j^{(i)}(\mathcal{S})$.

Example 1: For simplicity, we assume a one-dimensional dynamical system. Furthermore, we introduce the operator $\text{BOX}(\mathcal{A},a)$, which returns the smallest interval that is centered at a and contains the compact set $\mathcal{A}$. Using interval arithmetic to bound the Lagrange remainder, see e.g. [34, (4.13)] for $\kappa = 2$, $\ell_j(\mathcal{S})$ is a homogeneous polynomial of degree $\kappa+1$ in the edge-length of $\text{BOX}(\mathcal{S},x_{\text{eq}})$. Hence, choosing $\tilde{\psi}_1^{(i)}(\mathcal{S})$ as a piece-wise linear function defined by $\ell_1(\{x_{\text{eq}}\})$, $\ell_1\left(\text{BOX}\left(\mathcal{S}^{(i)},x_{\text{eq}}\right)\right)$, and $\ell_1(\text{BOX}(\mathcal{X},x_{\text{eq}}))$ satisfies the conditions in (20). Since the constraint in (19e) can be encoded as a set of two linear inequalities, differentiability is ensured. Please note that this simple example can be generalized to higher-dimensional systems by using the auxiliary variable $\check{\beta}$ with $\check{\beta} = \min\left\{\beta : \beta\,\text{BOX}\left(\mathcal{S}^{(i)},x_{\text{eq}}\right) \supseteq \mathcal{S}\right\}$ for interpolation.

To show that $\mathcal{S}^{(\cdot)}$ is volume-wise monotonically increasing, we require the following auxiliary result.

Lemma 1: Assume that the constraints in (19e) hold with equality. If all $\tilde{\psi}_j^{(i)}(\mathcal{S}), i \in \mathbb{N}$, are chosen according to (20a), then $\widetilde{\mathcal{R}}^{(i+1)}\left(\Delta t,\mathcal{S}^{(i)},u_{\mathcal{S}}^{(i)}\right) \subseteq \widetilde{\mathcal{R}}^{(i)}\left(\Delta t,\mathcal{S}^{(i)},u_{\mathcal{S}}^{(i)}\right)$.

A proof is provided in Appendix I.

Theorem 1: Let $\widetilde{\boldsymbol{P}}^{(m)}, m \in \mathbb{N}$, admit a feasible solution $\mathcal{S}^{(m)}$ and $u_{\mathcal{S}}^{(m)}$. For every $i > m, i \in \mathbb{N}$, it holds that $\text{VOLUME}\left(\mathcal{S}^{(i-1)}\right) \leq \text{VOLUME}\left(\mathcal{S}^{(i)}\right)$, i.e., the sequence $\mathcal{S}^{(\cdot)}$ is volume-wise monotonically increasing.

Proof: We show that $\mathcal{S}^{(m)}$,$u_{\mathcal{S}}^{(m)}$ is a feasible solution of $\widetilde{\boldsymbol{P}}^{(m+1)}$, which implies that $\text{VOLUME}\left(\mathcal{S}^{(m)}\right)$ is a lower bound of $\text{VOLUME}\left(\mathcal{S}^{(m+1)}\right)$. Since this argument can be applied for any $i > m$, the claim follows by induction.

Satisfaction of the constraint in (19c) follows for $\Delta c_{\mathcal{S}} = \mathbf{0}$ and $s = \mathbf{1}$ since $\mathcal{S}(\mathbf{0},\mathbf{1}) = \mathcal{S}^{(m)}$. Similarly, choosing $P = P^{(m)}$ satisfies the constraint in (19d). Since ψ_j is only lower-bounded by $\tilde{\psi}_j^{(m+1)}\left(\mathcal{S}^{(m)}\right)$, setting $\psi_j = \tilde{\psi}_j^{(m+1)}\left(\mathcal{S}^{(m)}\right)$ in (19e) is feasible. From Lemma 1, we therefore obtain $\widetilde{\mathcal{R}}^{(m+1)}\left(\Delta t,\mathcal{S}^{(m)},u_{\mathcal{S}}^{(m)}\right) \subseteq \widetilde{\mathcal{R}}^{(m)}\left(\Delta t,\mathcal{S}^{(m)},u_{\mathcal{S}}^{(m)}\right)$. Thus, the constraint in (19b) is satisfied, which concludes the proof. ■

So far, we have assumed that the initial guess $\mathcal{S}^{(0)}, u_{\mathcal{S}}^{(0)}$ admits a feasible solution of $\widetilde{\boldsymbol{P}}^{(1)}$. As proposed in [5, Remark 3], we can rewrite the conditions in (7c) and (8) as soft constraints so that our approach can handle an infeasible initial guess. Extending the results in Theorem 1 is straightforward.

IV. Discussion of the Algorithm

A. Computational Complexity

Every iteration of our algorithm consists of two main steps: computing parameterized approximations of the reachable sets and solving the polynomial program $\widetilde{\boldsymbol{P}}^{(i)}$. The computation of the parameterized reachable sets is based on the reachability algorithm presented in Sec. II-D, which has complexity $\mathcal{O}\left((\max(n_x,n_u))^5\right)$ [34, Sec. 4.1.4]. If we use second-order methods to solve $\widetilde{\boldsymbol{P}}^{(i)}$, the required number of function evaluations depends polynomially on the requested solution accuracy $\epsilon_{\text{opt}} > 0$ [39]. Since evaluating the polynomial constraint and cost functions (thus, their Jacobians and Hessians are polynomials too) has polynomial complexity in n_x and n_u [40], solving $\widetilde{\boldsymbol{P}}^{(i)}$ to a stationary point of accuracy ϵ_{opt} has polynomial complexity in n_x and n_u. Hence, every iteration of our algorithm has polynomial complexity in n_x and n_u. We cannot provide an upper bound on the required number of iterations, however, our algorithm usually terminates in a small number of iterations, see Sec. V.

B. Existence of Solutions and Formal Guarantees

Since $\widetilde{\boldsymbol{P}}^{(i)}$ approximates $\boldsymbol{P}_{\text{RCI}}$, feasibility of $\boldsymbol{P}_{\text{RCI}}$ does not imply feasibility of $\widetilde{\boldsymbol{P}}^{(1)}$. Similarly, a feasible solution of $\widetilde{\boldsymbol{P}}^{(i)}$ not necessarily satisfies the constraints in (6b) and (6c). However, if $\widetilde{\boldsymbol{P}}^{(m)}, m \in \mathbb{N}$, admits a feasible solution, it is guaranteed that every $\widetilde{\boldsymbol{P}}^{(i)}, i > m$, admits a feasible solution (see the proof of Theorem 1). Moreover, we verify satisfaction of the constraints in (6b) and (6c) for the converged solution by computing over-approximations of the reachable sets (Sec. III).

V. Numerical Experiments

We apply our robust control approach to several examples from the literature. In Sec. V-A, we compare our novel polynomialization (poly.) approach to our previous linearization (lin.) approach [5] and demonstrate the effect of choosing the center of $\mathcal{S}_{\text{RCI}}$ as an optimization variable. Afterwards, we analyze the scalability of our approach in Sec. V-B and close this section with a comparison with an approach from the literature in Sec. V-C.

For all examples, we use the convergence criterion proposed in [5]

$$\frac{\text{VOLUME}\left(\mathcal{S}^{(i)}\right)-\text{VOLUME}\left(\mathcal{S}^{(i-1)}\right)}{\text{VOLUME}\left(\mathcal{S}^{(i-1)}\right)}\leq\epsilon, \tag{21}$$

with $\epsilon=10^{-3}$ as the convergence tolerance, the Taylor order for the abstraction of the nonlinear dynamics is chosen as $\kappa=2$ (Sec. II-D) and $\eta_{\text{RCI}}=5n_x$. To enable a fair comparison, we choose the matrix of exponents O for the controller (Sec. III-A) so that we obtain a linear interpolation-based controller as in [5]. As an initial guess, we compute an RCI set for the linearized dynamics using the approach from [35].

Our implementation and the benchmark systems alongside all parameters will be made publicly available with the next release of the AROC[1] toolbox [41]. We use the open-source tool CORA [42] for reachability analysis and IPOPT [43] via the MATLAB interface included in the OPTI toolbox[2] for solving the polynomial programs $\widetilde{\boldsymbol{P}}^{(i)}$. All computations were conducted on a laptop equipped with an Intel Core i7-11370H and 64 GB of memory.

A. Comparison with the Linearization Approach

For the comparison with our previous work [5], we use a broad range of control systems such as a cartpole or a quadrotor. Table I provides a summary of the results. The fifth column indicates whether the initial guess admits a feasible solution of $\widetilde{\boldsymbol{P}}^{(1)}$. The sixth and seventh column provide the number of iterations of our algorithm until the convergence criterion in (21) is satisfied (# iter. opt.) and until the converged solution can be verified as safe (# iter. ver.), respectively. The average time required for solving $\widetilde{\boldsymbol{P}}^{(i)}$ is reported in the eighth column. The numbers in parentheses denote the corresponding number of convex programming iterations in [5] and the corresponding average computation times per convex program. The last column shows the quotient

$$J_n\left(\mathcal{S}_{\text{RCI}}\right)=\frac{\text{VOLUME}\left(\mathcal{S}_{\text{RCI}}\text{ poly. approach}\right)}{\text{VOLUME}\left(\mathcal{S}_{\text{RCI}}\text{ lin. approach}\right)}.$$

Both approaches manage to compute a feasible solution for all the examples in Table I. By formulating the containment checks in (7) and (8) as soft constraints (Sec. III-D), our approach can recover from an infeasible initial guess, see the Cartpole and Robot Arm example. Most importantly, the polynomialization approach outperforms the linearization approach in every example due to the less conservative approximations of the reachable sets, which is indicated by $J_n\left(\mathcal{S}_{\text{RCI}}\right)>1$ in the last column. However, the improved performance comes at the cost of increased computation times. We consider a chain of nonlinear mass-spring-damper systems to obtain a better impression of the scalability and the performance with increasing dimension of the state space in Sec. V-B.

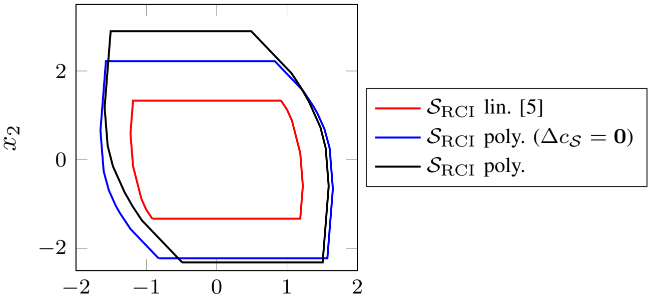

Fig. 1: Cart example: When considering the center of $\mathcal{S}_{\text{RCI}}$ as an optimization variable, $\mathcal{S}_{\text{RCI}}$ is translated in the positive x_2-direction. In both cases, the polynomialization approach yields a significantly larger RCI set compared to the linearization approach [5].

Next, we consider the cart example from [41] to demonstrate the efficacy of choosing the center as an optimization variable. The dynamics of the cart are governed by

$$\begin{aligned}\dot{x}_1&=x_2+w_1,\\ \dot{x}_2&=-x_2^2-x_1^3+u+w_2,\end{aligned}\tag{22}$$

where x_1,x_2 denote the position and the velocity of the cart, respectively. Measurements are taken with a sampling time of $\Delta t=0.1\,\text{s}$ and the origin is chosen as the equilibrium for the computation of $\mathcal{S}_{\text{RCI}}$. The results for a fixed center of $\mathcal{S}_{\text{RCI}}$, i.e., $\Delta c_{\mathcal{S}}=\mathbf{0}$, and a free center of $\mathcal{S}_{\text{RCI}}$ are shown in Fig. 1. In the latter case, the center of $\mathcal{S}_{\text{RCI}}$ is translated in the positive x_2-direction to exploit the nonlinear damping term in (22). Thereby, the volume of $\mathcal{S}_{\text{RCI}}$ can be increased by 7%. Compared to the linearization approach, the volume of $\mathcal{S}_{\text{RCI}}$ increases by more than 100% when using the novel polynomialization approach.

B. Analysis of the Scalability

To analyze the scalability of our approach, we use the chain of n_{mass} coupled nonlinear mass-spring-damper systems from [5]. The results for $n_{\text{mass}}\in\{1,\dots,5\}$ are summarized in Table II. As already indicated by the analysis in Sec. IV-A, the numbers in both Table I and Table II indicate that the computational complexity of our approach scales reasonably well with the dimension of the state space n_x and the input space n_u. In comparison with the convex approximations of $\boldsymbol{P}_{\text{RCI}}$ in [5], the computational effort for solving a polynomial approximation $\widetilde{\boldsymbol{P}}^{(i)}$ of $\boldsymbol{P}_{\text{RCI}}$ grows faster with the dimension of the state space (see the third column of Table II). However the last column of Table II indicates that using higher-order abstractions of the nonlinear dynamics for reachability analysis is performance-wise even more beneficial with increasing n_x, since we handle the nonlinearities in a less conservative way.

[1] https://tumcps.github.io/AROC/
[2] https://github.com/jonathancurrie/OPTI

TABLE I: Comparison with our previous work [5].

Example	n_x	n_u	n_w	$\widetilde{\boldsymbol{P}}^{(1)}$ feas.?	Poly. Approach (Lin. Approach [5])			$J_n\left(\mathcal{S}_{\text{RCI}}\right)$
					# iter. opt.	# iter. ver.	∅ solver time	
Jet Engine [9] (Sec. V-C)	2	1	1	✓	2 (8)	1 (1)	0.49 s (0.1 s)	1.07
Cart [41] (Sec. V-A)	2	1	2	✓	2 (10)	1 (0)	0.67 s (0.11 s)	2.30
Mass-Spring-Damper System [44]	2	1	2	✓	8 (8)	0 (0)	0.36 s (0.12 s)	1.06
Cartpole (dynamics of the pendulum) [45]	2	1	1	✓	10 (18)	1 (0)	0.27 s (0.13 s)	1.65
Cartpole [46]	4	1	1	✗	9 (16)	5 (0)	1.78 s (0.66 s)	7.33
Pendubot [47]	4	1	1	✓	14 (19)	0 (0)	2.01 s (0.86 s)	125.9
Robot Arm [41]	4	2	4	✗	16 (28)	2 (0)	1.83 s (0.55 s)	2.4
Longitudinal Quadrotor [3]	6	2	2	✓	7 (10)	6 (1)	32.0 s (0.79 s)	6.5
Chain of 5 Mass-Spring-Damper Systems Sec. V-B	10	5	5	✓	3 (25)	1 (0)	128.5 s (1.9 s)	1009.1

TABLE II: Analysis of the scalability (using n_{mass} coupled nonlinear mass-spring-damper systems, see also [5]).

n_{mass}	Poly. Approach (Lin. Approach [5])		$J_n\left(\mathcal{S}_{\text{RCI}}\right)$
	# iter. opt. + ver.	∅ solver time	
1 ($n_x = 2$)	3 + 1 (14 + 0)	0.42 s (0.11 s)	1.7
2 ($n_x = 4$)	3 + 1 (18 + 0)	4.9 s (0.49 s)	26.4
3 ($n_x = 6$)	3 + 1 (22 + 0)	19.1 s (0.75 s)	156.4
4 ($n_x = 8$)	3 + 0 (26 + 0)	76.2 s (1.5 s)	246.6
5 ($n_x = 10$)	3 + 1 (25 + 0)	128.5 s (1.9 s)	1009.1

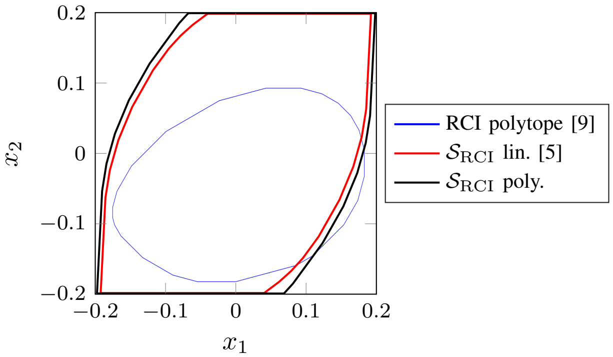

Fig. 2: Comparison with the approach from [9].

C. Comparison with an Approach from the Literature

We consider the Moore-Greitzer model of a jet engine whose dynamics are governed by [48]

$$\begin{aligned}\dot{x}_1 &= -x_2 - \frac{3}{2}x_1^2 - \frac{1}{2}x_1^3 + w,\\ \dot{x}_2 &= u.\end{aligned} \tag{23}$$

Measurements are taken with a sampling time of $\Delta t = 0.1$ time units. The origin is chosen as the equilibrium for the computation of $\mathcal{S}_{\text{RCI}}$.

In [9], an RCI polytope and a corresponding polynomial controller for the continuous-time polynomial dynamics are computed by solving a sequence of linear programs. The corresponding result from [9] for a linear controller and $\mathcal{S}_{\text{RCI}}$ for both our approaches are shown in Fig. 2.

Even though the authors in [9] do not need to abstract the polynomial dynamics and represent the RCI set using polytopes, which offer more flexibility compared to zonotopes, our approaches compute a significantly larger RCI set. Among others, this is due to the increased flexibility of our sampled-data setting compared to enforcing invariance for every point in time. In addition, our approach is easily applicable to general nonlinear systems, whereas it is not clear whether and how the computations in [9] can be automated.

VI. Conclusion

We presented a novel, formally correct approach for computing RCI sets of perturbed nonlinear sampled-data systems. By combining scalable reachability analysis with numerical optimization, we designed an algorithm that computes a sequence of candidate RCI sets with monotonically increasing volume. In addition to using polynomial difference inclusions without limiting the order of the abstraction, we introduced the center of the zonotopic RCI set as an optimization variable. We show that every iteration of our approach has polynomial complexity in the state dimension. Our results demonstrate the broad applicability of our approach and notable improvements in terms of the volume of the computed RCI sets compared to state-of-the-art approaches.

Appendix I
Proof of Lemma 1

We assume $\kappa = 2$ for simplicity since the extension to higher-order abstractions is straightforward. By plugging the approximation $\Xi([0,0]) \approx \Xi([0,\Delta t])$ from Sec. III-B into the reachability algorithm presented in Sec. II-D, we obtain

$$\widetilde{\mathcal{R}}^{(i)}\left(\Delta t, \mathcal{S}, u_{\mathcal{S}}\right) = \mathcal{R}_{\text{lin}}\left(\Delta t, \mathcal{S}, u_{\mathcal{S}}\right) \boxplus \mathcal{R}_{\text{abs}}\left(\Xi([0,0])\right),$$

where $\mathcal{R}_{\text{lin}}\left(\Delta t, \mathcal{S}, u_{\mathcal{S}}\right)$ returns the reachable sets of the linearized dynamics $f_{\text{lin}}(x(t), u(t), w(t))$ and $\boxplus$ denotes the exact addition as defined in [34, Prop. 3.1.20]. $\mathcal{R}_{\text{abs}}\left(\Xi([0,0])\right)$ returns the reachable set due to $\Xi([0,0])$ and consists of the input solution due to

- the constant input $\textsc{quadMap}\left(\mathcal{H}, \mathcal{S}, u_{\mathcal{S}}\right)$, which evaluates the quadratic forms due to the set of Hessians $\mathcal{H}$ of the dynamics in (1); and
- the approximation $\Psi^{(i)}\left(\mathcal{S}\right)$ of the Lagrange remainder.

Since we choose $(x_{\text{eq}}, u_{\text{eq}}, \mathbf{0})$ as the expansion point $\forall i \in \mathbb{N}_0$, $\mathcal{R}_{\text{lin}}\left(\Delta t, \mathcal{S}^{(i)}, u_{\mathcal{S}}^{(i)}\right)$ and $\textsc{quadMap}\left(\mathcal{H}, \mathcal{S}^{(i)}, u_{\mathcal{S}}^{(i)}\right)$ are identical for two subsequent iterations $\widetilde{\mathcal{R}}^{(i)}\left(\Delta t, \mathcal{S}^{(i)}, u_{\mathcal{S}}^{(i)}\right)$, $\widetilde{\mathcal{R}}^{(i+1)}\left(\Delta t, \mathcal{S}^{(i)}, u_{\mathcal{S}}^{(i)}\right)$. By assumption, the constraints in (19e) hold with equality and

$\tilde{\psi}_j^{(i)}\left(\mathcal{S}\right)$, $\tilde{\psi}_j^{(i+1)}\left(\mathcal{S}\right)$ are chosen according to the condition in (20a). Thus, we obtain $\Psi^{(i+1)}\left(\mathcal{S}^{(i)}\right) \subseteq \Psi^{(i)}\left(\mathcal{S}^{(i)}\right)$ from (16). The claim then follows by computing the input solutions as described in [29, Sec. 3.2].

ACKNOWLEDGMENT

We gratefully acknowledge financial support by the project justITSELF funded by the European Research Council (ERC) under grant agreement No 817629 and the German Federal Ministry for Economic Affairs and Climate Action project VaF under grant KK5135901KG0. Furthermore, we would like to thank Victor Gaßmann and Mark Wetzlinger for fruitful discussions on the topic.

REFERENCES

[1] D. Q. Mayne, "Robust and stochastic model predictive control: Are we going in the right direction?" *Annual Reviews in Control*, vol. 41, pp. 184–192, 2016.

[2] F. Blanchini, "Set invariance in control," *Automatica*, vol. 35, no. 11, pp. 1747–1767, 1999.

[3] I. M. Mitchell, J. Yeh, F. J. Laine, and C. J. Tomlin, "Ensuring safety for sampled data systems: An efficient algorithm for filtering potentially unsafe input signals," in *IEEE Conference on Decision and Control*, 2016, pp. 7431–7438.

[4] H. Krasowski, J. Thumm, M. Müller, L. Schäfer, X. Wang, and M. Althoff, "Provably safe reinforcement learning: Conceptual analysis, survey, and benchmarking," *Transactions on Machine Learning Research*, 2023.

[5] L. Schäfer, F. Gruber, and M. Althoff, "Scalable computation of robust control invariant sets of nonlinear systems," *IEEE Transactions on Automatic Control*, vol. 69, no. 2, pp. 755–770, 2024.

[6] M. Fiacchini, T. Alamo, and E. Camacho, "On the computation of convex robust control invariant sets for nonlinear systems," *Automatica*, vol. 46, no. 8, pp. 1334–1338, 2010.

[7] ——, "Invariant sets computation for convex difference inclusions systems," *Systems & Control Letters*, vol. 61, no. 8, pp. 819–826, 2012.

[8] A. Sala, C. Ariño, and R. Robles, "Gain-scheduled control via convex nonlinear parameter varying models," *IFAC-PapersOnLine*, vol. 52, no. 28, pp. 70–75, 2019.

[9] M. A. Ben Sassi and A. Girard, "Controller synthesis for robust invariance of polynomial dynamical systems using linear programming," *Systems & Control Letters*, vol. 61, no. 4, pp. 506–512, 2012.

[10] M. A. Ben Sassi, A. Girard, and S. Sankaranarayanan, "Iterative computation of polyhedral invariants sets for polynomial dynamical systems," in *IEEE Conference on Decision and Control*, 2014, pp. 6348–6353.

[11] R. Robles, A. Sala, and M. Bernal, "Performance-oriented quasi-LPV modeling of nonlinear systems," *International Journal of Robust and Nonlinear Control*, vol. 29, no. 5, pp. 1230–1248, 2019.

[12] M. Fiacchini, "Convex difference inclusions for systems analysis and design," Ph.D. dissertation, Universidad de Sevilla. Departamento de Ingeniería de Sistemas y Automática, 2010.

[13] M. A. Ben Sassi and A. Girard, "Computation of polytopic invariants for polynomial dynamical systems using linear programming," *Automatica*, vol. 48, no. 12, pp. 3114–3121, 2012.

[14] S. Bansal, M. Chen, S. Herbert, and C. J. Tomlin, "Hamilton-Jacobi reachability: A brief overview and recent advances," in *IEEE Conference on Decision and Control*, 2017, pp. 2242–2253.

[15] B. Xue, Q. Wang, N. Zhan, and M. Fränzle, "Robust invariant sets generation for state-constrained perturbed polynomial systems," in *International Conference on Hybrid Systems: Computation and Control*, 2019, pp. 128–137.

[16] B. Xue, Q. Wang, N. Zhan, S. Wang, and Z. She, "Synthesizing robust domains of attraction for state-constrained perturbed polynomial systems," *SIAM Journal on Control and Optimization*, vol. 59, no. 2, pp. 1083–1108, 2021.

[17] A. Oustry, C. Cardozo, P. Pantiatici, and D. Henrion, "Maximal positively invariant set determination for transient stability assessment in power systems," in *IEEE Conference on Decision and Control*, 2019, pp. 6572–6577.

[18] M. Korda, D. Henrion, and C. N. Jones, "Convex computation of the maximum controlled invariant set for polynomial control systems," *SIAM Journal on Control and Optimization*, vol. 52, no. 5, pp. 2944–2969, 2014.

[19] C. Schlosser and M. Korda, "Converging outer approximations to global attractors using semidefinite programming," *Automatica*, vol. 134, art. no. 109900, 2021.

[20] A. D. Ames, S. Coogan, M. Egerstedt, G. Notomista, K. Sreenath, and P. Tabuada, "Control barrier functions: Theory and applications," in *European Control Conference*, 2019, pp. 3420–3431.

[21] M. Rauscher, M. Kimmel, and S. Hirche, "Constrained robot control using control barrier functions," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2016, pp. 279–285.

[22] X. Xu, P. Tabuada, J. W. Grizzle, and A. D. Ames, "Robustness of control barrier functions for safety critical control," *IFAC-PapersOnLine*, vol. 48, no. 27, pp. 54–61, 2015.

[23] A. A. Ahmadi, G. Hall, A. Papachristodoulou, J. Saunderson, and Y. Zheng, "Improving efficiency and scalability of sum of squares optimization: Recent advances and limitations," in *IEEE Conference on Decision and Control*, 2017, pp. 453–462.

[24] W.-H. Chen, J. O'Reilly, and D. Ballance, "On the terminal region of model predictive control for non-linear systems with input/state constraints," *International Journal of Adaptive Control and Signal Processing*, vol. 17, pp. 195–207, 2003.

[25] S. Yu, C. Maier, H. Chen, and F. Allgöwer, "Tube MPC scheme based on robust control invariant set with application to Lipschitz nonlinear systems," *Systems & Control Letters*, vol. 62, no. 2, pp. 194–200, 2013.

[26] M. Lazar and M. Tetteroo, "Computation of terminal costs and sets for discrete–time nonlinear MPC," *IFAC-PapersOnLine*, vol. 51, no. 20, pp. 141–146, 2018.

[27] J. Bravo, D. Limon, T. Alamo, and E. Camacho, "On the computation of invariant sets for constrained nonlinear systems: An interval arithmetic approach," *Automatica*, vol. 41, no. 9, pp. 1583–1589, 2005.

[28] S. Brown, M. Khajenejad, S. Z. Yong, and S. Martínez, "Computing controlled invariant sets of nonlinear control-affine systems," in *IEEE Conference on Decision and Control*, 2023, pp. 7830–7836.

[29] M. Althoff, "Reachability analysis of nonlinear systems using conservative polynomialization and non-convex sets," in *International Conference on Hybrid Systems: Computation and Control*, 2013, pp. 173–182.

[30] V. Gaßmann and M. Althoff, "Verified polynomial controller synthesis for disturbed nonlinear systems," *IFAC-PapersOnLine*, vol. 54, no. 5, pp. 85–90, 2021.

[31] A. Platzer and E. M. Clarke, "The image computation problem in hybrid systems model checking," in *Hybrid Systems: Computation and Control*. Springer Berlin, Heidelberg, 2007, pp. 473–486.

[32] I. M. Mitchell, "Comparing forward and backward reachability as tools for safety analysis," in *Hybrid Systems: Computation and Control*. Springer Berlin, Heidelberg, 2007, pp. 428–443.

[33] 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.

[34] N. Kochdumper, "Extensions of polynomial zonotopes and their application to verification of cyber-physical systems," Ph.D. dissertation, Technische Universität München, 2022.

[35] K. Ghasemi, S. Sadraddini, and C. Belta, "Compositional synthesis via a convex parameterization of assume-guarantee contracts," in *International Conference on Hybrid Systems: Computation and Control*, 2020, pp. 1–10.

[36] B. Schürmann, R. Vignali, M. Prandini, and M. Althoff, "Set-based control for disturbed piecewise affine systems with state and actuation constraints," *Nonlinear Analysis: Hybrid Systems*, vol. 36, art. no. 100826, 2020.

[37] I. Mitchell, J. Budzis, and A. Bolyachevets, "Invariant, viability and discriminating kernel under-approximation via zonotope scaling: Poster abstract," in *International Conference on Hybrid Systems: Computation and Control*, 2019, pp. 268–269.

[38] E. Gover and N. Krikorian, "Determinants and the volumes of parallelotopes and zonotopes," *Linear Algebra and its Applications*, vol. 433, no. 1, pp. 28–40, 2010.

[39] C. Cartis, N. I. M. Gould, and P. L. Toint, "On the evaluation complexity of constrained nonlinear least-squares and general constrained nonlinear optimization using second-order methods," *SIAM Journal on Numerical Analysis*, vol. 53, no. 2, pp. 836–851, 2015.

[40] J. Czekansky and T. Sauer, "The multivariate horner scheme revisited," *BIT Numerical Mathematics*, vol. 55, pp. 1043–1056, 2015.

[41] N. Kochdumper, F. Gruber, B. Schürmann, V. Gaßmann, M. Klischat, and M. Althoff, "AROC: A toolbox for automated reachset optimal controller synthesis," in *International Conference on Hybrid Systems: Computation and Control*, 2021, pp. 1–6.

[42] M. Althoff, "An introduction to CORA 2015," in *Proc. of the Workshop on Applied Verification for Continuous and Hybrid Systems*, 2015, pp. 120–151.

[43] A. Wächter and L. T. Biegler, "On the implementation of a primal-dual interior point filter line search algorithm for large-scale nonlinear programming," *Math. Programming*, vol. 106, no. 1, pp. 25–57, 2006.

[44] M. E. Villanueva, J. C. Li, X. Feng, B. Chachuat, and B. Houska, "Computing ellipsoidal robust forward invariant tubes for nonlinear MPC," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 7175–7180, 2017.

[45] R. Findeisen, "Nonlinear model predictive control : a sampled data feedback perspective," Ph.D. dissertation, Universität Stuttgart, 2005.

[46] J. Theis, "Sum-of-squares applications in nonlinear controller synthesis," Master's thesis, University of California, Berkeley, 2012.

[47] H. Yin, A. Packard, M. Arcak, and P. Seiler, "Finite horizon backward reachability analysis and control synthesis for uncertain nonlinear systems," in *American Control Conference*, 2019, pp. 5020–5026.

[48] M. Krstic, I. Kanellakopoulos, and P. V. Kokotovic, *Nonlinear and Adaptive Control Design*, 1st ed. Wiley-Interscience, 1995.