Computing Safe Sets of Linear Sampled-Data Systems

Felix Gruber and Matthias Althoff

***Abstract*—Leveraging autonomous systems in safety-critical applications requires formal robustness guarantees against uncertainties. We address this issue by computing safe terminal sets with corresponding safety-preserving terminal controllers, which ensure robust constraint satisfaction for an infinite time horizon. To maximize the region of operation, we also construct as large as possible safe initial sets that can be safely steered into the safe terminal set in finite time. We use scalable reachability analysis and convex optimization to efficiently compute safe sets of sampled-data systems. These systems are composed of a physical plant evolving in continuous time and a digital controller being implemented in discrete time. We further verify the effectiveness of our robust control approach using a simple double-integrator system and a vehicle-platooning benchmark.**

***Index Terms*—Robust control, predictive control for linear systems, sampled-data control.**

I. Introduction

GUARANTEEING safety for an infinite time horizon is crucial yet challenging to verify when deploying autonomous systems or learning-based control in safety-critical applications. Thus, the state sets that guarantee robust state and input constraint satisfaction at all times are widely used in the robust control synthesis.

For instance, ensuring recursive feasibility in robust model predictive control can be achieved by using a safe robust invariant terminal set with a corresponding terminal penalty [1], [2]. As soon as the system state enters this set, the safety-preserving terminal controller guarantees the satisfaction of the state and input constraints at all times. Recently, safe sets are also used in safe learning-based control as part of the supervisory safety filter [3], [4]. This filter accepts only inputs satisfying the input constraint and causing the state of the system to stay within the safe set. If the desired control input is rejected, the safety-preserving backup control is applied instead.

The largest safe set is known as the discriminating kernel, maximal robust control invariant (RCI) set, or infinite reachable set [2], [5]. Because of its high relevance in robust control synthesis, computing the exact discriminating kernel and approximations thereof has a rich history. The exact set for discrete-time systems can be obtained by standard set recursion [2], [6]. However, the procedure fails to terminate in finite time in most cases. Thus, various approaches for computing approximations have been proposed in the literature.

This work was supported in part by the European Commission Projects justITSELF and interACT under Grant 817629 and Grant 723395.

The authors are with the Department of Informatics, Technical University of Munich, Boltzmannstr. 3, 85748 Garching bei München, Germany. E-mail: {felix.gruber, althoff}@tum.de

Digital Object Identifier: 10.1109/LCSYS.2020.3002476

Polytopic RCI under- and over-approximations are presented in [5], where arbitrarily small violations of the state and input constraints are tolerated in the case of an over-approximation. To prevent the polytopic representation of an RCI set from becoming too complex, its desired number of representing halfspaces can be chosen freely in [7]. To obtain RCI sets of desired complexity in the case of linear state feedback control, a sequence of semi-definite programs is solved. In contrast to explicit representations, RCI sets are represented implicitly in [8], where the corresponding safety-preserving control is obtained by solving a convex optimization problem.

To improve computational complexity when constructing an under-approximation of the finite-horizon discriminating kernel, ellipsoids instead of polytopes are used as a set representation in [9]. However, safety is ensured only for a finite time horizon. Nevertheless, compared with the exponential complexity of the standard polytopic approach with respect to the state space dimension, representing reachable sets by ellipsoids results in increased scalability. As a scalable alternative, zonotopes are used as a set representation in [10]. Because zonotopes can exactly represent typical axis-aligned box constraints, zonotopic approximations often produce less conservative results compared with ellipsoidal ones [10].

In this letter, by using convex optimization, we efficiently compute zonotopic safe sets with corresponding controllers that ensure robust constraint satisfaction for an infinite time horizon. Inspired by [3], we compute a) safe terminal sets that guarantee robust constraint satisfaction at all times and b) safe initial sets that are as large as possible and can be safely steered into a safe terminal set in finite time. Moreover, we consider systems that are described by sampled-data models [11], where the physical plant evolves in continuous time, whereas the digital controller is implemented in discrete time.

The rest of this letter is structured as follows: In Section II, zonotopes as an efficient set representation are introduced and the control goal is formulated. Subsequently, our reachability analysis is presented in Section III, followed by the computation of safe sets in Section IV. Finally, two numerical examples are considered in Section V, and conclusions in addition to suggestions for future work are provided in Section VI.

II. Preliminaries

In this section, we introduce zonotopes as an efficient set representation. Additionally, we recall two approaches for determining whether a zonotope contains another zonotope. Finally, we state the control goal.

A. Set Representation by Zonotopes

A crucial aspect when computing reachable sets is the choice of set representation. We use zonotopes so that the computational complexity of our reachability analysis algorithm scales only cubicly with the dimension of the state space [12]. Moreover, zonotopes can be stored efficiently as matrices and are closed under Minkowski addition and linear transformation [13].

A zonotope $\mathcal{Z} \subset \mathbb{R}^{n_z}$ in generator representation is defined by

$$\mathcal{Z} = \left\{ z \in \mathbb{R}^{n_z} \;\middle|\; z = c + G\lambda,\ \|\lambda\|_\infty \leq 1 \right\},$$

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 of $\mathcal{Z}$ with $\eta(\mathcal{Z})$ denoting the number of generators. To obtain a more compact notation, we use $\mathcal{Z} = \langle c, G \rangle$. The order of $\mathcal{Z}$ is defined by $o(\mathcal{Z}) = \frac{\eta(\mathcal{Z})}{n_z}$.

According to [13], the Minkowski addition of two zonotopes $\mathcal{Z}_1 = \langle c_1, G_1 \rangle$ and $\mathcal{Z}_2 = \langle c_2, G_2 \rangle$, where c_1 and c_2 have the same size, and the multiplication by a matrix M are

$$\begin{aligned} \mathcal{Z}_1 \oplus \mathcal{Z}_2 &= \left\{ z_1 + z_2 \;\middle|\; z_1 \in \mathcal{Z}_1,\ z_2 \in \mathcal{Z}_2 \right\} \\ &= \left\langle c_1 + c_2, \begin{bmatrix} G_1 & G_2 \end{bmatrix} \right\rangle \\ M\mathcal{Z}_1 &= \langle Mc_1, MG_1 \rangle. \end{aligned}$$

Additionally, we introduce the directed Hausdorff distance

$$d(\mathcal{Z}_1, \mathcal{Z}_2) = \min \left\{ \delta \in \mathbb{R}_{\geq 0} \;\middle|\; \mathcal{Z}_1 \subseteq \mathcal{Z}_2 \oplus \delta \langle \mathbf{0}, I \rangle \right\} \tag{1}$$

between $\mathcal{Z}_1$ and $\mathcal{Z}_2$ [14], where $\langle \mathbf{0}, I \rangle$ denotes the unit ball corresponding to the infinity norm. Thus, $d(\mathcal{Z}_1, \mathcal{Z}_2) = 0$ if and only if $\mathcal{Z}_1 \subseteq \mathcal{Z}_2$. Moreover, according to [12], the smallest axis-aligned box enclosure of $\mathcal{Z} = \langle c, G \rangle$ is

$$\mathsf{boxEnclosure}(\mathcal{Z}) = \langle c, \mathsf{diag}(|G|\mathbf{1}) \rangle, \tag{2}$$

where the absolute value is applied elementwise, $\mathbf{1}$ denotes a vector of ones, and the function diag returns a diagonal matrix with the input vector as the diagonal.

B. Zonotope Containment

We recall two approaches for determining if a zonotope $\mathcal{Z}_1$ is contained within another zonotope $\mathcal{Z}_2$, i.e., if $\mathcal{Z}_1 \subseteq \mathcal{Z}_2$ holds. The first approach transforms $\mathcal{Z}_2$ from generator to halfspace representation [15]. Then, $\mathcal{Z}_1 = \langle c_1, G_1 \rangle$ is contained in $\mathcal{Z}_2 = \left\{ z_2 \in \mathbb{R}^{n_{z_2}} \;\middle|\; H_2 z_2 \leq h_2 \right\}$ if and only if

$$H_2 c_1 + \left| H_2 G_1 \right| \mathbf{1} \leq h_2 \tag{3}$$

is fulfilled [16], where the inequality is applied elementwise.

The second approach solves a linear feasibility problem [14]. To determine whether $\mathcal{Z}_1 = \langle c_1, G_1 \rangle$ is contained in $\mathcal{Z}_2 = \langle c_2, G_2 \rangle$, it is checked if a matrix Γ and a vector γ exist such that

$$G_1 = G_2 \Gamma \tag{4a}$$

$$c_2 - c_1 = G_2 \gamma \tag{4b}$$

$$\left\| \begin{bmatrix} \Gamma & \gamma \end{bmatrix} \right\|_\infty \leq 1, \tag{4c}$$

where the infinity norm is defined as the maximum absolute row sum. In contrast to (3), (4) is only a sufficient condition for zonotope containment. Nevertheless, (4) can be solved for comparably higher-order zonotopes by using efficient convex optimization algorithms [17].

C. Problem Statement

We consider continuous-time linear time-invariant systems that evolve according to

$$\dot{x}(t) = Ax(t) + Bu(t) + w(t), \tag{5}$$

where $x(t) \in \mathbb{R}^{n_x}$ is the system state, $u(t) \in \mathbb{R}^{n_u}$ is the input, and $w(t) \in \mathbb{R}^{n_x}$ is the unknown disturbance at time $t \in \mathbb{R}_{\geq 0}$. The disturbance trajectory $w(\cdot)$ is bounded by the disturbance set $\mathcal{W} \subset \mathbb{R}^{n_x}$, i.e., $w(t) \in \mathcal{W}$ for all times t. To obtain a more compact notation, we use $w(\cdot) \in \mathcal{W}$. Moreover, the system in (5) is constrained by

$$x(\cdot) \in \mathcal{X} \tag{6a}$$

$$u(\cdot) \in \mathcal{U}, \tag{6b}$$

where $\mathcal{X} \subset \mathbb{R}^{n_x}$ and $\mathcal{U} \subset \mathbb{R}^{n_u}$ are the state and input constraint sets, respectively. We assume that $\mathcal{X}$, $\mathcal{U}$, and $\mathcal{W}$ contain the origin. Additionally, $\mathcal{X}$ and $\mathcal{W}$ are assumed to be given in generator representation, whereas $\mathcal{U}$ is provided in generator or halfspace representation. Because these three sets are typically described by axis-aligned boxes, they can easily be expressed in both representations.

The initial state of the system $x(0)$ lies within the initial state set $\mathcal{Z}_{\text{init}} = \langle c_{\text{init}}, G_{\text{init}} \rangle \subseteq \mathcal{X}$, i.e., it can be expressed by

$$x(0) = c_{\text{init}} + G_{\text{init}} \lambda_{\text{init}}, \tag{7}$$

where a not necessarily unique initial scaling vector $\lambda_{\text{init}} \in \mathbb{R}^{\eta(\mathcal{Z}_{\text{init}})}$ with $\|\lambda_{\text{init}}\|_\infty \leq 1$ exists. Based on λ_{init}, the digital controller provides a piecewise constant control signal only at periodic sampling time points $t_k = k\Delta t$ with $k \in \mathbb{Z}_{\geq 0}$ and $\Delta t \in \mathbb{R}_{>0}$. To define a meaningful sampled-data control problem, we assume that the tuple (A_D, B_D) with $A_D = e^{A\Delta t}$ and $B_D = \left(\int_0^{\Delta t} e^{A\tau} \, \mathrm{d}\tau \right) B$ is stabilizable. Based on a stabilizing feedback matrix $K \in \mathbb{R}^{n_u \times n_x}$, we use the simple control law

$$u(t) = Kx(t_k) + c_u(t_k) + G_u(t_k)\lambda_{\text{init}} \quad \text{for } t \in [t_k, t_{k+1}), \tag{8}$$

where $\mathcal{Z}_u(t_k) = \langle c_u(t_k), G_u(t_k) \rangle$ with generator matrix $G_u(t_k) \in \mathbb{R}^{n_u \times \eta(\mathcal{Z}_{\text{init}})}$ is the correction input zonotope at t_k. Thus, in addition to the zonotopic parameterized control used in [10], our controller in (8) also consists of a state feedback component.

In this letter, the control goal is to find a large initial state set $\mathcal{Z}_{\text{init}}$ with correction input zonotope sequence $\mathcal{Z}_u(\cdot)$ such that the constraints in (6) are satisfied for an infinite time horizon.

III. Reachability Analysis

In this section, we compute reachable sets for discrete time points $t \in [0, \Delta t)$. Subsequently, we extend this approach for the entire time horizon $t \in \mathbb{R}_{\geq 0}$.

A. First Time Interval

To accommodate for the piecewise constant control law in (8), we augment the state space:

$$\underbrace{\begin{bmatrix}\dot{x}(t)\\ \dot{u}(t)\end{bmatrix}}_{\dot{\tilde{x}}(t)} = \underbrace{\begin{bmatrix}A & B\\ \mathbf{0} & \mathbf{0}\end{bmatrix}}_{\widetilde{A}} \underbrace{\begin{bmatrix}x(t)\\ u(t)\end{bmatrix}}_{\tilde{x}(t)} + \underbrace{\begin{bmatrix}w(t)\\ \mathbf{0}\end{bmatrix}}_{\widetilde{w}(t)}, \tag{9}$$

where $\mathbf{0}$ denotes a matrix of zeros. To project a set of augmented states $\widetilde{\mathcal{Z}} \subset \mathbb{R}^{n_x+n_u}$ onto the original state and input space, respectively, we define

$$\Pi_x\big(\widetilde{\mathcal{Z}}\big) = \left\{ x \in \mathbb{R}^{n_x} \;\middle|\; u \in \mathbb{R}^{n_u},\ \begin{bmatrix}x\\ u\end{bmatrix} \in \widetilde{\mathcal{Z}} \right\}$$
$$\Pi_u\big(\widetilde{\mathcal{Z}}\big) = \left\{ u \in \mathbb{R}^{n_u} \;\middle|\; x \in \mathbb{R}^{n_x},\ \begin{bmatrix}x\\ u\end{bmatrix} \in \widetilde{\mathcal{Z}} \right\}.$$

For instance, the center and generator matrix of the zonotope $\Pi_u\big(\langle c, G\rangle\big)$ are obtained by deleting the first n_x rows of c and G, respectively.

The unique solution of (9) at time $t \in [0, \Delta t)$ is denoted by $\tilde{\xi}\big(t, \tilde{x}(0), w(\cdot)\big) \in \mathbb{R}^{n_x+n_u}$, where $\tilde{x}(0)$ is the augmented initial state and $w(\cdot)$ is the disturbance trajectory. When considering the disturbance set $\mathcal{W}$ and an augmented initial state set $\widetilde{\mathcal{Z}}_{\text{init}} \subset \mathbb{R}^{n_x+n_u}$, instead of a single initial state $\tilde{x}(0)$, we obtain the exact reachable set

$$\widetilde{\mathcal{R}}_{\text{exact}}(t, \widetilde{\mathcal{Z}}_{\text{init}}) = \big\{ \tilde{x}(t) \in \mathbb{R}^{n_x+n_u} \;\big|\; \tilde{x}(0) \in \widetilde{\mathcal{Z}}_{\text{init}}, \\ w(\cdot) \in \mathcal{W},\ \tilde{x}(t) = \tilde{\xi}\big(t, \tilde{x}(0), w(\cdot)\big) \big\},$$

which is the set of augmented states that the system in (9) can reach at time $t \in [0, \Delta t)$. Because it is impossible to obtain this set for general systems [18], [19], we settle for tight zonotopic over-approximations $\widetilde{\mathcal{R}}_{\text{over}}(t, \widetilde{\mathcal{Z}}_{\text{init}}) \supseteq \widetilde{\mathcal{R}}_{\text{exact}}(t, \widetilde{\mathcal{Z}}_{\text{init}})$ that are computed according to [12], [20].

B. Entire Time Horizon

Based on the reachability analysis for $[0, \Delta t)$, we present Alg. 1 to compute reachable sets $\widetilde{\mathcal{R}}_{\text{hybrid}}\big(t, \mathcal{Z}_{\text{init}}, \mathcal{Z}_u(\cdot)\big)$ for any time point $t \in \mathbb{R}_{\geq 0}$. Its inputs are the specified time t, the initial state set $\mathcal{Z}_{\text{init}} = \big\langle c_{\text{init}}, G_{\text{init}} \big\rangle$, and the correction input zonotope sequence $\mathcal{Z}_u(\cdot) = \big\langle c_u(\cdot), G_u(\cdot) \big\rangle$. Essentially, to accommodate for the piecewise constant control law in (8), Alg. 1 computes reachable sets for consecutive time steps of size Δt until the specified time t is reached.

In line 4 of Alg. 1, we initialize the augmented initial state set $\widetilde{\mathcal{Z}}_{\text{init}}$ based on the control law in (8) and the augmented system in (9). In lines 5 to 10, we compute reachable sets for consecutive time steps of size Δt until the specified time t is bigger than t_{k+1} for some k. In line 9, we horizontally concatenate $G_u(t_k)$ and a matrix of zeros to account for the Minkowski addition in line 6 resulting from the disturbance set $\mathcal{W}$. Finally, the reachable set for the specified time t is obtained in line 11 of Alg. 1.

To obtain a more compact notation, we use

$$\mathcal{R}_x\big(t, \mathcal{Z}_{\text{init}}, \mathcal{Z}_u(\cdot)\big) = \Pi_x\Big(\widetilde{\mathcal{R}}_{\text{hybrid}}\big(t, \mathcal{Z}_{\text{init}}, \mathcal{Z}_u(\cdot)\big)\Big)$$
$$\mathcal{R}_u\big(t, \mathcal{Z}_{\text{init}}, \mathcal{Z}_u(\cdot)\big) = \Pi_u\Big(\widetilde{\mathcal{R}}_{\text{hybrid}}\big(t, \mathcal{Z}_{\text{init}}, \mathcal{Z}_u(\cdot)\big)\Big)$$

Algorithm 1 Computing $\widetilde{\mathcal{R}}_{\text{hybrid}}\big(t, \mathcal{Z}_{\text{init}}, \mathcal{Z}_u(\cdot)\big)$

Input: t, $\big\langle c_{\text{init}}, G_{\text{init}} \big\rangle$, $\big\langle c_u(\cdot), G_u(\cdot) \big\rangle$
Output: $\widetilde{\mathcal{Z}}_t$

1: $k \leftarrow 0$
2: $\tilde{c}_{\text{init}} \leftarrow \begin{bmatrix} c_{\text{init}} \\ Kc_{\text{init}} + c_u(t_k) \end{bmatrix}$
3: $\widetilde{G}_{\text{init}} \leftarrow \begin{bmatrix} G_{\text{init}} \\ KG_{\text{init}} + G_u(t_k) \end{bmatrix}$
4: $\widetilde{\mathcal{Z}}_{\text{init}} \leftarrow \big\langle \tilde{c}_{\text{init}}, \widetilde{G}_{\text{init}} \big\rangle$
5: **while** $t_{k+1} < t$ **do**
6: $\quad \big\langle c_x, G_x \big\rangle \leftarrow \Pi_x\big(\widetilde{\mathcal{R}}_{\text{over}}(\Delta t, \widetilde{\mathcal{Z}}_{\text{init}})\big)$
7: $\quad k \leftarrow k + 1$
8: $\quad \tilde{c}_{\text{init}} \leftarrow \begin{bmatrix} c_x \\ Kc_x + c_u(t_k) \end{bmatrix}$
9: $\quad \widetilde{G}_{\text{init}} \leftarrow \begin{bmatrix} G_x \\ KG_x + \begin{bmatrix} G_u(t_k) & \mathbf{0} \end{bmatrix} \end{bmatrix}$
10: $\quad \widetilde{\mathcal{Z}}_{\text{init}} \leftarrow \big\langle \tilde{c}_{\text{init}}, \widetilde{G}_{\text{init}} \big\rangle$
11: $\widetilde{\mathcal{Z}}_t \leftarrow \widetilde{\mathcal{R}}_{\text{over}}(t - t_k, \widetilde{\mathcal{Z}}_{\text{init}})$

to denote the projections of the reachable set onto the original state and input space, respectively. Up to now, we have only performed reachability analysis for discrete time points. Nevertheless, the state and input constraints in (6) must be satisfied not only at but also between sampling times. Thus, we also compute reachable sets

$$\widetilde{\mathcal{R}}_{\text{hybrid}}\big(\tau_k, \mathcal{Z}_{\text{init}}, \mathcal{Z}_u(\cdot)\big) = \bigcup_{t \in \tau_k} \widetilde{\mathcal{R}}_{\text{hybrid}}\big(t, \mathcal{Z}_{\text{init}}, \mathcal{Z}_u(\cdot)\big)$$

for time intervals $\tau_k = [t_k, t_{k+1})$ according to [12], [20].

In summary, we can efficiently compute reachable sets for arbitrary time points and time intervals. In the following section, we use the presented reachable set computations to construct safe sets.

IV. Safe Sets

In this section, we compute safe terminal sets with corresponding terminal controllers that guarantee robust constraint satisfaction for an infinite time horizon. Additionally, to increase the region of operation, we maximize the size of an initial state set while ensuring that all initial states can be safely steered into a safe terminal set.

A. Safe Terminal Set

We call a set $\mathcal{Z}_{\text{ter}} \subseteq \mathcal{X}$ a *safe terminal set* with the corresponding stabilizing terminal controller

$$u(t) = Kx(t_k) \quad \text{for } t \in \tau_k, \tag{10}$$

i.e., $c_u(\cdot)$ and $G_u(\cdot)$ are $\mathbf{0}$ in (8), if there exists some terminal time step $k_{\text{ter}} \in \mathbb{Z}_{>0}$ such that

$$\mathcal{R}_x\big(t_{k_{\text{ter}}}, \mathcal{Z}_{\text{ter}}, \{0\}\big) \subseteq \mathcal{Z}_{\text{ter}} \tag{11a}$$
$$\mathcal{R}_x\big(\tau_k, \mathcal{Z}_{\text{ter}}, \{0\}\big) \subseteq \mathcal{X} \quad \text{for } k \in \{0, 1, 2, \dots, k_{\text{ter}}\} \tag{11b}$$
$$\mathcal{R}_u\big(\tau_k, \mathcal{Z}_{\text{ter}}, \{0\}\big) \subseteq \mathcal{U} \quad \text{for } k \in \{0, 1, 2, \dots, k_{\text{ter}}\} \tag{11c}$$

holds [21]. Thus, in contrast to invariant sets, the state of the system might leave $\mathcal{Z}_{\text{ter}}$ during $(0, t_{k_{\text{ter}}})$. Nevertheless, during

this time, the state and input constraints in (6) are always fulfilled. Consequently, robust constraint satisfaction can be achieved for an infinite time horizon when the initial state lies within $\mathcal{Z}_{\text{ter}}$.

We present Alg. 2 to compute a safe terminal set $\mathcal{Z}_{\text{ter}}$ with a corresponding terminal time step k_{ter}. Its input $\epsilon \in \mathbb{R}_{>0}$ denotes the convergence tolerance, which is typically chosen close to 0. Essentially, Alg. 2 proceeds in two steps. First, two zonotope sequences are computed that converge to an over-approximation of the discrete-time minimal robust positively invariant (mRPI) set [11]. Second, the zonotope order of this over-approximation is reduced as much as possible while ensuring that the constraints in (11b) and (11c) are satisfied.

Algorithm 2 Safe terminal set

Input: ϵ
Output: $\mathcal{Z}_{\text{ter}},\ k_{\text{ter}}$
1: $k \leftarrow 1$
2: $\mathcal{Z}_{\{0\}}(t_k) \leftarrow \mathcal{R}_x\big(t_k, \{0\}, \{0\}\big)$
3: $\mathcal{Z}_{\mathcal{X}}(t_k) \leftarrow \mathcal{R}_x\big(t_k, \mathcal{X}, \{0\}\big)$
4: **while** $\epsilon \leq d\Big(\mathcal{Z}_{\mathcal{X}}(t_k),\ \mathsf{boxEnclosure}\big(\mathcal{Z}_{\{0\}}(t_k)\big)\Big)$ **do**
5: $\quad k \leftarrow k + 1$
6: $\quad \mathcal{Z}_{\{0\}}(t_k) \leftarrow \mathcal{R}_x\big(t_k, \{0\}, \{0\}\big)$
7: $\quad \mathcal{Z}_{\mathcal{X}}(t_k) \leftarrow \mathcal{R}_x\big(t_k, \mathcal{X}, \{0\}\big)$
8: $k_{\text{ter}} \leftarrow k$
9: $o_{\text{ter}} \leftarrow 0$
10: **while** $o_{\text{ter}} < o\big(\mathcal{Z}_{\mathcal{X}}(t_{k_{\text{ter}}})\big)$ **do**
11: $\quad o_{\text{ter}} \leftarrow o_{\text{ter}} + 1$
12: $\quad \mathcal{Z}_{\text{ter}} \leftarrow$ over-approx. of $\mathcal{Z}_{\mathcal{X}}(t_{k_{\text{ter}}})$ having order o_{ter}
13: $\quad$ **if** (11b) and (11c) are satisfied for $\mathcal{Z}_{\text{ter}}$, k_{ter} **then**
14: $\quad\quad$ **break**
15: $\quad$ **else**
16: $\quad\quad \mathcal{Z}_{\text{ter}} \leftarrow \emptyset$

In lines 1 to 7 of Alg. 2, we compute the set of reachable states for consecutive time steps corresponding to the following two initial state sets, namely, the origin $\{0\}$ and the state constraint set $\mathcal{X}$. We denote these zonotope sequences by $\mathcal{Z}_{\{0\}}(\cdot)$ and $\mathcal{Z}_{\mathcal{X}}(\cdot)$. Because the feedback matrix K in (10) is stabilizing, $\mathcal{Z}_{\{0\}}(\cdot)$ and $\mathcal{Z}_{\mathcal{X}}(\cdot)$ would converge to the discrete-time mRPI set [11], if no over-approximation of reachable sets to reduce computational complexity was used. To achieve low computation times, we use a simple convergence criterion in line 4 based on the directed Hausdorff distance in (1). Instead of computing $d\big(\mathcal{Z}_{\mathcal{X}}(t_k), \mathcal{Z}_{\{0\}}(t_k)\big)$ using (4), we use box enclosures because they can be efficiently obtained by (2) and transformed to halfspace representation such that (3) can be applied.

To reduce the complexity of the computations in Section IV-B, we want the safe terminal set $\mathcal{Z}_{\text{ter}} \supseteq \mathcal{Z}_{\mathcal{X}}(t_{k_{\text{ter}}})$ to have a reduced zonotope order compared with $\mathcal{Z}_{\mathcal{X}}(t_{k_{\text{ter}}})$ [22], [23]. Thus, in lines 10 to 16 of Alg. 2, we increment this zonotope order starting from 1 until the constraints in (11b) and (11c) are satisfied. However, if these constraints are even violated for the tight over-approximation $\mathcal{Z}_{\mathcal{X}}(t_{k_{\text{ter}}})$ of the discrete-time mRPI set, Alg. 2 returns an empty set.

Proposition 1: If the first output $\mathcal{Z}_{\text{ter}}$ of Alg. 2 is a nonempty set, then $\mathcal{Z}_{\text{ter}}$ is a safe terminal set with the second output k_{ter} being the corresponding terminal time step. □

Proof: Because $\mathcal{Z}_{\text{ter}}$ is a nonempty set, we know that the check in line 13 of Alg. 2 was passed successfully. Thus, the constraints in (11b) and (11c) are fulfilled for $\mathcal{Z}_{\text{ter}}$ and k_{ter}. Because (11b) is satisfied, it follows that

$$\mathcal{Z}_{\text{ter}} \subseteq \mathcal{R}_x\big(\tau_0, \mathcal{Z}_{\text{ter}}, \{0\}\big) \subseteq \mathcal{X},$$

which results in

$$\mathcal{R}_x\big(t_{k_{\text{ter}}}, \mathcal{Z}_{\text{ter}}, \{0\}\big) \subseteq \mathcal{R}_x\big(t_{k_{\text{ter}}}, \mathcal{X}, \{0\}\big) \overset{\text{line 12 of Alg. 2}}{\subseteq} \mathcal{Z}_{\text{ter}},$$

implying the satisfaction of (11a). ■

B. Safe Initial Set

If the initial state lies within a safe terminal set $\mathcal{Z}_{\text{ter}}$, the terminal controller in (10) ensures the satisfaction of the constraints in (6) for an infinite time horizon. Thus, we could use this safe set, e.g., as a terminal set in robust model predictive control. Nevertheless, we want to obtain a state set that guarantees robust constraint satisfaction at all times while being as large as possible to maximize the region of operation. One way to achieve this goal is to construct a large initial state set that can be safely steered into $\mathcal{Z}_{\text{ter}}$ in finite time. We call such a set a *safe initial set*. Subsequently, we compute under-approximations of the largest safe initial set, which is known as the robust sampled-data capture basin [3].

Ideally, we want to maximize the volume of the safe initial set $\mathcal{Z}_{\text{init}}$. However, computing the volume of a general zonotope is combinatorially complex with respect to the number of columns of the generator matrix [24]. Nevertheless, in the special case of $\mathcal{Z}_{\text{init}}$ being a parallelotope, maximizing the determinant of the generator matrix results in the maximum volume. When constraining this generator matrix to be positive definite, the maximization can be cast and efficiently solved as a convex optimization problem [17]. However, restricting $\mathcal{Z}_{\text{init}}$ to be a parallelotope can be conservative.

Instead of maximizing the actual volume, we use a heuristic based on generator scaling [10]. To obtain a large safe initial set $\mathcal{Z}_{\text{init}}$ with correction input zonotope sequence $\mathcal{Z}_u(\cdot)$, we solve the following convex optimization problem:

$$\begin{aligned}
\max_{\phi, c_{\text{init}}, \mathcal{Z}_u(\cdot)} \quad & J_{\mathcal{Z}_{\text{init}}}(\phi) && \text{(12a)}\\
\text{s. t.} \quad & \mathcal{Z}_{\text{init}} = \big\langle c_{\text{init}}, G_{\text{user}}\, \mathsf{diag}(\phi)\big\rangle && \text{(12b)}\\
& \mathcal{R}_x\big(t_{k_{\text{init}}}, \mathcal{Z}_{\text{init}}, \mathcal{Z}_u(\cdot)\big) \subseteq \mathcal{Z}_{\text{ter}} && \text{(12c)}\\
& \text{for } k \in \{0, 1, 2, \ldots, k_{\text{init}}\}\text{:} && \\
& \mathcal{R}_x\big(\tau_k, \mathcal{Z}_{\text{init}}, \mathcal{Z}_u(\cdot)\big) \subseteq \mathcal{X} && \text{(12d)}\\
& \mathcal{R}_u\big(\tau_k, \mathcal{Z}_{\text{init}}, \mathcal{Z}_u(\cdot)\big) \subseteq \mathcal{U}, && \text{(12e)}
\end{aligned}$$

where $J_{\mathcal{Z}_{\text{init}}}$ is a concave cost function, $\phi \in \mathbb{R}_{\geq 0}^{\eta(\mathcal{Z}_{\text{init}})}$ is a generator scaling vector, $G_{\text{user}} \in \mathbb{R}^{n_x \times \eta(\mathcal{Z}_{\text{init}})}$ is a user-defined generator matrix, and $k_{\text{init}} \in \mathbb{Z}_{>0}$ is the initial time step. To check for zonotope containment, we use the approaches presented in Section II-B, resulting in the maximization of a concave cost function subject to linear constraints.

A reasonable cost function $J_{\mathcal{Z}_{\text{init}}}$ in (12a) is the geometric mean or the sum of the input vector elements. The user-defined generator matrix G_{user} can be chosen as the generator matrix of the obtained safe terminal set $\mathcal{Z}_{\text{ter}}$. Alternatively, we can uniformly sample from the unit hypersphere and use the obtained points as columns of G_{user}. Because uniform sampling in high-dimensional spaces is a complex task, it is beneficial to examine the sparsity of the system matrix [10]. The initial time step k_{init} corresponds to the time the safe terminal set $\mathcal{Z}_{\text{ter}}$ is reached. Thus, this parameter is used to balance between accuracy and computational complexity.

To explicitly obtain the control law in (8) for $t \in [0, t_{k_{\text{init}}})$, we must compute the not necessarily unique initial scaling vector λ_{init} in (7) at time 0. This can be achieved by solving the following convex optimization problem:

$$\min_{\lambda_{\text{init}}} \quad J_{\lambda_{\text{init}}}(\lambda_{\text{init}}) \tag{13a}$$

$$\text{s. t.} \quad x(0) = c_{\text{init}} + G_{\text{init}}\lambda_{\text{init}} \tag{13b}$$

$$\|\lambda_{\text{init}}\|_\infty \leq 1, \tag{13c}$$

where $J_{\lambda_{\text{init}}}$ is a convex cost function. In the special case of $\mathcal{Z}_{\text{init}}$ being a parallelotope, we can invert the optimized generator matrix $G_{\text{init}} = G_{\text{user}}\,\text{diag}(\phi)$ to obtain the unique scaling vector $\lambda_{\text{init}} = G_{\text{init}}^{-1}\big(x(0) - c_{\text{init}}\big)$.

In summary, we propose a dual-mode control approach. During $[0, t_{k_{\text{init}}})$, we safely steer the initial state $x(0) \in \mathcal{Z}_{\text{init}}$ based on the control law in (8) into $\mathcal{Z}_{\text{ter}}$ and switch to the terminal controller in (10) at $t_{k_{\text{init}}}$. Thus, we are able to satisfy the state and input constraints in (6) for an infinite time horizon while providing a large region of operation.

V. Numerical Examples

In this section, we demonstrate the effectiveness of our robust control approach using a simple double-integrator system and a vehicle-platooning benchmark. For both numerical examples, the sampling time is $\Delta t = 0.1\,\text{s}$, and the convergence tolerance used in Alg. 2 is $\epsilon = 0.01$.

For the reachability analysis computations, we use the open-source tool CORA [20]. All optimization problems are modeled using YALMIP [25] with parameter 'allownonconvex' set to 0 and solved using MOSEK [26] with default parameters. Our computations are conducted on a laptop equipped with an Intel Core i7-7820HQ and 32 GB of RAM.

A. Double-Integrator System

We consider the simple double-integrator system

$$\dot{x}(t) = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u(t) + w(t)$$

with disturbance set $\mathcal{W} = [-0.1, 0.1]^2$ and state and input constraint sets $\mathcal{X} = [-1, 1]^2$ and $\mathcal{U} = [-1, 1]$, respectively. The stabilizing feedback matrix K is obtained using LQR-based controller synthesis, where the state and input weighting matrices are identity matrices. The columns of the user-defined generator matrix G_{user} in (12b) are chosen as 10 uniformly distributed points around the top half unit circle. Because we use the linear cost $J_{\mathcal{Z}_{\text{init}}} = \mathbf{1}^T\phi$ in (12a), the convex optimization problem in (12) is a linear program [17].

In Fig. 1, we visualize the zonotopes that are obtained during the execution of Alg. 2. Running this algorithm to obtain the safe terminal set $\mathcal{Z}_{\text{ter}}$ takes 0.2 s.

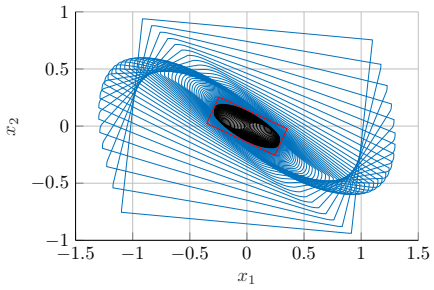
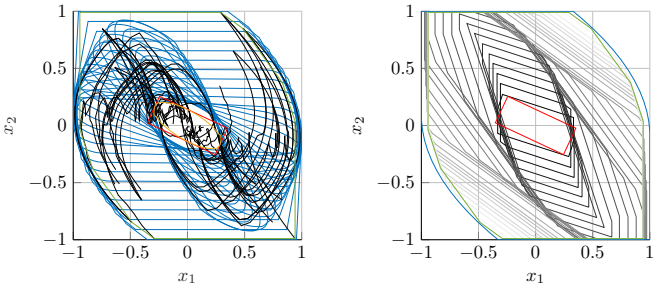


Figure 1. Zonotopes obtained during the execution of Alg. 2. The safe terminal set $\mathcal{Z}_{\text{ter}}$ is shown in red. Additionally, the zonotope sequences $\mathcal{Z}_{\mathcal{X}}(\cdot)$ and $\mathcal{Z}_{\{0\}}(\cdot)$ are plotted in blue and black, respectively.

In Fig. 2a, we present the optimization results for the initial time step $k_{\text{init}} = 30$, i.e., an initial time horizon of 3 s is considered. Solving this linear program takes 0.5 s. To obtain the initial scaling vector λ_{init} for the control law in (8), we use the convex cost $J_{\lambda_{\text{init}}} = \|\lambda_{\text{init}}\|_\infty$ in (13a).



(a) The reachable sets for time intervals and $t_{k_{\text{init}}} = 3\,\text{s}$ are shown in blue and orange, respectively. Additionally, 50 random trajectories for $t \in [0, t_{k_{\text{init}}}]$ are plotted in black.

(b) The safe initial sets for all $k_{\text{init}} \in \{1, 2, \ldots, 30\}$ are shown, where a lighter gray tone corresponds to a higher value of k_{init}. Additionally, a tight invariant under-approximation of the discrete-time maximal RCI set is plotted in blue.

Figure 2. The safe initial set $\mathcal{Z}_{\text{init}}$ for the initial time step $k_{\text{init}} = 30$ and the safe terminal set $\mathcal{Z}_{\text{ter}}$ are visualized in green and red, respectively.

In Fig. 2b, we show the optimized safe initial sets for all initial time steps $k_{\text{init}} \in \{1, 2, \ldots, 30\}$. When increasing k_{init} beyond 30, the terminal constraint in (12c) becomes inactive and the safe initial sets are the same. To demonstrate that our approach is not overly conservative, we also show a tight invariant under-approximation of the discrete-time maximal RCI set, which is computed according to [5]. In contrast to this under-approximation, our safe initial sets guarantee robust constraint satisfaction not only at but also between sampling times.

B. Vehicle-Platooning Benchmark

To demonstrate the applicability of our approach to larger systems, we consider a vehicle-platooning benchmark with

nine states and three inputs [27]. The dynamics corresponding to the i^{th} following vehicle with $i \in \{1, 2, 3\}$ is

$$\ddot{e}_i = a_{i-1} - a_i$$
$$\dot{a}_i = -\frac{1}{T} a_i + \frac{1}{T} u_i,$$

where $T = 0.5\,\text{s}$. The acceleration of the leading vehicle is modeled as disturbance $a_0 \in [-2, 2]\,\frac{\text{m}}{\text{s}^2}$. The state and input constraints for $i \in \{1, 2, 3\}$ are as follows: $e_i \in [-10, 10]\,\text{m}$, $\dot{e}_i \in [5, 5]\,\frac{\text{m}}{\text{s}}$, $a_i \in [-8, 8]\,\frac{\text{m}}{\text{s}^2}$, and $u_i \in [-8, 8]\,\frac{\text{m}}{\text{s}^2}$. The given stabilizing feedback matrix K is obtained using an LMI-based controller synthesis [27].

The user-defined generator matrix G_{user} in (12b) is chosen as the generator matrix of $\mathcal{Z}_{\text{ter}}$. In (12), we use the geometric mean of the input vector as cost $J_{\mathcal{Z}_{\text{init}}}$ and choose $k_{\text{init}} = 40$. Solving the resulting convex optimization problem takes less than 30 s, whereas executing Alg. 2 to obtain $\mathcal{Z}_{\text{ter}}$ takes 1 s. In Fig. 3, we show two-dimensional projections of the optimization results.

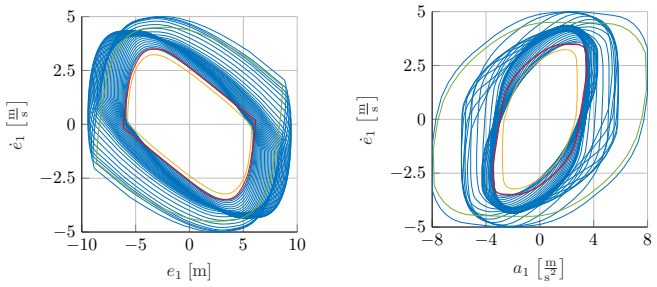


Figure 3. Two-dimensional projections of zonotopes. The safe initial set $\mathcal{Z}_{\text{init}}$ and the safe terminal set $\mathcal{Z}_{\text{ter}}$ are plotted in green and red, respectively. Additionally, the reachable sets for time intervals and $t_{k_{\text{init}}}$ are shown in blue and orange, respectively.

VI. Conclusions and Future Work

We have presented an efficient approach for computing safe sets of linear sampled-data systems subject to additive disturbances. First, we construct safe terminal sets, which are over-approximations of the discrete-time mRPI set. Second, we solve a convex optimization problem for obtaining large safe initial sets to maximize the region of operation. Because we use zonotopes as an efficient set representation, our robust control approach is suitable for ensuring the safety of large systems, as shown in the vehicle-platooning benchmark. In the future, we intend to deploy the obtained safe sets as part of a scalable supervisory safety filter that mediates between safety and performance.

References

[1] D. Q. Mayne, "Model predictive control: Recent developments and future promise," *Automatica*, vol. 50, no. 12, pp. 2967–2986, 2014.

[2] F. Blanchini and S. Miani, *Set-theoretic methods in control*, 2nd ed. Birkhäuser, 2015.

[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] K. P. Wabersich and M. N. Zeilinger, "Linear model predictive safety certification for learning-based control," in *IEEE Conference on Decision and Control*, 2018, pp. 7130–7135.

[5] M. Rungger and P. Tabuada, "Computing robust controlled invariant sets of linear systems," *IEEE Transactions on Automatic Control*, vol. 62, no. 7, pp. 3665–3670, 2017.

[6] D. P. Bertsekas, "Infinite time reachability of state-space regions by using feedback control," *IEEE Transactions on Automatic Control*, vol. 17, no. 5, pp. 604–613, 1972.

[7] A. Gupta and P. Falcone, "Full-complexity characterization of control-invariant domains for systems with uncertain parameter dependence," *IEEE Control Systems Letters*, vol. 3, no. 1, pp. 19–24, 2019.

[8] S. V. Raković and M. Barić, "Parameterized robust control invariant sets for linear systems: Theoretical advances and computational remarks," *IEEE Transactions on Automatic Control*, vol. 55, no. 7, pp. 1599–1614, 2010.

[9] S. Kaynama, I. M. Mitchell, M. Oishi, and G. A. Dumont, "Scalable safety-preserving robust control synthesis for continuous-time linear systems," *IEEE Transactions on Automatic Control*, vol. 60, no. 11, pp. 3065–3070, 2015.

[10] I. M. Mitchell, J. Budzis, and A. Bolyachevets, "Invariant, viability and discriminating kernel under-approximation via zonotope scaling," *arXiv preprint 1901.01006*, 2019.

[11] S. V. Raković, F. A. C. C. Fontes, and I. V. Kolmanovsky, "Reachability and invariance for linear sampled-data systems," in *IFAC World Congress*, 2017, pp. 3057–3062.

[12] M. Althoff, "Reachability analysis of large linear systems with uncertain inputs in the Krylov subspace," *IEEE Transactions on Automatic Control*, vol. 65, no. 2, pp. 477–492, 2020.

[13] W. Kühn, "Rigorously computed orbits of dynamical systems without the wrapping effect," *Computing*, vol. 61, pp. 47–67, 1998.

[14] S. Sadraddini and R. Tedrake, "Linear encodings for polytope containment problems," in *IEEE Conference on Decision and Control*, 2019, pp. 4367–4372.

[15] M. Althoff, O. Stursberg, and M. Buss, "Computing reachable sets of hybrid systems using a combination of zonotopes and polytopes," *Nonlinear Analysis: Hybrid Systems*, vol. 4, no. 2, pp. 233–249, 2010.

[16] 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*, 2020. [Online]. Available: https://doi.org/10.1016/j.nahs.2019.100826

[17] S. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge University Press, 2004.

[18] G. Lafferriere, G. J. Pappas, and S. Yovine, "Symbolic reachability computation for families of linear vector fields," *Journal of Symbolic Computation*, vol. 32, no. 3, pp. 231–253, 2001.

[19] C. Moler and C. Van Loan, "Nineteen dubious ways to compute the exponential of a matrix, twenty-five years later," *SIAM Review*, vol. 45, no. 1, pp. 3–49, 2003.

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

[21] F. Gruber and M. Althoff, "Scalable robust model predictive control for linear sampled-data systems," in *IEEE Conference on Decision and Control*, 2019, pp. 438–444.

[22] A.-K. Kopetzki, B. Schürmann, and M. Althoff, "Methods for order reduction of zonotopes," in *IEEE Conference on Decision and Control*, 2017, pp. 5626–5633.

[23] X. Yang and J. K. Scott, "A comparison of zonotope order reduction techniques," *Automatica*, vol. 95, pp. 378–384, 2018.

[24] 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.

[25] J. Löfberg, "YALMIP : A toolbox for modeling and optimization in MATLAB," in *IEEE International Symposium on Computer Aided Control Systems Design*, 2004, pp. 284–289.

[26] MOSEK Aps, "The MOSEK optimization toolbox for MATLAB manual. Version 8.1," 2019. [Online]. Available: https://docs.mosek.com/8.1/toolbox/index.html

[27] I. Ben Makhlouf and S. Kowalewski, "Networked cooperative platoon of vehicles for testing methods and verification tools," in *Workshop on Applied Verification for Continuous and Hybrid Systems*, 2014, pp. 37–42.