Scalable Robust Safety Filter with Unknown Disturbance Set

Felix Gruber and Matthias Althoff

***Abstract*— Equipping any controller with formal safety guarantees can be achieved by using safety filters. These filters modify the desired control input in the least restrictive way to guarantee safety. However, it is an unresolved issue to construct scalable safety filters without assuming the availability of the disturbance set. We address this issue by proposing an efficient approach to implementing safety filters. In particular, we perform offline set membership identification to obtain a linear model that is conformant to a finite set of training data. Based on this conformant model, we compute a set-based safe backup controller with a corresponding safe set. Because a new measurement obtained online might invalidate the model conformance, we update the model, the safe backup controller, and the safe set online to restore formal safety guarantees. We use scalable reachability analysis and convex optimization algorithms to perform these updates as quickly as possible. We demonstrate the usefulness and scalability of our safety filter approach using four numerical examples from the literature.**

***Index Terms*— Supervisory control, robust control, optimal control, reachability analysis, system identification.**

I. Introduction

EXCELLENT control performance is typically achieved using sophisticated control methods with fine-tuned parameters. Due to the high complexity of these performance controllers, it is usually cumbersome to formally verify safety. Nevertheless, providing formal safety guarantees for any controller can be accomplished using an additional safety filter and a corresponding safe backup controller. Such a filter aims to modify the desired input of the unverified performance controller in a minimally invasive way so that safety is always guaranteed. Therefore, safety filters are supervisory mediators between a simple, safe backup controller and an unverified, sophisticated performance controller.

Because the simple concept of safety filters is compelling, they are used in a wide variety of areas, such as safe reinforcement learning [1], [2], human-in-the-loop control [3], motion planning [4], [5], collision avoidance [6], and fault-tolerant systems [7]. Moreover, different naming conventions have been introduced in the literature because safety filters are widely used in several disciplines. For instance, they are closely related to safety shields [8], verified control envelopes [9], and sandboxing control [10].

Safety filters can be efficiently implemented using, e.g., reachability analysis [11], [12], invariance control [13], [14], barrier functions [15], [16], or command governors [17]. These implementations usually use predictive control techniques [18]–[20], where an optimal control problem is iteratively solved online on a moving time horizon. By increasing this time horizon, the region of operation of the safe backup controller, also known as the safe set, can be enlarged [21].

The conservativeness of a safety filter is mainly determined by the size of its safe set. The largest safe set is known as the discriminating kernel [22], infinite-time reachable set [23], or maximal robust control invariant (MRCI) set [24]. This safe set can be computed for discrete-time, linear time-invariant systems by standard set recursion. Because these computations typically fail to terminate in finite time, various approaches exist for approximating the MRCI set.

Polytopic robust control invariant (RCI) under-approximations and over-approximations are presented in [25], where arbitrarily small constraint violations are tolerated in case of an over-approximation. To prevent the polytopic representation of an RCI set from becoming too complex, the desired number of representing halfspaces can also be fixed [26] or chosen freely [27]. In addition to explicit set representations, polytopic RCI sets can also be represented implicitly, e.g., by the Minkowski sum of a finite number of polytopes [28]. Instead of polytopes, other set representations, such as ellipsoids [24], [29] or zonotopes [30], [31], are also used to decrease the computational complexity. However, most existing approaches are unsuitable for ensuring the safety of large-scale systems due to their conservativeness, exponential computational complexity, or limitation to finite time horizons.

Because formal safety guarantees are model-based, they are only valid as long as the identified model of the unknown system is valid [32]–[34]. However, perfect models are usually unavailable. Thus, control approaches using a finite set of training data have recently gained interest. For instance, a conformant model and an RCI approximation of the minimal RCI set are simultaneously computed in [35]. However, there is no guarantee that an unseen measurement obtained online also lies within this RCI set because only a finite training data set was used for its construction. Thus, additional assumptions are required to provide formal safety guarantees for an infinite time horizon. For instance, the disturbance set is assumed to be known while the system dynamics is unknown [36],

Manuscript received Month Date, Year; revised Month Date, Year. This work was supported by the European Commission Project justITSELF under Grant 817629. The authors are with the Department of Informatics, Technical University of Munich, Boltzmannstr. 3, 85748 Garching b. München, Germany. Email: {felix.gruber, althoff}@tum.de

[37], which is quite unrealistic. The availability of such *a priori* disturbance sets is also a standard assumption in most robust model predictive control (MPC) approaches [18]–[20]. Alternatively, to obtain a margin that ensures safety online, the tightest estimate of the disturbance set can be multiplied by a safety factor greater than one [38], [39]. However, it is unclear how to choose this safety factor without introducing excessive conservativeness to ensure safety for an infinite time horizon.

In this paper, we propose an efficient approach for implementing safety filters. In contrast to the existing methods, our approach is scalable while making no assumptions about the availability of the disturbance set. In particular, we

- perform offline set membership identification to identify a conformant linear model based on a finite set of training data,
- use this identified model to compute an explicit zonotopic safe set with a corresponding safe backup controller,
- present our minimally invasive safety filter algorithm to verify safety online when applying the desired input of the performance controller,
- quickly update the conformant model, the safe set, and the safe backup controller online because a new measurement might invalidate the identified model due to the unknown disturbance set,
- perform these conformance updates in real time even for medium-sized problems, and
- consider all online computation times for solving optimization problems to guarantee safety despite such computational delays.

This paper is organized as follows: Efficient set representations and the control goal are introduced in Section II. In Section III, conformant models are computed based on training data, our set-based safe backup controller is introduced, and large safe sets are constructed. Subsequently, our safety filter algorithm and our online conformance updates are presented in Section IV, followed by a demonstration of its effectiveness using four numerical examples in Section V. Finally, conclusions are provided in Section VI.

II. Preliminaries

In this section, we introduce essential representations of closed, bounded, convex sets. In addition, we recall two approaches for solving the zonotope containment problem. Finally, we formulate the control problem.

A. Set Representations

A polytope can be seen as the intersection of halfspaces and is defined as follows:

Definition 1 (Polytope): A polytope $\mathcal{P} \subset \mathbb{R}^n$ in halfspace representation is defined by

$$\mathcal{P} = \{s \in \mathbb{R}^n \mid Hs \leq h\},$$

where $H \in \mathbb{R}^{m \times n}$ and $h \in \mathbb{R}^m$ are the data representing the halfspaces, $m \in \mathbb{N}_{>0}$ is the number of these halfspaces, and the inequality is applied elementwise. We use $\mathcal{P} = \langle H, h \rangle_P$ to obtain a more concise notation.

A special case of a polytope is a zonotope, which is centrally symmetric and defined as follows:

Definition 2 (Zonotope): A zonotope $\mathcal{Z} \subset \mathbb{R}^n$ in generator representation is defined by

$$\mathcal{Z} = \{s \in \mathbb{R}^n \mid s = c + G\lambda,\ |\lambda| \leq \mathbf{1}\},$$

where $c \in \mathbb{R}^n$ is the center, $G \in \mathbb{R}^{n \times \eta(\mathcal{Z})}$ is the generator matrix with $\eta(\mathcal{Z})$ denoting the number of generators, the absolute value is applied elementwise, and $\mathbf{1}$ denotes a vector of ones. We use $\mathcal{Z} = \langle c, G \rangle_Z$ to obtain a more concise notation.

According to [40], the Minkowski addition of two zonotopes $\mathcal{Z}_1 = \langle c_1, G_1 \rangle_Z \subset \mathbb{R}^n$ and $\mathcal{Z}_2 = \langle c_2, G_2 \rangle_Z \subset \mathbb{R}^n$ and the multiplication by a matrix $M \in \mathbb{R}^{m \times n}$ are computed as

$$\begin{aligned} \mathcal{Z}_1 \oplus \mathcal{Z}_2 &= \{z_1 + z_2 \mid z_1 \in \mathcal{Z}_1,\ z_2 \in \mathcal{Z}_2\} \\ &= \left\langle c_1 + c_2, \begin{bmatrix} G_1 & G_2 \end{bmatrix} \right\rangle_Z, \qquad (1a) \\ M\mathcal{Z}_1 &= \{Mz_1 \mid z_1 \in \mathcal{Z}_1\} \\ &= \langle Mc_1, MG_1 \rangle_Z. \qquad (1b) \end{aligned}$$

Because these two important set operations have a polynomial complexity, zonotopes are well suited for reachability analysis [41], [42], i.e., for representing the set of states a system can reach.

The parameter vector $\lambda \in \mathbb{R}^{\eta(\mathcal{Z})}$ with $|\lambda| \leq \mathbf{1}$ is not necessarily unique for parameterizing any $s \in \langle c, G \rangle_Z \subset \mathbb{R}^n$, unless G is invertible. In this special case, the zonotope is called a parallelotope, and the unique parameter vector is

$$\lambda = G^{-1}(s - c). \qquad (2)$$

Moreover, a parallelotope with a diagonal generator matrix G is called an axis-aligned box. For these boxes, (2) can be efficiently computed because the inverse of a diagonal matrix is obtained by replacing each element on the diagonal with its reciprocal. In addition, axis-aligned boxes can be uniquely represented by their lower and upper bounds, as presented in the following definition.

Definition 3 (Axis-Aligned Box): An axis-aligned box $\mathcal{B} \subset \mathbb{R}^n$ in interval representation is defined by

$$\mathcal{B} = \left\{s \in \mathbb{R}^n \;\middle|\; \underline{\mathcal{B}} \leq s \leq \overline{\mathcal{B}}\right\},$$

where $\underline{\mathcal{B}} \in \mathbb{R}^n$ and $\overline{\mathcal{B}} \in \mathbb{R}^n$ denote the lower and upper bound of $\mathcal{B}$, respectively. We use $\mathcal{B} = \left[\underline{\mathcal{B}}, \overline{\mathcal{B}}\right]$ to obtain a more concise notation.

A conversion from the generator representation $\langle c_\mathcal{B}, G_\mathcal{B} \rangle_Z$ of the axis-aligned box $\mathcal{B}$ to its interval representation $\left[\underline{\mathcal{B}}, \overline{\mathcal{B}}\right]$ is achieved by $\left[\underline{\mathcal{B}}, \overline{\mathcal{B}}\right] = \left[c_\mathcal{B} - \texttt{diag}(|G_\mathcal{B}|),\ c_\mathcal{B} + \texttt{diag}(|G_\mathcal{B}|)\right]$ and vice versa by $c_\mathcal{B} = 0.5\left(\underline{\mathcal{B}} + \overline{\mathcal{B}}\right)$, $G_\mathcal{B} = 0.5\,\texttt{diag}\left(\overline{\mathcal{B}} - \underline{\mathcal{B}}\right)$. Here, the function `diag` returns a diagonal matrix with the input as the diagonal if the input is a vector; otherwise, `diag` returns a vector of the diagonal elements of the input matrix. These conversions directly follow from Definitions 2 and 3.

B. Zonotope Containment

We recall two approaches for determining if a zonotope $\mathcal{Z}_1 \subset \mathbb{R}^n$ is contained within another zonotope $\mathcal{Z}_2 \subset \mathbb{R}^n$, which is co-NP-complete [43]. The first zonotope containment

approach transforms $\mathcal{Z}_2$ from generator to halfspace representation [44], which is usually a computationally complex task [43]. According to [45], $\mathcal{Z}_1 = \langle c_1, G_1 \rangle_Z$ is contained in $\mathcal{Z}_2 = \langle H_2, h_2 \rangle_P$ if and only if

$$H_2 c_1 + |H_2 G_1| \, \mathbf{1} \leq h_2. \tag{3}$$

The second zonotope containment approach directly solves a linear feasibility problem, which allows us to efficiently incorporate containment constraints into a convex optimization problem (COP) [46]. According to [47], $\langle c_1, G_1 \rangle_Z \subseteq \langle c_2, G_2 \rangle_Z$ if there exist $\Gamma \in \mathbb{R}^{\eta(\mathcal{Z}_2) \times \eta(\mathcal{Z}_1)}$ and $\gamma \in \mathbb{R}^{\eta(\mathcal{Z}_2)}$ such that

$$G_1 = G_2 \Gamma \tag{4a}$$
$$c_2 - c_1 = G_2 \gamma \tag{4b}$$
$$\left| \begin{bmatrix} \Gamma & \gamma \end{bmatrix} \right| \mathbf{1} \leq \mathbf{1}. \tag{4c}$$

C. Problem Formulation

In this paper, we consider an unknown, discrete-time, time-invariant system that evolves according to

$$x_{k+1} = f\left(x_k, u_k, w_k\right), \tag{5}$$

where $x_k \in \mathbb{R}^{n_x}$ is the system state, $u_k \in \mathbb{R}^{n_u}$ is the control input, and $w_k \in \mathbb{R}^{n_w}$ is the disturbance at time $k\Delta t$ with time step $k \in \mathbb{N}_{\geq 0}$ and sampling period $\Delta t \in \mathbb{R}_{>0}$. The disturbance always lies within the unknown, bounded disturbance set $\mathcal{W} \subset \mathbb{R}^{n_w}$, i.e., $w_k \in \mathcal{W}$ for all k. We use $w_{(\cdot)} \in \mathcal{W}$ to obtain a more concise notation, i.e., we refer by $w_{(\cdot)}$ to the whole disturbance sequence and by w_k to the value of this sequence at time step k. We also use this concise notation for other sequences and their values at sampling times. Moreover, the unknown system in (5) is constrained by

$$x_{(\cdot)} \in \mathcal{X} \tag{6a}$$
$$u_{(\cdot)} \in \mathcal{U}, \tag{6b}$$

where $\mathcal{X} = \langle H_{\mathcal{X}}, h_{\mathcal{X}} \rangle_P \subset \mathbb{R}^{n_x}$ and $\mathcal{U} = \langle H_{\mathcal{U}}, h_{\mathcal{U}} \rangle_P \subset \mathbb{R}^{n_u}$ are the known state and input constraint sets, respectively, which contain the origin. Because $\mathcal{X}$ and $\mathcal{U}$ are typically represented by axis-aligned boxes, they can be easily expressed in halfspace, generator, and interval representation.

To gain some knowledge about the unknown system in (5), sufficiently exciting training data $\{x_k, u_k, x_{k+1}\}_{k=1}^N$ is available offline [36], [48], where $N \in \mathbb{N}_{>0}$ denotes the number of measurements. To deal with unstable systems, the training data is not required to be recorded from a single run of the system but can be obtained by performing multiple short experiments.

In this paper, the control goal is to determine the minimal modification $\|\tilde{u}_k - u_k\|_p$ of a desired input $\tilde{u}_k \in \mathbb{R}^{n_u}$ with $k \in \mathbb{N}_{\geq 0}$ such that the safety constraints in (6) are satisfied, where $\|\cdot\|_p$ represents any p-norm with $p \geq 1$. In the following section, we present our safe backup control approach that ensures the satisfaction of (6).

III. Safe Backup Control

In this section, we construct linear models that are conformant to the offline training data. In addition, we introduce our set-based safe backup controller and compute corresponding large safe sets so that (6) is satisfied. Throughout this section, we assume that the offline training data has captured all system behaviors. Because the system in (5) is unknown, we remove this unrealistic assumption in Section IV by updating our conformant model, safe backup controller, and safe set online.

A. Model Conformance

To provide formal safety guarantees, we first identify linear models that are conformant to the offline training data [34].

Definition 4 (Conformant Model): Let $\{x_k, u_k, x_{k+1}\}_{k=1}^N$ be a finite set of training data. Then, $\mathbf{M} = \left(\widehat{A}, \widehat{B}, \widehat{\mathcal{W}}\right)$ is a conformant model if for all $k \in \mathbb{N}_{[1,N]} = \{1, 2, \ldots, N\}$

$$x_{k+1} = \widehat{A} x_k + \widehat{B} u_k + \hat{w}_k \tag{7a}$$
$$\hat{w}_k \in \widehat{\mathcal{W}}, \tag{7b}$$

where $\widehat{A} \in \mathbb{R}^{n_x \times n_x}$, $\widehat{B} \in \mathbb{R}^{n_x \times n_u}$, and $\widehat{\mathcal{W}} \subset \mathbb{R}^{n_x}$ are the estimated system matrix, input matrix, and disturbance set, respectively.

To decrease the conservativeness of our safety filter, we want to find the conformant model $\mathbf{M} = \left(\widehat{A}, \widehat{B}, \widehat{\mathcal{W}}\right)$ whose estimated disturbance set $\widehat{\mathcal{W}}$ has the smallest volume. For simplicity, this is typically achieved in two steps [38], [49]: First, a standard system identification is performed to obtain $\widehat{A}$ and $\widehat{B}$ [33]. Second, an optimization problem is solved to minimize the volume of $\widehat{\mathcal{W}}$. Instead of this two-step approach, we propose to address both aspects simultaneously by solving

$$\underset{\mathbf{M}=(\widehat{A},\widehat{B},\widehat{\mathcal{W}})}{\text{minimize}} \quad \text{volume of } \widehat{\mathcal{W}} \tag{8a}$$
$$\text{subject to} \quad \mathbf{M} \text{ is a conformant model.} \tag{8b}$$

Hence, we use state-space representations in contrast to existing set membership identification methods [39], [50], which exploit autoregressive exogenous structures.

The volume of a general zonotope $\mathcal{Z} = \langle c, G \rangle_Z \subset \mathbb{R}^{n_x}$ can be computed exactly [51] or estimated using sampling-based techniques [52]. However, both approaches are computationally too expensive for large-scale systems, so we must use a suitable heuristic to cast (8) as an efficiently-solvable COP [46], [53]. For instance, suitable choices for the cost in (8a) are the Frobenius norm of G [54] or the 1-norm of G, which is defined as the maximum absolute column sum. Nevertheless, we can exactly solve (8) when restricting $\widehat{\mathcal{W}} = \langle c_{\widehat{\mathcal{W}}}, G_{\widehat{\mathcal{W}}} \rangle_Z$ to be a parallelotope with a symmetric positive definite generator matrix $G_{\widehat{\mathcal{W}}}$, as shown in the following proposition.

Proposition 1: Let $\{x_k, u_k, x_{k+1}\}_{k=1}^N$ be a finite set of training data. In addition, let $A^\star, B^\star, c^\star, G^\star$ be the solution

of the COP

$$\begin{aligned} \underset{A,B,c,G}{\text{minimize}} \quad & -\log(\det(G)) && \text{(9a)}\\ \text{subject to} \quad & G = G^T \succ \mathbf{0} && \text{(9b)}\\ & \text{for } k \in \mathbb{N}_{[1,N]}\text{:} \\ & \left|Gx_{k+1} - Ax_k - Bu_k - c\right| \leq \mathbf{1}, && \text{(9c)} \end{aligned}$$

where $\mathbf{0}$ denotes a matrix of zeros of appropriate dimensions. Then, $\mathbf{M}^\star = \left(\widehat{A}, \widehat{B}, \widehat{\mathcal{W}}\right)$ is the solution of (8), where $\widehat{A} = (G^\star)^{-1}A^\star$, $\widehat{B} = (G^\star)^{-1}B^\star$, and $\widehat{\mathcal{W}} = \left\langle (G^\star)^{-1}c^\star, (G^\star)^{-1}\right\rangle_Z$ is restricted to be a parallelotope with a symmetric positive definite generator matrix.

Proof: Model conformance constraint: Based on (2), the unique parameter vector $\lambda_k \in \mathbb{R}^{\eta(\widehat{\mathcal{W}})}$ for any $\hat{w}_k \in \widehat{\mathcal{W}} = \left\langle c_{\widehat{\mathcal{W}}}, G_{\widehat{\mathcal{W}}}\right\rangle_Z$ is given by $\lambda_k = G_{\widehat{\mathcal{W}}}^{-1}\left(\hat{w}_k - c_{\widehat{\mathcal{W}}}\right)$ with $|\lambda_k| \leq \mathbf{1}$. By additionally using (7a), we obtain the model conformance constraint

$$\left|G_{\widehat{\mathcal{W}}}^{-1}x_{k+1} - G_{\widehat{\mathcal{W}}}^{-1}\widehat{A}x_k - G_{\widehat{\mathcal{W}}}^{-1}\widehat{B}u_k - G_{\widehat{\mathcal{W}}}^{-1}c_{\widehat{\mathcal{W}}}\right| \leq \mathbf{1},$$

which is equivalent to (9c).

Cost function: Because $G_{\widehat{\mathcal{W}}}$ is symmetric positive definite, the volume of $\widehat{\mathcal{W}}$ is proportional to $\det(G_{\widehat{\mathcal{W}}})$ [51]. In addition, $\log(\det(M))$ equals $-\log\left(\det(M^{-1})\right)$ for any symmetric positive definite matrix $M \in \mathbb{R}^{n\times n}$, the inverse of a symmetric positive definite matrix is also symmetric positive definite, and $\det(M)$ is logarithmically concave on the set of symmetric positive definite matrices [46], [53]. Thus, the convex cost function in (9a) selects the conformant model whose estimated disturbance set has the smallest volume. ■

A matrix must be inverted when using Proposition 1. Numerical problems when computing the inverse of a matrix can be avoided by further restricting $\widehat{\mathcal{W}}$ to be an axis-aligned box, i.e., by restricting the corresponding generator matrix to be a diagonal matrix. In this case, (8) is a simple linear program [46], as shown in the following proposition.

Proposition 2: Let $\{x_k, u_k, x_{k+1}\}_{k=1}^N$ be a finite set of training data. If $A^\star, B^\star, c^\star, g^\star$ is the solution of the linear program

$$\begin{aligned} \underset{A,B,c,g}{\text{minimize}} \quad & \mathbf{1}^T g && \text{(10a)}\\ \text{subject to} \quad & \text{for } k \in \mathbb{N}_{[1,N]}\text{:} \\ & \left|x_{k+1} - Ax_k - Bu_k - c\right| \leq g, && \text{(10b)} \end{aligned}$$

then $\mathbf{M}^\star = \left(\widehat{A}, \widehat{B}, \widehat{\mathcal{W}}\right)$ is the solution of (8), where $\widehat{A} = A^\star$, $\widehat{B} = B^\star$, and $\widehat{\mathcal{W}} = \langle c^\star, \texttt{diag}(g^\star)\rangle_Z$ is restricted to be an axis-aligned box.

Proof: Model conformance constraint: For any $\hat{w}_k \in \widehat{\mathcal{W}} = \left\langle c_{\widehat{\mathcal{W}}}, G_{\widehat{\mathcal{W}}}\right\rangle_Z$, there exists a $\lambda_k \in \mathbb{R}^{\eta(\widehat{\mathcal{W}})}$ with $|\lambda_k| \leq \mathbf{1}$ such that $\hat{w}_k - c_{\widehat{\mathcal{W}}} = G_{\widehat{\mathcal{W}}}\lambda_k$. These conditions can be equivalently reformulated as $\left|\hat{w}_k - c_{\widehat{\mathcal{W}}}\right| \leq \texttt{diag}\left(\left|G_{\widehat{\mathcal{W}}}\right|\right)$ because $G_{\widehat{\mathcal{W}}}$ is a diagonal matrix and zonotopes are centrally symmetric sets. By additionally using (7a), the model conformance constraint in (10b) is obtained.

Cost function: The volume of $\left\langle c_{\widehat{\mathcal{W}}}, G_{\widehat{\mathcal{W}}}\right\rangle_Z$ equals the product of the elements of $\texttt{diag}\left(\left|G_{\widehat{\mathcal{W}}}\right|\right)$, which is a nonconvex function. Nevertheless, because the model conformance constraint in (10b) is linear, it can be equivalently separated into a single constraint for each of the n_x dimensions. Thus, no coupling exists between any of the n_x elements of $\texttt{diag}\left(\left|G_{\widehat{\mathcal{W}}}\right|\right)$. Therefore, minimizing the sum of any n_x convex functions whose single arguments are the elements of $\texttt{diag}\left(\left|G_{\widehat{\mathcal{W}}}\right|\right)$ also minimizes the product of these elements, resulting in the smallest volume of $\widehat{\mathcal{W}}$. We choose these n_x convex functions as identity maps to obtain a simple COP, resulting in the linear cost function in (10a). ■

By restricting $\widehat{\mathcal{W}}$ to be a parallelotope with a symmetric positive definite generator matrix or an axis-aligned box, we can exactly and efficiently solve the optimization problem in (8). However, using such restricted set representations might be too conservative for some applications. To overcome this potential issue, we propose another set membership identification approach that allows $\widehat{\mathcal{W}}$ to be a general zonotope and approximates the volume minimization of $\widehat{\mathcal{W}}$ by finding the minimum scaling factor $s_{\mathcal{X}}^\star \in \mathbb{R}_{\geq 0}$ such that $\widehat{\mathcal{W}} \subseteq s_{\mathcal{X}}^\star\mathcal{X}$. To cast this problem as a COP, we use the generator scaling framework [30], i.e., we fix the arbitrary orientations of the generators of $\widehat{\mathcal{W}}$ and optimize only their scaling factors, as shown in the following proposition.

Proposition 3: Let $\{x_k, u_k, x_{k+1}\}_{k=1}^N$ be a finite set of training data. In addition, let $A^\star, B^\star, c^\star, s_{\mathcal{X}}^\star, \lambda_1^\star, \lambda_2^\star, \ldots, \lambda_N^\star$ be the solution of the COP

$$\begin{aligned} \underset{\substack{A,B,c,s_{\mathcal{X}}\\ \lambda_1,\lambda_2,\ldots,\lambda_N}}{\text{minimize}} \quad & J_{\mathbf{M}}\left(s_{\mathcal{X}}, \lambda_1, \lambda_2, \ldots, \lambda_N\right) && \text{(11a)}\\ \text{subject to} \quad & \lambda_{\max} = \texttt{max}\left(\left|\begin{bmatrix}\lambda_1 & \lambda_2 & \ldots & \lambda_N\end{bmatrix}\right|\right) && \text{(11b)}\\ & 0 \leq s_{\mathcal{X}} && \text{(11c)}\\ & \left\langle c, G_{\text{fixed}}\texttt{diag}\left(\lambda_{\max}\right)\right\rangle_Z \subseteq s_{\mathcal{X}}\mathcal{X} && \text{(11d)}\\ & \text{for } k \in \mathbb{N}_{[1,N]}\text{:} \\ & x_{k+1} - Ax_k - Bu_k = c + G_{\text{fixed}}\lambda_k, && \text{(11e)} \end{aligned}$$

where $J_{\mathbf{M}}$ is a convex cost function, the function $\texttt{max}$ returns a vector containing the maximum value of each row, and $G_{\text{fixed}} \in \mathbb{R}^{n_x \times \eta(\widehat{\mathcal{W}})}$ is a user-defined fixed matrix. Then, $\mathbf{M}^\star = \left(\widehat{A}, \widehat{B}, \widehat{\mathcal{W}}\right)$ is a conformant model, where $\widehat{A} = A^\star$, $\widehat{B} = B^\star$, and $\widehat{\mathcal{W}} = \langle c^\star, G_{\text{fixed}}\texttt{diag}\left(\lambda_{\max}^\star\right)\rangle_Z$ with $\lambda_{\max}^\star = \texttt{max}\left(\left|\begin{bmatrix}\lambda_1^\star & \lambda_2^\star & \ldots & \lambda_N^\star\end{bmatrix}\right|\right)$. In addition, $\widehat{\mathcal{W}} \subseteq s_{\mathcal{X}}^\star\mathcal{X}$.

Proof: For any $\lambda_k \in \mathbb{R}^{\eta(\widehat{\mathcal{W}})}$, there exists a $\underline{\lambda}_k \in \mathbb{R}^{\eta(\widehat{\mathcal{W}})}$ with $|\underline{\lambda}_k| \leq \mathbf{1}$ such that $G_{\text{fixed}}\lambda_k = G_{\text{fixed}}\texttt{diag}(|\lambda_k|)\underline{\lambda}_k$. In addition, $\langle \bar{c}, G_{\text{fixed}}\texttt{diag}\left(|\lambda_k|\right)\rangle_Z \subseteq \left\langle \bar{c}, G_{\text{fixed}}\texttt{diag}\left(\left|\bar{\lambda}_k\right|\right)\right\rangle_Z$ for any $\bar{c} \in \mathbb{R}^{n_x}$ and $\bar{\lambda}_k \in \mathbb{R}^{\eta(\widehat{\mathcal{W}})}$ with $|\lambda_k| \leq \left|\bar{\lambda}_k\right|$. By using these relations, the fact that $|\lambda_k| \leq \lambda_{\max}$ for $k \in \mathbb{N}_{[1,N]}$ because of (11b), the model conformance constraint in (11e), and the system dynamics in (7a), it follows that the optimal $\mathbf{M}^\star$ is a conformant model. In addition, the constraints in (11c) and (11d) enforce $\widehat{\mathcal{W}} \subseteq s_{\mathcal{X}}^\star\mathcal{X}$. ■

It is straightforward to show that the optimal models obtained by solving (10) and (11) are identical when choosing $J_{\mathbf{M}} = \mathbf{1}^T\texttt{max}\left(\left|\begin{bmatrix}\lambda_1 & \lambda_2 & \ldots & \lambda_N\end{bmatrix}\right|\right)$ and $G_{\text{fixed}} = I$ in (11), where I denotes the identity matrix of appropriate dimensions. Thus, the COP in (11) offers more flexibility at the expense of an increased computational cost.

In summary, we can efficiently compute linear models that are conformant to the offline training data and have an

estimated zonotopic disturbance set of small volumes. Based on these optimal conformant models, we introduce our set-based safe backup controller and compute large safe sets in the following subsections.

B. Safe Backup Controller

Because the optimal conformant model $\mathbf{M}^\star = (\widehat{A}, \widehat{B}, \widehat{\mathcal{W}})$ is time-invariant, without loss of generality, the initial time is chosen to be zero throughout this section. When the initial set is $\mathcal{Z}_{x,0} = \langle c_{x,0}, G_{x,0} \rangle_Z \subseteq \mathcal{X}$, the initial state of the system $x_0 \in \mathcal{Z}_{x,0}$ can be expressed by

$$x_0 = c_{x,0} + G_{x,0}\lambda_{x,0} \tag{12a}$$
$$|\lambda_{x,0}| \leq \mathbf{1}, \tag{12b}$$

where $\lambda_{x,0} \in \mathbb{R}^{\eta(\mathcal{Z}_{x,0})}$ represents the not necessarily unique initial parameter vector.

To define a meaningful control problem, we assume that the tuple $(\widehat{A}, \widehat{B})$ is stabilizable for the remainder of this paper. If the assumption is violated, we could increase the number of measurements N, add more actuators, and follow more sophisticated experiment design approaches [33], [55]. Then, based on a stabilizing state feedback matrix $K \in \mathbb{R}^{n_u \times n_x}$, our set-based safe backup controller is

$$u_k = Kx_k + c_{u,k} + G_{u,k}\lambda_{x,0}, \tag{13}$$

where $\mathcal{Z}_{u,k} = \langle c_{u,k}, G_{u,k} \rangle_Z$ with generator matrix $G_{u,k} \in \mathbb{R}^{n_u \times \eta(\mathcal{Z}_{x,0})}$ is the correction input zonotope at time step k, which will be optimized in Subsection III-C. Thus, in addition to the zonotopic parameterized control used in [30], our controller in (13) also consists of a stabilizing state feedback component to improve the control performance [56].

To ensure safety at all times, we must guarantee that the states and inputs satisfy the constraints in (6) despite the unknown, bounded disturbances. Thus, we perform reachability analysis, i.e., we compute the sets of states and inputs in (7a) and (13) that are reachable for all $x_0 \in \mathcal{Z}_{x,0}$ and $\hat{w}_{(\cdot)} \in \widehat{\mathcal{W}}$.

Before we compute these reachable sets, we introduce the combined state and input $\begin{bmatrix} x^T & u^T \end{bmatrix}^T$. To project a zonotope of combined states and inputs $\mathcal{Z} \subset \mathbb{R}^{n_x+n_u}$ onto the original state and input space [57], we define the two matrices

$$\Pi_x = \begin{bmatrix} I & \mathbf{0} \end{bmatrix} \in \mathbb{R}^{n_x \times (n_x+n_u)} \tag{14a}$$
$$\Pi_u = \begin{bmatrix} \mathbf{0} & I \end{bmatrix} \in \mathbb{R}^{n_u \times (n_x+n_u)}. \tag{14b}$$

For instance, the projection of $\mathcal{Z}$ onto the original state space is computed by $\Pi_x \mathcal{Z}$. In addition, we introduce the recursively defined set sequence

$$\begin{aligned} &\mathcal{R}(k, \mathcal{Z}_{x,0}, \mathcal{Z}_{u,(\cdot)}, \widehat{\mathcal{W}}) \\ &\quad = \left\langle c_{\mathcal{R}(k,\mathcal{Z}_{x,0},\mathcal{Z}_{u,(\cdot)},\widehat{\mathcal{W}})}, G_{\mathcal{R}(k,\mathcal{Z}_{x,0},\mathcal{Z}_{u,(\cdot)},\widehat{\mathcal{W}})} \right\rangle_Z \\ &\quad = \left\langle \begin{bmatrix} c_{x,k} \\ Kc_{x,k} + c_{u,k} \end{bmatrix}, \begin{bmatrix} G_{x,k} \\ KG_{x,k} + \begin{bmatrix} G_{u,k} & \mathbf{0} \end{bmatrix} \end{bmatrix} \right\rangle_Z, \end{aligned} \tag{15a}$$

$$\begin{aligned} &\langle c_{x,k+1}, G_{x,k+1} \rangle_Z \\ &\quad = \begin{bmatrix} \widehat{A} & \widehat{B} \end{bmatrix} \mathcal{R}(k, \mathcal{Z}_{x,0}, \mathcal{Z}_{u,(\cdot)}, \widehat{\mathcal{W}}) \oplus \widehat{\mathcal{W}}, \end{aligned} \tag{15b}$$

where $k \in \mathbb{N}_{\geq 0}$ and $\mathcal{Z}_{u,(\cdot)} = \langle c_{u,(\cdot)}, G_{u,(\cdot)} \rangle_Z$ is the correction input zonotope sequence in (13). In the following proposition, we prove that the sets in (15) are the reachable sets of (7a) and (13) for all $x_0 \in \mathcal{Z}_{x,0}$ and $\hat{w}_{(\cdot)} \in \widehat{\mathcal{W}} = \langle c_{\widehat{\mathcal{W}}}, G_{\widehat{\mathcal{W}}} \rangle_Z$,

Proposition 4: For all $x_0 \in \mathcal{Z}_{x,0}$ and $\hat{w}_{(\cdot)} \in \widehat{\mathcal{W}}$, applying the safe backup control input in (13) to the system in (7a) results in

$$\begin{bmatrix} x_k \\ u_k \end{bmatrix} \in \mathcal{R}(k, \mathcal{Z}_{x,0}, \mathcal{Z}_{u,(\cdot)}, \widehat{\mathcal{W}}),$$

where $k \in \mathbb{N}_{\geq 0}$.

Proof: We prove that applying the safe backup control input in (13) to the system in (7a) results in

$$\begin{bmatrix} x_k \\ u_k \end{bmatrix} = c_{\mathcal{R}(k,\mathcal{Z}_{x,0},\mathcal{Z}_{u,(\cdot)},\widehat{\mathcal{W}})} + G_{\mathcal{R}(k,\mathcal{Z}_{x,0},\mathcal{Z}_{u,(\cdot)},\widehat{\mathcal{W}})} \begin{bmatrix} \lambda_{x,0} \\ \lambda_{\widehat{\mathcal{W}},k} \end{bmatrix}, \tag{16}$$

where $\lambda_{x,0} \in \mathbb{R}^{\eta(\mathcal{Z}_{x,0})}$ satisfies (12) and $\lambda_{\widehat{\mathcal{W}},k} \in \mathbb{R}^{k\eta(\widehat{\mathcal{W}})}$ satisfies

$$\lambda_{\widehat{\mathcal{W}},k} = \begin{bmatrix} \lambda_{\hat{w}_0}^T & \lambda_{\hat{w}_1}^T & \dots & \lambda_{\hat{w}_{k-1}}^T \end{bmatrix}^T \tag{17a}$$
$$\left|\lambda_{\widehat{\mathcal{W}},k}\right| \leq \mathbf{1} \tag{17b}$$
$$\hat{w}_k = c_{\widehat{\mathcal{W}}} + G_{\widehat{\mathcal{W}}}\lambda_{\hat{w}_k}. \tag{17c}$$

We proceed by induction:

Base case: For $k = 0$, we obtain

$$\begin{aligned} \begin{bmatrix} x_0 \\ u_0 \end{bmatrix} &\overset{(12),(13)}{=} \begin{bmatrix} c_{x,0} + G_{x,0}\lambda_{x,0} \\ K(c_{x,0} + G_{x,0}\lambda_{x,0}) + c_{u,0} + G_{u,0}\lambda_{x,0} \end{bmatrix} \\ &\overset{(15a)}{=} c_{\mathcal{R}(0,\mathcal{Z}_{x,0},\mathcal{Z}_{u,(\cdot)},\widehat{\mathcal{W}})} + G_{\mathcal{R}(0,\mathcal{Z}_{x,0},\mathcal{Z}_{u,(\cdot)},\widehat{\mathcal{W}})} \begin{bmatrix} \lambda_{x,0} \\ \lambda_{\widehat{\mathcal{W}},0} \end{bmatrix}. \end{aligned}$$

Induction hypothesis: (16) and (17) hold for some $k \in \mathbb{N}_{\geq 0}$.

Induction step: For the state at $k+1$, we obtain

$$\begin{aligned} x_{k+1} &\overset{(7a),(17c)}{=} \begin{bmatrix} \widehat{A} & \widehat{B} \end{bmatrix} \begin{bmatrix} x_k \\ u_k \end{bmatrix} + c_{\widehat{\mathcal{W}}} + G_{\widehat{\mathcal{W}}}\lambda_{\hat{w}_k} \\ &\overset{(16),(17a)}{=} \begin{bmatrix} \widehat{A} & \widehat{B} \end{bmatrix} c_{\mathcal{R}(k,\mathcal{Z}_{x,0},\mathcal{Z}_{u,(\cdot)},\widehat{\mathcal{W}})} + c_{\widehat{\mathcal{W}}} \\ &\quad + \left[\begin{bmatrix} \widehat{A} & \widehat{B} \end{bmatrix} G_{\mathcal{R}(k,\mathcal{Z}_{x,0},\mathcal{Z}_{u,(\cdot)},\widehat{\mathcal{W}})} \quad G_{\widehat{\mathcal{W}}} \right] \begin{bmatrix} \lambda_{x,0} \\ \lambda_{\widehat{\mathcal{W}},k+1} \end{bmatrix} \\ &\overset{(1),(14a),(15b),(15a)}{=} \Pi_x c_{\mathcal{R}(k+1,\mathcal{Z}_{x,0},\mathcal{Z}_{u,(\cdot)},\widehat{\mathcal{W}})} \\ &\quad + \Pi_x G_{\mathcal{R}(k+1,\mathcal{Z}_{x,0},\mathcal{Z}_{u,(\cdot)},\widehat{\mathcal{W}})} \begin{bmatrix} \lambda_{x,0} \\ \lambda_{\widehat{\mathcal{W}},k+1} \end{bmatrix}, \end{aligned} \tag{18}$$

where a not necessarily unique $\lambda_{\hat{w}_k} \in \mathbb{R}^{\eta(\widehat{\mathcal{W}})}$ with $|\lambda_{\hat{w}_k}| \leq \mathbf{1}$ is guaranteed to exist because $\hat{w}_k \in \widehat{\mathcal{W}}$. Similarly, for the input at $k+1$, we obtain

$$\begin{aligned} u_{k+1} &\overset{(13)}{=} Kx_{k+1} + c_{u,k+1} + G_{u,k+1}\lambda_{x,0} \\ &\overset{(18)}{=} K\Pi_x c_{\mathcal{R}(k+1,\mathcal{Z}_{x,0},\mathcal{Z}_{u,(\cdot)},\widehat{\mathcal{W}})} + c_{u,k+1} \\ &\quad + K\Pi_x G_{\mathcal{R}(k+1,\mathcal{Z}_{x,0},\mathcal{Z}_{u,(\cdot)},\widehat{\mathcal{W}})} \begin{bmatrix} \lambda_{x,0} \\ \lambda_{\widehat{\mathcal{W}},k+1} \end{bmatrix} \\ &\quad + \begin{bmatrix} G_{u,k+1} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \lambda_{x,0} \\ \lambda_{\widehat{\mathcal{W}},k+1} \end{bmatrix} \\ &\overset{(1),(14b),(15b),(15a)}{=} \Pi_u c_{\mathcal{R}(k+1,\mathcal{Z}_{x,0},\mathcal{Z}_{u,(\cdot)},\widehat{\mathcal{W}})} \\ &\quad + \Pi_u G_{\mathcal{R}(k+1,\mathcal{Z}_{x,0},\mathcal{Z}_{u,(\cdot)},\widehat{\mathcal{W}})} \begin{bmatrix} \lambda_{x,0} \\ \lambda_{\widehat{\mathcal{W}},k+1} \end{bmatrix}, \end{aligned}$$

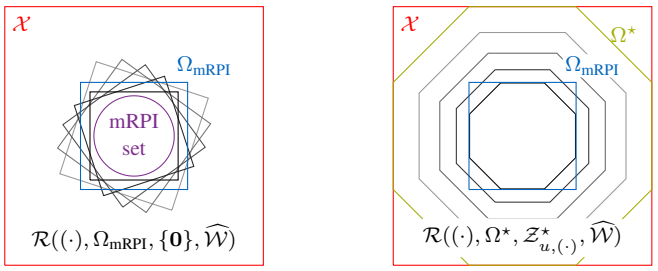


Fig. 1. Two-step safe set approach. The small safe set $\boldsymbol{\Omega}_{\text{mRPI}}$ can be safely steered into itself (a), and the optimal large safe set $\boldsymbol{\Omega}^\star$ can be safely steered into $\boldsymbol{\Omega}_{\text{mRPI}}$ (b). Reachable sets are shown, with a lighter gray tone corresponding to a smaller prediction horizon.

which completes the proof of (16). Finally, the result follows based on Definition 2, (12), and (17). ∎

In summary, we can efficiently compute the sets of states and inputs in (7a) and (13) that are reachable for all $x_0 \in \mathcal{Z}_{x,0}$ and $\hat{w}_{(\cdot)} \in \widehat{\mathcal{W}}$. We use these reachable set computations in the following subsection to construct large safe sets.

C. Safe Sets

For the remainder of this paper, we assume that the estimated disturbance set $\widehat{\mathcal{W}}$ of $\mathbf{M}^\star = (\widehat{A}, \widehat{B}, \widehat{\mathcal{W}})$ contains the origin. This assumption can easily be satisfied by adding the constraint $\mathbf{0} \in \widehat{\mathcal{W}}$ to the optimization problems in Subsection III-A or by performing a suitable coordinate transformation. Based on this standard assumption, we define the minimal robust positively invariant (mRPI) set of the system in (7a) when using the state feedback controller $u_k = Kx_k$ for $k \in \mathbb{N}_{\geq 0}$, which is identical to the controller in (13) when choosing $\langle c_{u,k}, G_{u,k} \rangle_Z = \{\mathbf{0}\}$.

Definition 5 (mRPI Set): The minimal robust positively invariant (mRPI) set of the system in (7a) with $u_k = Kx_k$, $k \in \mathbb{N}_{\geq 0}$, and $\hat{w}_{(\cdot)} \in \widehat{\mathcal{W}}$ is $\Pi_x \mathcal{R}(\infty, \{\mathbf{0}\}, \{\mathbf{0}\}, \widehat{\mathcal{W}})$ [24], [58].

The mRPI set has the valuable property that it is the limit set for all state sequences [59, Rmk. 4.1]. Thus, it equals $\Pi_x \mathcal{R}(\infty, \mathcal{Z}, \{\mathbf{0}\}, \widehat{\mathcal{W}})$, where $\mathcal{Z} \subset \mathbb{R}^{n_x}$ is any closed, bounded set. In addition to this important invariant set, we define safe sets, which are not necessarily RCI sets [24], [25].

Definition 6 (Safe Set): A set $\Omega \subseteq \mathcal{X}$ is a safe set of the system in (7a) if a controller exists such that the safety constraints in (6) are satisfied for all $x_0 \in \Omega$ and $\hat{w}_{(\cdot)} \in \widehat{\mathcal{W}}$.

We want to find large safe sets to increase the region of operation of our safety filter. To achieve this in a scalable way, our safe set approach proceeds in two steps: First, we compute a small safe set Ω_{mRPI} using the simple controller $u_k = Kx_k$. Although Ω_{mRPI} is not necessarily invariant, it can be safely steered into itself in finite time, i.e., the states and applied inputs lie within $\mathcal{X}$ and $\mathcal{U}$ during this time, as illustrated in Fig. 1a. Second, we solve a COP whose solution is the large safe set $\Omega^\star$ that can be safely steered into Ω_{mRPI} in finite time, as illustrated in Fig. 1b. Subsequently, we describe this two-step safe set approach in more detail.

For our first step, we require the existence of a small safe set $\Omega_{\text{mRPI}} \subseteq \mathcal{X}$ with $k_\circ \in \mathbb{N}_{\geq 0}$ satisfying

$$\Omega_{\text{mRPI}} = \Pi_x \mathcal{R}(k_\circ, \mathcal{X}, \{\mathbf{0}\}, \widehat{\mathcal{W}}) \tag{19a}$$

$$\text{for } k \in \mathbb{N}_{[0,k_\circ-1]}: \quad \mathcal{R}(k, \Omega_{\text{mRPI}}, \{\mathbf{0}\}, \widehat{\mathcal{W}}) \subseteq (\mathcal{X} \times \mathcal{U}). \tag{19b}$$

The existence of Ω_{mRPI} is also a widely used assumption in robust MPC [18]–[20]. If such an Ω_{mRPI} does not exist, the mRPI set $\mathcal{S} \subset \mathbb{R}^{n_x}$ has the following property: $(\mathcal{S} \times K\mathcal{S}) \not\subseteq (\mathcal{X} \times \mathcal{U})$ or $\mathcal{S} \times K\mathcal{S}$ touches the bounds of $\mathcal{X} \times \mathcal{U}$. In this case, a different stabilizing state feedback matrix K is required to satisfy (19). In the following lemma, we show essential properties of an Ω_{mRPI} that fulfills (19).

Lemma 1: Let $\Omega_{\text{mRPI}} \subseteq \mathcal{X}$ with $k_\circ \in \mathbb{N}_{\geq 0}$ satisfy (19). Then, Ω_{mRPI} with corresponding controller $u_k = Kx_k$ is a safe set, i.e., applying these control inputs to the system in (7a) ensures the satisfaction of (6) for all $x_0 \in \Omega_{\text{mRPI}}$ and $\hat{w}_{(\cdot)} \in \widehat{\mathcal{W}}$. In addition, Ω_{mRPI} is an over-approximation of the mRPI set.

Proof: Safe set: By

$$\Pi_x \mathcal{R}(k_\circ, \Omega_{\text{mRPI}}, \{\mathbf{0}\}, \widehat{\mathcal{W}}) \overset{(19b)}{\subseteq} \Pi_x \mathcal{R}(k_\circ, \mathcal{X}, \{\mathbf{0}\}, \widehat{\mathcal{W}}) \overset{(19a)}{=} \Omega_{\text{mRPI}},$$

we show that Ω_{mRPI} can be steered into itself in $k_\circ$ steps, which is also known as $k_\circ$-step recurrent [60]. In contrast to invariant sets, the state of the system might leave Ω_{mRPI} for $k \in \mathbb{N}_{[1,k_\circ-1]}$. Nevertheless, during this time, the states and applied inputs lie within $\mathcal{X}$ and $\mathcal{U}$ because of (19b). Thus, it follows by induction that the constraints in (6) are satisfied for all $x_0 \in \Omega_{\text{mRPI}}$ and $\hat{w}_{(\cdot)} \in \widehat{\mathcal{W}}$ when applying the control inputs $u_k = Kx_k$ to the system in (7a). Therefore, Ω_{mRPI} is a safe set.

Over-approximation of mRPI set: Subsequently, we show by contradiction that Ω_{mRPI} is an over-approximation of the mRPI set $\mathcal{S} \subset \mathbb{R}^{n_x}$, i.e., we assume that $\mathcal{S} \not\subseteq \Omega_{\text{mRPI}}$. Because $\mathbf{0} \in \widehat{\mathcal{W}}$ and $\mathbf{0} \in \mathcal{X}$ by assumption, we know that $\mathbf{0} \in \mathcal{S}$ and $\mathbf{0} \in \Omega_{\text{mRPI}}$. In addition, $\mathcal{S}$ is the limit set for all state sequences of the system in (7a) when using the controller $u_k = Kx_k$ [59, Rmk. 4.1]. Therefore, there exists a disturbance sequence $\hat{w}_{(\cdot)}$ that steers the state sequence of (7a) starting at $\mathbf{0}$ to any point in $\mathcal{S}$ and remains at this point. Because $\mathcal{S} \not\subseteq \Omega_{\text{mRPI}}$, there exists a state sequence starting in Ω_{mRPI} that leaves Ω_{mRPI} and never returns to Ω_{mRPI}, which contradicts Ω_{mRPI} being a safe set. As a result, the assumption $\mathcal{S} \not\subseteq \Omega_{\text{mRPI}}$ is wrong, which shows that Ω_{mRPI} is an over-approximation of the mRPI set. ∎

Typically, when Ω_{mRPI} satisfies (19), it is a small safe set. To obtain a large safe set, which increases the region of operation of our safety filter, we solve a COP in the second step of our safe set approach. Let $s^\star_\Omega$, $c^\star_\Omega$, $\mathcal{Z}^\star_{u,(\cdot)}$ be the solution of the

COP

$$\begin{aligned}
\underset{s_\Omega, c_\Omega, \mathcal{Z}_{u,(\cdot)}}{\text{maximize}} \quad & J_\Omega(s_\Omega) && \text{(20a)}\\
\text{subject to} \quad & \Omega = \langle c_\Omega, G_{\text{fixed}}\texttt{diag}(s_\Omega)\rangle_Z && \text{(20b)}\\
& \Pi_x\mathcal{R}(k_{\text{mRPI}}, \Omega, \mathcal{Z}_{u,(\cdot)}, \{\mathbf{0}\}) = \{\mathbf{0}\} && \text{(20c)}\\
& \text{for } k \in \mathbb{N}_{[0,k_{\text{mRPI}}-1]}\text{:} &&\\
& \mathcal{R}(k, \Omega, \mathcal{Z}_{u,(\cdot)}, \widehat{\mathcal{W}}) \subseteq (\mathcal{X} \times \mathcal{U}), && \text{(20d)}
\end{aligned}$$

where J_Ω is a concave cost function, $s_\Omega \in \mathbb{R}_{\geq 0}^{\eta(\Omega)}$ is a generator scaling vector, $G_{\text{fixed}} \in \mathbb{R}^{n_x \times \eta(\Omega)}$ is a fixed generator matrix, and $k_{\text{mRPI}} \in \mathbb{N}_{>0}$ is the time step when Ω_{mRPI} is reached. Then, the optimal large safe set is $\Omega^\star = \langle c_\Omega^\star, G_{\text{fixed}}\texttt{diag}(s_\Omega^\star)\rangle_Z$.

Lemma 2: Let $\Omega_{\text{mRPI}} \subseteq \mathcal{X}$ with $k_\circ \in \mathbb{N}_{\geq 0}$ satisfy (19) and let $s_\Omega^\star$, $c_\Omega^\star$, $\mathcal{Z}_{u,(\cdot)}^\star$ be the solution of (20) for any $k_{\text{mRPI}} \in \mathbb{N}_{>0}$. Then, the controller in (13) with optimal correction input zonotope sequence $\mathcal{Z}_{u,(\cdot)}^\star$ safely steers the system in (7a) starting from $\Omega^\star = \langle c_\Omega^\star, G_{\text{fixed}}\texttt{diag}(s_\Omega^\star)\rangle_Z$ into Ω_{mRPI} in k_{mRPI} steps, i.e., $x_k \in \mathcal{X}$, $u_k \in \mathcal{U}$, and $x_{k_{\text{mRPI}}} \in \Omega_{\text{mRPI}}$ for all $x_0 \in \Omega^\star$, $\hat{w}_k \in \widehat{\mathcal{W}}$, and $k \in \mathbb{N}_{[0,k_{\text{mRPI}}-1]}$.

Proof: We show that all $x_0 \in \Omega^\star$ can be steered into Ω_{mRPI} in k_{mRPI} steps by

$$\begin{aligned}
&\Pi_x\mathcal{R}(k_{\text{mRPI}}, \Omega^\star, \mathcal{Z}_{u,(\cdot)}^\star, \widehat{\mathcal{W}})\\
&\quad \overset{(1),(14a)}{=} \Pi_x\mathcal{R}(k_{\text{mRPI}}, \Omega^\star, \mathcal{Z}_{u,(\cdot)}^\star, \{\mathbf{0}\})\\
&\qquad \oplus \Pi_x\mathcal{R}(k_{\text{mRPI}}, \{\mathbf{0}\}, \{\mathbf{0}\}, \widehat{\mathcal{W}})\\
&\quad \overset{(20c)}{=} \Pi_x\mathcal{R}(k_{\text{mRPI}}, \{\mathbf{0}\}, \{\mathbf{0}\}, \widehat{\mathcal{W}})\\
&\quad \overset{[58]}{\subseteq} \Pi_x\mathcal{R}(\infty, \{\mathbf{0}\}, \{\mathbf{0}\}, \widehat{\mathcal{W}})\\
&\quad \overset{\text{Lemma 1}}{\subseteq} \Omega_{\text{mRPI}},
\end{aligned}$$

where the superposition principle is exploited in the first step. In addition, the constraint in (20d) ensures $x_k \in \mathcal{X}$ and $u_k \in \mathcal{U}$ for all $k \in \mathbb{N}_{[0,k_{\text{mRPI}}-1]}$. Thus, the controller in (13) with optimal correction input zonotope sequence $\mathcal{Z}_{u,(\cdot)}^\star$ steers all $x_0 \in \Omega^\star$ safely into Ω_{mRPI} in k_{mRPI} steps. ∎

In contrast to the COP solved in our previous work [31], where the terminal constraint $\Pi_x\mathcal{R}(k_{\text{mRPI}}, \Omega, \mathcal{Z}_{u,(\cdot)}, \widehat{\mathcal{W}}) \subseteq \Omega_{\text{mRPI}}$ is used instead of (20c), the number of constraints in (20) is independent of Ω_{mRPI}. This slight modification is especially advantageous if the parameter k_{mRPI} is large, resulting in many additional optimization variables and constraints based on (4).

To efficiently solve the COP in (20), we subsequently recommend how to choose the involved parameters. To cover $\mathcal{X}$, we can uniformly sample from the unit hypersphere and use the obtained points as columns of the user-defined fixed generator matrix G_{fixed}. Because uniform sampling in high-dimensional spaces is a complex task, it is beneficial to examine the sparsity of the system matrix [30]. Alternatively, a viable choice for G_{fixed} can be the generator matrix of Ω_{mRPI} because it already incorporates some effects of $\widehat{\mathcal{W}}$. Ideally, we want to maximize the volume of Ω in (20a). Thus, using the sum or the geometric mean of the input vector elements for the cost function J_Ω in (20a) are reasonable heuristics. For instance, the geometric mean is a monotonic function of the volume of Ω if $G_{\text{fixed}} = I$. As shown in Lemma 2, the parameter k_{mRPI} corresponds to the time step when all states starting in Ω reach Ω_{mRPI}. Thus, this parameter is used to balance accuracy and computational complexity, e.g., usually $\Omega_{\text{mRPI}} \not\subseteq \Omega^\star$ when choosing $k_{\text{mRPI}} = 1$, as shown in the following theorem.

Theorem 1: Let $\Omega_{\text{mRPI}} \subseteq \mathcal{X}$ with $k_\circ \in \mathbb{N}_{\geq 0}$ satisfy (19). Then, the COP in (20) is always feasible, and the cost in (20a) is monotonically increasing with increasing $k_{\text{mRPI}} \in \mathbb{N}_{>0}$.

Proof: Feasibility: When choosing $s_\Omega = \mathbf{0}$, $c_\Omega = \mathbf{0}$, and $\mathcal{Z}_{u,(\cdot)} = \{\mathbf{0}\}$, we always obtain $\Omega = \{\mathbf{0}\}$ in (20b). Because $\Pi_x\mathcal{R}(k, \{\mathbf{0}\}, \{\mathbf{0}\}, \{\mathbf{0}\}) = \{\mathbf{0}\}$ for any $k \in \mathbb{N}_{\geq 0}$, the constraint in (20c) is satisfied for any $k_{\text{mRPI}} \in \mathbb{N}_{>0}$. In addition, the satisfaction of (20d) for any k_{mRPI} follows from $\mathbf{0} \in \Omega_{\text{mRPI}}$ and Lemma 1. Thus, the COP in (20) is always feasible.

Monotonically increasing cost: Let $s_\Omega^\star$, $c_\Omega^\star$, $\mathcal{Z}_{u,(\cdot)}^\star$ be the solution of (20) for any k_{mRPI}. Subsequently, we show that $s_\Omega^{\star,+}$, $c_\Omega^{\star,+}$, $\mathcal{Z}_{u,(\cdot)}^{\star,+}$ is feasible for $k_{\text{mRPI}}+1$, where $s_\Omega^{\star,+} = s_\Omega^\star$, $c_\Omega^{\star,+} = c_\Omega^\star$, and $\mathcal{Z}_{u,(\cdot)}^{\star,+}$ is obtained by appending $\{\mathbf{0}\}$ to $\mathcal{Z}_{u,(\cdot)}^\star$. By reusing the previous solution, the cost in (20a) of both COPs is the same. Thus, when optimizing over all feasible s_Ω, c_Ω, $\mathcal{Z}_{u,(\cdot)}$, the cost in (20a) for $k_{\text{mRPI}}+1$ is at least as high as for k_{mRPI}. By reusing $s_\Omega^\star$ and $c_\Omega^\star$, the same $\Omega^\star = \langle c_\Omega^\star, G_{\text{fixed}}\texttt{diag}(s_\Omega^\star)\rangle_Z$ is obtained in (20b) for both k_{mRPI} and $k_{\text{mRPI}}+1$. Because $\mathcal{Z}_{u,k_{\text{mRPI}}}^{\star,+} = \{\mathbf{0}\}$ and $\Pi_x\mathcal{R}(1, \{\mathbf{0}\}, \{\mathbf{0}\}, \{\mathbf{0}\}) = \{\mathbf{0}\}$, the constraint in (20c) is also satisfied. Fulfillment of (20d) for $k = k_{\text{mRPI}}$ is proven by

$$\begin{aligned}
&\mathcal{R}(k_{\text{mRPI}}, \Omega^\star, \mathcal{Z}_{u,(\cdot)}^{\star,+}, \widehat{\mathcal{W}})\\
&\quad = \mathcal{R}(k_{\text{mRPI}}, \Omega^\star, \mathcal{Z}_{u,(\cdot)}^{\star,+}, \{\mathbf{0}\}) \oplus \mathcal{R}(k_{\text{mRPI}}, \{\mathbf{0}\}, \{\mathbf{0}\}, \widehat{\mathcal{W}})\\
&\quad \overset{(14),\,(20c),\,\mathcal{Z}_{u,k_{\text{mRPI}}}^{\star,+}=\{\mathbf{0}\}}{\subseteq} \Pi_x\mathcal{R}(k_{\text{mRPI}}, \{\mathbf{0}\}, \{\mathbf{0}\}, \widehat{\mathcal{W}})\\
&\qquad \times \Pi_u\mathcal{R}(k_{\text{mRPI}}, \{\mathbf{0}\}, \{\mathbf{0}\}, \widehat{\mathcal{W}})\\
&\quad \overset{\text{Lemma 1, [58]}}{\subseteq} \Omega_{\text{mRPI}} \times K\Omega_{\text{mRPI}}\\
&\quad \overset{(19)}{\subseteq} \mathcal{X} \times \mathcal{U}
\end{aligned}$$

where the superposition principle is exploited in the first step. Thus, $s_\Omega^\star$, $c_\Omega^\star$, $\mathcal{Z}_{u,(\cdot)}^{\star,+}$ is actually feasible for $k_{\text{mRPI}}+1$. ∎

Finally, by combining the presented two steps of our safe set approach, we can prove satisfaction of (6) for the system in (7a) and all $x_0 \in \Omega^\star$ and $\hat{w}_{(\cdot)} \in \widehat{\mathcal{W}}$. To show this, we first define the resulting correction input zonotope sequence

$$\begin{aligned}
\mathcal{Z}_{u,k}^{\Omega^\star} &= \left\langle c_{u,k}^{\Omega^\star}, G_{u,k}^{\Omega^\star} \right\rangle_Z\\
&= \begin{cases} \mathcal{Z}_{u,k}^\star & \text{for } k \in \mathbb{N}_{[0,k_{\text{mRPI}}-1]} \\ \{\mathbf{0}\} & \text{otherwise} \end{cases}, \qquad \text{(21)}
\end{aligned}$$

where $s_\Omega^\star$, $c_\Omega^\star$, $\mathcal{Z}_{u,(\cdot)}^\star$ is the solution of (20).

Proposition 5: Let $\Omega_{\text{mRPI}} \subseteq \mathcal{X}$ with $k_\circ \in \mathbb{N}_{\geq 0}$ satisfy (19) and let $\Omega^\star$ be the optimal safe set obtained by solving (20) for any $k_{\text{mRPI}} \in \mathbb{N}_{>0}$. Then, applying the control inputs in (13) with correction input zonotope $\mathcal{Z}_{u,k}^{\Omega^\star}$ to the system in (7a) ensures the satisfaction of the constraints in (6) for all $x_0 \in \Omega^\star$ and $\hat{w}_{(\cdot)} \in \widehat{\mathcal{W}}$.

Proof: For $k \in \mathbb{N}_{[0,k_{\text{mRPI}}-1]}$, Lemma 2 ensures $x_k \in \mathcal{X}$, $u_k \in \mathcal{U}$, and $x_{k_{\text{mRPI}}} \in \Omega_{\text{mRPI}} \subseteq \mathcal{X}$ for all $x_0 \in \Omega^\star$ and $\hat{w}_k \in \widehat{\mathcal{W}}$. For all $k \geq k_{\text{mRPI}}$, Lemma 1 guarantees $x_k \in \mathcal{X}$ and $u_k \in \mathcal{U}$ when using the controller $u_k = Kx_k$, which is identical to the controller in (13) when $\langle c_{u,k}, G_{u,k} \rangle_Z = \{\mathbf{0}\}$. Thus, the satisfaction of the safety constraints in (6) is ensured for all $x_0 \in \Omega^\star$ and $\hat{w}_{(\cdot)} \in \widehat{\mathcal{W}}$. ∎

In summary, we can efficiently compute safe sets with corresponding safe backup controllers using scalable reachability analysis and convex optimization algorithms. In the following section, we incorporate these safe sets and controllers into our safety filter.

IV. Safety Filter

We want to avoid having big spikes when switching between a desired control input and a safe backup control input [61]. Thus, our safety filter aims to minimize modifying a desired control input so that the satisfaction of the safety constraints in (6) is ensured. We achieve this goal by always enforcing the state of the unknown system in (5) to stay within the optimal safe set $\Omega^\star$. Because the desired control inputs are only known during runtime, we solve an optimal control problem online, which takes a nonnegligible amount of time [20], [62]. In this section, we explicitly consider such computational delays instead of assuming that optimization problems can be solved instantaneously. In addition, we present our safety filter algorithm. Finally, we propose our online conformance update to restore formal safety guarantees as soon as we detect that $\hat{w}_k \notin \widehat{\mathcal{W}}$ with $k \in \mathbb{N}_{\geq 0}$ for the optimal conformant model $\mathbf{M}^\star = \left(\widehat{A}, \widehat{B}, \widehat{\mathcal{W}}\right)$.

A. Consideration of Computation Times

If $x_0 \in \Omega^\star$, Proposition 5 guarantees robust constraint satisfaction for the system in (7a) and all $\hat{w}_{(\cdot)} \in \widehat{\mathcal{W}}$. To compute the safe backup control input in (13) for the initial set $\mathcal{Z}_{x,0} = \Omega^\star$, we must find the not necessarily unique initial parameter vector $\lambda_{x,0} \in \mathbb{R}^{\eta(\Omega^\star)}$ satisfying (12).

Obtaining $\lambda_{x,0}$ can be achieved by solving the linear feasibility problem in (12), which causes a nonnegligible computational delay that invalidates the formal safety guarantees [20], [62]. Alternatively, if the extreme points of $\Omega^\star$ are given, closed-form expressions of $\lambda_{x,0}$ exist [63]. However, obtaining the extreme points of a general zonotope is a computationally complex task [43]. To ensure the scalability of our approach, we present an efficient method for computing $\lambda_{x,0}$ without assuming that an optimization problem can be solved instantaneously at time step 0, as shown in the following lemma.

Lemma 3: Let a parallelotope $\mathcal{P} = \langle c_1, G_1 \rangle_Z \subset \mathbb{R}^n$ and a zonotope $\mathcal{Z} = \langle c_2, G_2 \rangle_Z \subset \mathbb{R}^n$ be given. In addition, let $\Gamma \in \mathbb{R}^{\eta(\mathcal{Z}) \times n}$ and $\gamma \in \mathbb{R}^{\eta(\mathcal{Z})}$ exist such that (4) is satisfied. Then, a valid parameter vector $\lambda_\mathcal{Z} \in \mathbb{R}^{\eta(\mathcal{Z})}$ of $\mathcal{Z}$ with $|\lambda_\mathcal{Z}| \leq \mathbf{1}$ for parameterizing any $s \in \mathcal{P}$ is $\lambda_\mathcal{Z} = -\gamma + \Gamma G_1^{-1}(s - c_1)$, i.e., s can be expressed by $s = c_2 + G_2 \lambda_\mathcal{Z}$.

Proof: Based on (2), any $s \in \mathcal{P}$ can be parameterized by the unique parameter vector $\lambda_\mathcal{P} = G_1^{-1}(s - c_1)$ of $\mathcal{P}$ with $|\lambda_\mathcal{P}| \leq \mathbf{1}$. If (4a) and (4b) are satisfied, it follows that

$$c_1 + G_1 \lambda_\mathcal{P} = c_2 + G_2(-\gamma + \Gamma \lambda_\mathcal{P})$$

for any $\lambda_\mathcal{P} \in \mathbb{R}^n$. In addition, satisfaction of (4c) implies $|-\gamma + \Gamma \lambda_\mathcal{P}| \leq \mathbf{1}$ for all $\lambda_\mathcal{P}$ with $|\lambda_\mathcal{P}| \leq \mathbf{1}$. Thus, choosing $\lambda_\mathcal{Z} = -\gamma + \Gamma \lambda_\mathcal{P}$ results in a valid parameter vector of $\mathcal{Z}$ for any $s = c_1 + G_1 \lambda_\mathcal{P}$. ∎

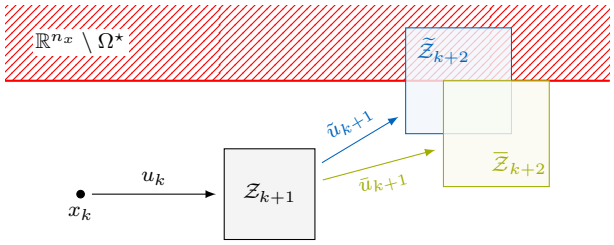


Fig. 2. Set-based safety filter. Because applying the desired input $\tilde{\boldsymbol{u}}_{\boldsymbol{k+1}}$ at time step $\boldsymbol{k+1}$ might lead to leaving the optimal safe set $\boldsymbol{\Omega^\star}$ at time step $\boldsymbol{k+2}$, it is minimally modified to obtain the safe input $\bar{\boldsymbol{u}}_{\boldsymbol{k+1}}$.

If we know that x_0 lies within a small parallelotope that is contained in the zonotopic optimal safe set $\Omega^\star$, we can compute $\lambda_{x,0}$ at time step 0 based on Lemma 3 without solving an optimization problem. Before presenting our optimal control problem that explicitly considers all computation times for solving optimization problems online, subsequently, we give an overview of our set-based safety filter approach.

Because the desired control inputs are only known during runtime, we must solve an optimal control problem online. To explicitly consider such nonnegligible online computational delays, we verify the desired input not for the current but for the next time step, as illustrated in Fig. 2. In particular, at time step $k \in \mathbb{N}_{\geq 0}$, the state x_k is measured, and the input u_k that was previously verified as safe is applied until $k+1$. During this time, we want to verify safety when applying the desired input $\tilde{u}_{k+1} \in \mathbb{R}^{n_u}$ at $k+1$. If safety might be violated, we minimally modify $\tilde{u}_{k+1}$ to obtain the safe input $\bar{u}_{k+1} \in \mathcal{U}$ that ensures $x_{k+2} \in \Omega^\star$. By evaluating the system in (7a) in a set-based fashion, the reachable sets in Fig. 2 are

$$\begin{aligned}
\mathcal{Z}_{k+1} &= \left\{\widehat{A}x_k + \widehat{B}u_k\right\} \oplus \widehat{\mathcal{W}} \\
\widetilde{\mathcal{Z}}_{k+2} &= \widehat{A}\mathcal{Z}_{k+1} \oplus \left\{\widehat{B}\tilde{u}_{k+1}\right\} \oplus \widehat{\mathcal{W}} \\
\overline{\mathcal{Z}}_{k+2} &= \widehat{A}\mathcal{Z}_{k+1} \oplus \left\{\widehat{B}\bar{u}_{k+1}\right\} \oplus \widehat{\mathcal{W}}.
\end{aligned}$$

We also want to mention that it might be infeasible to find a safe input $\bar{u}_{k+1}$ because the optimal safe set $\Omega^\star = \langle c_\Omega^\star, G_\Omega^\star \rangle_Z$ is typically not an RCI set.

To verify or, if necessary, minimally modify the desired input $\tilde{u}_{k+1}$, we solve an optimal control problem that considers all online computation times starting at time step $k \in \mathbb{N}_{\geq 0}$. Let

$\bar{u}^\star_{k+1}, \gamma^\star_{k+2}, \Gamma^\star_{k+2}, c^\star_{\mathcal{P}_{k+2}}, G^\star_{\mathcal{P}_{k+2}}$ be the solution of the COP

$$\begin{aligned}
\underset{\substack{\bar{u}_{k+1},\gamma_{k+2},\Gamma_{k+2}\\ c_{\mathcal{P}_{k+2}},G_{\mathcal{P}_{k+2}}}}{\text{minimize}} \quad & \|\tilde{u}_{k+1} - \bar{u}_{k+1}\|_p && \text{(22a)}\\
\text{subject to} \quad & \bar{u}_{k+1} \in \mathcal{U} && \text{(22b)}\\
& \overline{\mathcal{Z}}_{k+2} = \{\widehat{A}^2 x_k + \widehat{A}\widehat{B}u_k + \widehat{B}\bar{u}_{k+1}\} \\
& \qquad \oplus \widehat{A}\widehat{\mathcal{W}} \oplus \widehat{\mathcal{W}} && \text{(22c)}\\
& \langle c_{\mathcal{P}_{k+2}}, G_{\mathcal{P}_{k+2}} \rangle_Z = \texttt{para}(\overline{\mathcal{Z}}_{k+2}) && \text{(22d)}\\
& G_{\mathcal{P}_{k+2}} = G^\star_\Omega \Gamma_{k+2} && \text{(22e)}\\
& c^\star_\Omega - c_{\mathcal{P}_{k+2}} = G^\star_\Omega \gamma_{k+2} && \text{(22f)}\\
& \left|\begin{bmatrix}\Gamma_{k+2} & \gamma_{k+2}\end{bmatrix}\right| \mathbf{1} \leq \mathbf{1}, && \text{(22g)}
\end{aligned}$$

where the function para tightly encloses the input zonotope by a parallelotope [64], [65]. Then, the optimal safe input at time step $k+1$ is $\bar{u}^\star_{k+1}$.

Instead of the standard constraint $\overline{\mathcal{Z}}_{k+2} \subseteq \Omega^\star$, we use (22d) through (22g) based on Lemma 3 to ensure that the set-based safe backup control input in (13) with (21) can be computed without solving an optimization problem at time step $k+2$. In particular, the required initial parameter vector is

$$\lambda_{x,0} = -\gamma^\star_{k+2} + \Gamma^\star_{k+2}(G^\star_{\mathcal{P}_{k+2}})^{-1}(x_{k+2} - c^\star_{\mathcal{P}_{k+2}}). \tag{23}$$

In addition, $G^\star_{\mathcal{P}_{k+2}}$ only depends on the generator matrix of $\widehat{A}\widehat{\mathcal{W}} \oplus \widehat{\mathcal{W}}$ based on (22d). Therefore, $G^\star_{\mathcal{P}_{k+2}}$ and its inverse $(G^\star_{\mathcal{P}_{k+2}})^{-1}$ are independent of the current time step k and, thus, are computed only once. Because the inverse of a diagonal matrix can be easily obtained, simple axis-aligned box over-approximations instead of general parallelotopic ones can also be used in (22d). As a result, we only need to perform a few simple matrix operations to compute the safe backup control input in (13) with (21) at time step $k+2$. Moreover, if solving (22) requires a longer time than the sampling period Δt to complete, we abort the optimization prematurely to maintain the validity of our formal safety guarantees.

In summary, the optimal control problem in (22) minimally modifies the desired input $\tilde{u}_{k+1}$ while ensuring that the safe backup control input in (13) can be computed at time step $k+2$ without solving an optimization problem. In the following subsection, we show how the COP in (22) is integrated into our safety filter algorithm.

B. Algorithm

We now present Alg. 1 that implements our safety filter. This algorithm proceeds in two steps: First, the safe input u_k applied to the unknown system in (5) at time step k with $k \in \mathbb{N}_{\geq 0}$ is computed. Second, the COP in (22) is solved to verify or, if necessary, minimally modify the desired input $\tilde{u}_{k+1}$. If (22) is infeasible, i.e., if $\bar{u}_{k+1}$ equals the empty set $\emptyset$, we use the safe backup control input at time step $k+1$. In the following theorem, we show that Alg. 1 achieves the control goal formulated in Subsection II-C.

Theorem 2: Let $\Omega_{\text{mRPI}} \subseteq \mathcal{X}$ with $k_\circ \in \mathbb{N}_{\geq 0}$ satisfy (19), let $\Omega^\star$ be the optimal safe set obtained by solving (20), and let the corresponding optimal model $\mathbf{M}^\star = (\widehat{A}, \widehat{B}, \widehat{\mathcal{W}})$ be also conformant to all online obtained data. In addition, let $x_0 \in \Omega^\star$, $\{\widehat{A}x_0 + \widehat{B}u_0\} \oplus \widehat{\mathcal{W}} \subseteq \Omega^\star$, $u_0 \in \mathcal{U}$, and the COP in (22) be feasible for x_0, u_0, $\tilde{u}_1$. Then, the applied control inputs in Alg. 1 are minimal modifications of the desired inputs so that the safety constraints in (6) are satisfied for the unknown system in (5).

Algorithm 1 Safety filter

1: $\bar{u}^\star_0 \leftarrow u_0$
2: **for** $k \leftarrow 0, 1, 2, \ldots$ **do**
3: get x_k and $\tilde{u}_{k+1}$
4: **if** $\bar{u}^\star_k \neq \emptyset$ **then** ▷ use solution of (22)
5: $u_k \leftarrow \bar{u}^\star_k$
6: $k_{x,0} \leftarrow 0$ ▷ reset initial time step
7: **else** ▷ use safe backup control input
8: **if** $k_{x,0} = 0$ **then**
9: $\lambda_{x,0} \leftarrow -\gamma^\star_k + \Gamma^\star_k (G^\star_{\mathcal{P}_k})^{-1}(x_k - c^\star_{\mathcal{P}_k})$ ▷ (23)
10: $k_{x,0} \leftarrow k$ ▷ update initial time step
11: **end if**
12: $u_k \leftarrow Kx_k + c^{\Omega^\star}_{u,k-k_{x,0}} + G^{\Omega^\star}_{u,k-k_{x,0}} \lambda_{x,0}$ ▷ (21)
13: **end if**
14: apply u_k to the unknown system in (5)
15: $\bar{u}^\star_{k+1}, \gamma^\star_{k+2}, \Gamma^\star_{k+2}, c^\star_{\mathcal{P}_{k+2}}, G^\star_{\mathcal{P}_{k+2}} \leftarrow$ solve (22) for $x_k, u_k, \tilde{u}_{k+1}$
16: **end for**

Proof: Because $\mathbf{M}^\star$ is also conformant to all online obtained data, the satisfaction of the safety constraints in (6) for the estimated system in (7a) implies constraint satisfaction for the unknown system in (5). Thus, it is sufficient to consider (7a).

We use our safe backup controller to guarantee safety for an infinite time horizon. Because the initial time was chosen to be zero during its construction in Section III, we appropriately shift the counter $k \in \mathbb{N}_{\geq 0}$ of the correction input zonotope $\mathcal{Z}^{\Omega^\star}_{u,k}$ in line 12 of Alg. 1. Then, applying the resulting safe backup control inputs to the system ensures the satisfaction of the safety constraints in (6) based on Proposition 5.

If the COP in (22) is feasible, the control inputs in line 5 of Alg. 1 are minimal modifications of the desired inputs for the cost function in (22a). In addition, the state and input constraints are satisfied for the next time step because of the incorporated reachability analysis in (22) and $\Omega^\star \subseteq \mathcal{X}$. Moreover, if the COP in (22) is infeasible, we use the safe backup controller until it is feasible again. ■

In summary, Alg. 1 ensures the satisfaction of the safety constraints in (6) while considering all computation times for solving optimization problems. This statement is only valid if the optimal conformant model $\mathbf{M}^\star = (\widehat{A}, \widehat{B}, \widehat{\mathcal{W}})$ is valid at all times, which, however, is constructed offline in Subsection III-A based on a finite set of training data. Because the system in (5) is unknown, we have no guarantee that $\hat{w}_k \in \widehat{\mathcal{W}}$ for all $k \in \mathbb{N}_{\geq 0}$. Thus, we perform conformance updates online to restore formal safety guarantees if a model invalidation is detected, as presented in the following subsection.

C. Online Conformance Updates

We update $\mathbf{M}^\star$, $\Omega^\star$, and $\mathcal{Z}^{\Omega^\star}_{u,(\cdot)}$ online as soon as $\hat{w}_k \notin \widehat{\mathcal{W}}$ to restore formal safety guarantees, similar to [12]. Restoring a conformant model can be achieved by solving the COPs presented in Subsection III-A, including the offline training data and all online obtained data. Although the number of constraints scales only linearly with the amount of data, this approach quickly poses computational and memory problems as time proceeds. Therefore, an update is needed that is independent of the amount of online data, which implies independence of the elapsed time.

We address this issue by fixing $\widehat{A}$ and $\widehat{B}$ of our offline-constructed optimal conformant model $\mathbf{M}^\star = \left(\widehat{A}, \widehat{B}, \widehat{\mathcal{W}}\right)$ and by minimally enlarging $\widehat{\mathcal{W}}$ to restore model conformance. To enable a fast update procedure, we restrict $\widehat{\mathcal{W}}$ to be a simple axis-aligned box $\left[\underline{\widehat{\mathcal{W}}}, \overline{\widehat{\mathcal{W}}}\right]$. In general, we denote by $s^{(i)}$ the i^{th} element of the vector $s \in \mathbb{R}^{n_x}$ with $i \in \mathbb{N}_{[1,n_x]}$. If we detect that $\hat{w}_k^{(i)} < \underline{\widehat{\mathcal{W}}}^{(i)}$ or $\overline{\widehat{\mathcal{W}}}^{(i)} < \hat{w}_k^{(i)}$ for any i, we set $\underline{\widehat{\mathcal{W}}}^{(i)}$ or $\overline{\widehat{\mathcal{W}}}^{(i)}$ equal to $\hat{w}_k^{(i)}$ to restore model conformance. After updating $\mathbf{M}^\star$, we also check if there still exists an $\Omega_{\text{mRPI}} \subseteq \mathcal{X}$ with $k_\circ \in \mathbb{N}_{\geq 0}$ satisfying (19). In addition, we update $\Omega^\star$ by solving (20) and compute the resulting correction input zonotope sequence $\mathcal{Z}^{\Omega^\star}_{u,(\cdot)}$ in (21).

Because no conformant model is available during these online updates, the satisfaction of the safety constraints in (6) can no longer be formally guaranteed. Thus, quickly performing these updates and reducing the probability of constraint violation using the previous safe backup controller is the best we can do in this situation. Therefore, if $\hat{w}_k \notin \widehat{\mathcal{W}}$, we set the control input $\bar{u}_k^\star$ in Alg. 1 equal to $\emptyset$ as long as our online conformance update process is running.

V. Numerical Examples

In this section, we demonstrate the effectiveness of our safety filter approach using four numerical examples from the literature. To compute reachable sets, we use our open-source reachability analysis toolbox CORA [66]. Moreover, all optimization problems are modeled using YALMIP [67] with the Boolean parameter "allownonconvex" set to false and solved using MOSEK [68] with default parameters. To show the low conservativeness of our large safe sets, we also compute tight RCI under-approximations of the MRCI set for the two low-dimensional examples following the approach in [25] by using the toolbox MPT3 [69]. Our computations are conducted on a laptop equipped with an Intel Core i7-1185G7 and 32 GB memory.

For all four numerical examples, we make the following choices: The sampling period is $\Delta t = 0.1\,\text{s}$. In addition, we increment $k_{\text{mRPI}} \in \mathbb{N}_{>0}$ based on Theorem 1 until the cost of the COP in (20) is unchanged for two consecutive iterations or 50 iterations are reached, which is done to ensure finite termination of the iterative procedure. This final k_{mRPI} is used for all subsequent online conformance updates. The cost in (20a) is $J_\Omega = \mathbf{1}^T s_\Omega$ so that (20) is a simple linear program [46]. After solving (20) for the offline training data, we erase the i^{th} column of G_{fixed} if the i^{th} element of the optimal generator scaling vector $s_\Omega^\star$ is smaller than 0.05. This erasure is done because these generators of G_{fixed} significantly increase the computation times when solving (20) online to perform conformance updates. However, the shape of the optimal $\Omega^\star$ is typically only slightly changed as $\widehat{A}$ and $\widehat{B}$ are fixed. Moreover, we choose the 2-norm for the cost function in (22a).

As mentioned in Subsection II-C, the training data is not required to be recorded from a single run of the system but can be obtained by performing multiple short experiments. This useful property allows us to handle unstable systems efficiently. Because the chosen experiment design [33], [55] for training data generation is irrelevant to our approach, for simplicity, we generate the training data by sampling uniformly from $\mathcal{X}$, $\mathcal{U}$, and $\mathcal{W}$.

A. 2D System without Disturbances

We consider the mass-spring-damper example presented in [21]. The unknown system in (5) is described by

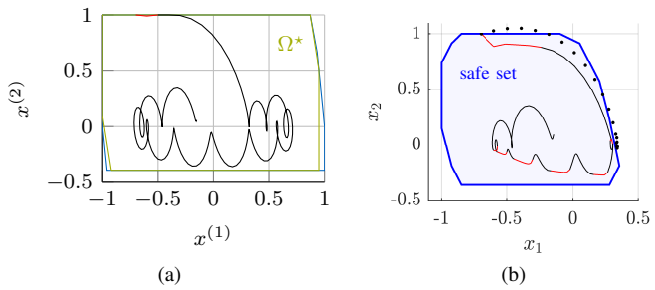
$$x_{k+1} = \begin{bmatrix} 1.0 & 0.1 \\ -0.3 & 0.8 \end{bmatrix} x_k + \begin{bmatrix} 0.0 \\ 0.1 \end{bmatrix} u_k + w_k.$$

The disturbance set is $\mathcal{W} = \{\mathbf{0}\}$. The axis-aligned input and state constraint boxes are described by $\mathcal{U} = [-2.5, 2.5]$, $\underline{\mathcal{X}} = \begin{bmatrix} -1 & -0.4 \end{bmatrix}^T$, and $\overline{\mathcal{X}} = \begin{bmatrix} 1 & 1 \end{bmatrix}^T$, respectively. The stabilizing feedback matrix $K = \begin{bmatrix} -4.12 & -5.32 \end{bmatrix}$ is computed using LQR-based controller synthesis based on approximate system and input matrices that are assumed to be known. In addition, it is assumed that training data $\{x_i, u_i, x_{i+1}\}_{i=1}^{600}$ is generated by sampling uniformly from $\mathcal{X}$ and $\mathcal{U}$. The initial state is $x_0 = \begin{bmatrix} -0.7 & 1 \end{bmatrix}^T$ and the desired input is $\tilde{u}_k = 2\sin(0.01\pi k) + 0.5\sin(0.12\pi k)$ for $k \in \mathbb{N}_{[0,200]}$.

By solving the linear program in (10), we obtain the conformant model $\mathbf{M} = \left(\widehat{A}, \widehat{B}, \widehat{\mathcal{W}}\right)$ based on the available training data. Because $\mathbf{M}$ equals the unknown model $(A, B, \{\mathbf{0}\})$ up to floating-point precision, we never have to update $\mathbf{M}$ online. To cover $\mathcal{X}$, we choose the columns of $G_{\text{fixed}} \in \mathbb{R}^{2\times 20}$ in (20b) to be 20 uniformly distributed points around the top half unit circle.

In Fig. 3a, we present the simulation results when choosing $u_0 = -0.2$ for the initial input. As observed, our safety filter minimally modifies the desired input only in the first two time steps. Thus, our method intervenes significantly less than the safety filter approach in [21], whose performance is shown in Fig. 3b. To illustrate the low conservativeness of our optimal safe set $\Omega^\star$ in Fig. 3a, we also visualize a tight RCI under-approximation of the MRCI set [25].

To compare the set membership identification methods presented in Subsection III-A, we subsequently assume that the unknown disturbance set $\mathcal{W}$ is not the origin but given by $[-0.1, 0.1]^2$. In Fig. 4, we show the volumes of the estimated disturbance sets $\widehat{\mathcal{W}}_{(9)}$, $\widehat{\mathcal{W}}_{(10)}$, and $\widehat{\mathcal{W}}_{\text{LLS}}$ that are obtained by solving (9), (10), and a linear least-squares system identification problem with subsequent parallelotopic volume minimization, respectively. As can be observed in Fig. 4, the volume of $\widehat{\mathcal{W}}_{(9)}$ is always smaller than the volume of $\widehat{\mathcal{W}}_{(10)}$.



(a) (b)

Fig. 3. Comparison of different safety filter approaches for the 2D system. Red (black) color corresponds to states for which the desired input is (is not) minimally modified to always guarantee safety. (a) Our approach. A tight RCI under-approximation of the MRCI set is visualized in blue, which shows the low conservativeness of our large safe set $\boldsymbol{\Omega}^\star$. (b) This figure is taken from [21]. The dotted black curve can be ignored.

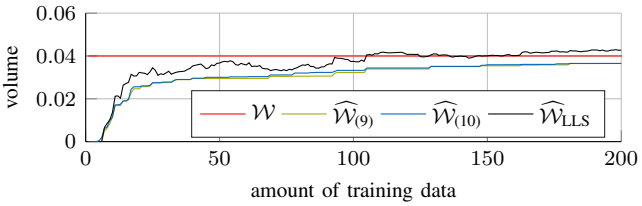


Fig. 4. Comparison of different system identification approaches for the 2D system. In addition to the unknown disturbance set $\boldsymbol{\mathcal{W}}$, we visualize $\widehat{\boldsymbol{\mathcal{W}}}_{\text{LLS}}$, obtained by performing linear least-squares system identification with subsequent parallelotopic volume minimization. Moreover, $\widehat{\boldsymbol{\mathcal{W}}}_{(9)}$ and $\widehat{\boldsymbol{\mathcal{W}}}_{(10)}$ are obtained by solving (9) and (10), respectively.

In addition, both volumes are monotonically increasing and converging to the volume of $\mathcal{W}$ from below. In contrast to this monotonic increase, the volume of $\widehat{\mathcal{W}}_{\text{LLS}}$ fluctuates and even exceeds $\mathcal{W}$. This observation shows the advantage when performing system identification and volume minimization in one step.

B. Unstable 3D System

To demonstrate the usefulness of our online conformance updates proposed in Subsection IV-C, we consider the unstable system presented in [36]. The unknown system in (5) is described by

$$x_{k+1} = \begin{bmatrix} -0.5 & 1.4 & 0.4 \\ -0.9 & 0.3 & -1.5 \\ 1.1 & 1.0 & -0.4 \end{bmatrix} x_k + \begin{bmatrix} 0.1 & -0.3 \\ -0.1 & -0.7 \\ 0.7 & -1.0 \end{bmatrix} u_k + w_k,$$

and the state feedback matrix is

$$K = \begin{bmatrix} -2.45 & -1.29 & -2.40 \\ -0.61 & -0.03 & -2.18 \end{bmatrix}. \tag{24}$$

We assume that the unknown disturbance set is $\mathcal{W} = [-0.04, 0.04]^3$, and the known state and input constraint sets are $\mathcal{X} = [-1, 1]^3$ and $\mathcal{U} = [-1, 1]^2$. The initial state x_0 and the initial input u_0 are the origin. The desired input $\tilde{u}_k$ and the disturbance w_k are uniformly sampled online from $\mathcal{U}$ and $\mathcal{W}$ for all $k \in \mathbb{N}_{[0,10^5]}$.

We generate training data $\{x_i, u_i, x_{i+1}\}_{i=1}^{100}$ by sampling uniformly from $\mathcal{X}$, $\mathcal{U}$, and $\mathcal{W}$. By solving the linear program

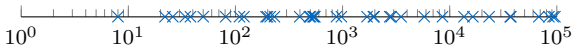
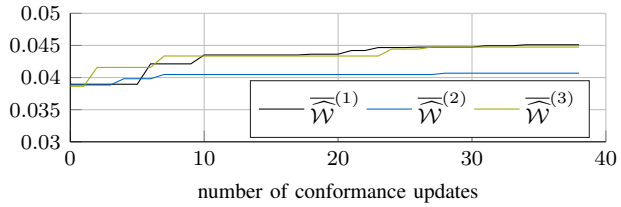
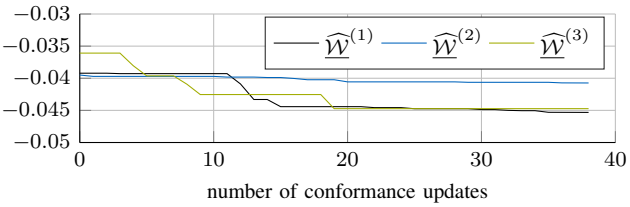


Fig. 5. Online conformance updates of the 3D system. The time steps $\boldsymbol{k}$ are marked when conformance updates are performed, i.e., when $\boldsymbol{w_k} \notin \widehat{\boldsymbol{\mathcal{W}}}$ is detected.



(a)



(b)

Fig. 6. Evolution of the upper (a) and lower (b) bounds of the axis-aligned estimated disturbance box $\widehat{\boldsymbol{\mathcal{W}}}$ of the 3D system, which is initialized in (25).

in (10), we obtain the optimal conformant linear model $\mathrm{M}^\star = \left(\widehat{A}, \widehat{B}, \widehat{\mathcal{W}}\right)$ with

$$\widehat{A} = \begin{bmatrix} -0.5001 & 1.4013 & 0.3991 \\ -0.8998 & 0.3001 & -1.5004 \\ 1.0997 & 0.9997 & -0.4020 \end{bmatrix}$$

$$\widehat{B} = \begin{bmatrix} 0.0994 & -0.2966 \\ -0.0997 & -0.6997 \\ 0.6983 & -0.9977 \end{bmatrix}$$

$$\widehat{\mathcal{W}} = \begin{bmatrix} -0.0392 \\ -0.0395 \\ -0.0361 \end{bmatrix}, \ \overline{\widehat{\mathcal{W}}} = \begin{bmatrix} 0.0390 \\ 0.0389 \\ 0.0387 \end{bmatrix}. \tag{25}$$

Thus, the state feedback matrix in (24) stabilizes the estimated system $\left(\widehat{A}, \widehat{B}\right)$. Nevertheless, any stabilizing feedback matrix could be deployed, e.g., using LQR-based controller synthesis [70]. Moreover, because $\widehat{\mathcal{W}} \subset \mathcal{W}$, model invalidation will likely occur, requiring us to update $\widehat{\mathcal{W}}$ online. To cover $\mathcal{X}$, we choose the columns of $G_{\text{fixed}} \in \mathbb{R}^{3\times 70}$ in (20b) to be 70 uniformly distributed points around the unit sphere.

In Fig. 5, we plot the 38 time steps $k \in \mathbb{N}_{\geq 0}$ when $w_k \notin \widehat{\mathcal{W}}$ is detected. We update the model, the safe set, and the safe backup controller at these time steps, as proposed in Subsection IV-C. Using a logarithmic scale makes it clear that most updates occur early on.

In Fig. 6, we visualize the evolution of the lower and upper bounds of the estimated disturbance set $\widehat{\mathcal{W}}$, which is initialized in (25). As more uniformly sampled disturbances are gathered online, the changes in all three dimensions become smaller.

In Fig. 7, we show two-dimensional projections of the initial optimal safe set and a tight RCI under-approximation of the initial MRCI set based on the estimated disturbance bounds in

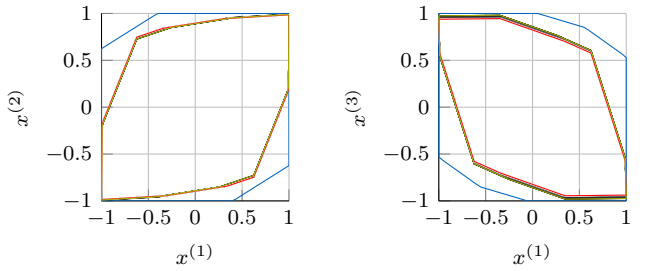


Fig. 7. Evolution of safe sets of the 3D system. The initial optimal safe set, the final optimal safe set, and a tight RCI under-approximation of the initial MRCI set are shown in green, red, and blue, respectively. In addition, the updated optimal safe sets are visualized, with a lighter gray tone corresponding to a higher number of updates.

(25). In addition, we visualize the 38 updated optimal safe sets corresponding to updated conformant models. As can be seen in the $x^{(1)}$-$x^{(2)}$-plot in Fig. 7, the updated safe sets shrink in some generator directions but also grow in some others. Because computing a tight RCI under-approximation of the initial MRCI set takes more than 1 min, these computations are unsuitable for online updating. Nevertheless, our online conformance updates, which include updating $\widehat{\mathcal{W}}$, verifying the existence of an $\Omega_{\text{mRPI}} \subseteq \mathcal{X}$ satisfying (19), and updating $\Omega^\star$ and corresponding $\mathcal{Z}_{u,(\cdot)}^{\Omega^\star}$, take 57 ms on average with a standard deviation of 4 ms. Thus, these updates are performed in real time.

C. Nonlinear Continuous-Time 6D System

To demonstrate the generalizability of our approach, we consider the nonlinear, continuous-time, longitudinal quadrotor model proposed in [30], [71]. The unknown system is described by the set of ordinary differential equations

$$\dot{x}^{(1)} = x^{(3)} \tag{26a}$$
$$\dot{x}^{(2)} = x^{(4)} \tag{26b}$$
$$\dot{x}^{(3)} = u^{(1)} n_1 \sin(x^{(5)}) \tag{26c}$$
$$\dot{x}^{(4)} = u^{(1)} n_1 \cos(x^{(5)}) - g \tag{26d}$$
$$\dot{x}^{(5)} = x^{(6)} \tag{26e}$$
$$\dot{x}^{(6)} = -d_0 x^{(5)} - d_1 x^{(6)} + n_0 u^{(2)}, \tag{26f}$$

where $x^{(1)}$ to $x^{(6)}$ represent the horizontal and vertical positions, horizontal and vertical velocities, roll, and roll velocity, respectively. The constant parameters are $g = 9.81$, $d_0 = 70$, $d_1 = 17$, $n_0 = 55$, $n_1 = 0.89/1.4$, and the axis-aligned state and input constraint boxes are described by

$$\underline{\mathcal{X}} = \begin{bmatrix} -1.7 & 0.3 & -0.8 & -1 & -\pi/12 & -\pi/2 \end{bmatrix}^T$$
$$\overline{\mathcal{X}} = \begin{bmatrix} 1.7 & 2.0 & 0.8 & 1 & \pi/12 & \pi/2 \end{bmatrix}^T$$
$$\underline{\mathcal{U}} = \begin{bmatrix} g/n_1 - 1.5 & -\pi/12 \end{bmatrix}^T$$
$$\overline{\mathcal{U}} = \begin{bmatrix} g/n_1 + 1.5 & \pi/12 \end{bmatrix}^T.$$

To satisfy our assumption that $\mathcal{X}$ and $\mathcal{U}$ contain the origin, we perform a simple coordinate transformation, i.e., we shift $x^{(2)}$ by -1.15 and $u^{(1)}$ by $-g/n_1$. We generate training data $\{x_i, u_i, x_{i+1}\}_{i=1}^{100}$ by using the MATLAB function ode45[1] to solve (26) and by sampling uniformly from $\mathcal{X}$ and $\mathcal{U}$. In addition, we compute the stabilizing feedback matrix K using LQR-based controller synthesis [70], where the state and input weighting matrices are $Q = 10I$ and $R = I$. The fixed generator matrix $G_{\text{fixed}} \in \mathbb{R}^{6\times 48}$ in (20b) is taken from [30]. The initial state is $x_0 = \begin{bmatrix} 0 & 1.15 & 0 & 0 & 0 & 0 \end{bmatrix}^T$ and the initial input is $u_0 = \begin{bmatrix} g/n_1 & 0 \end{bmatrix}^T$. Moreover, the desired input $\tilde{u}_k$ is uniformly sampled online from $\mathcal{U}$ for all $k \in \mathbb{N}_{[0,10^5]}$.

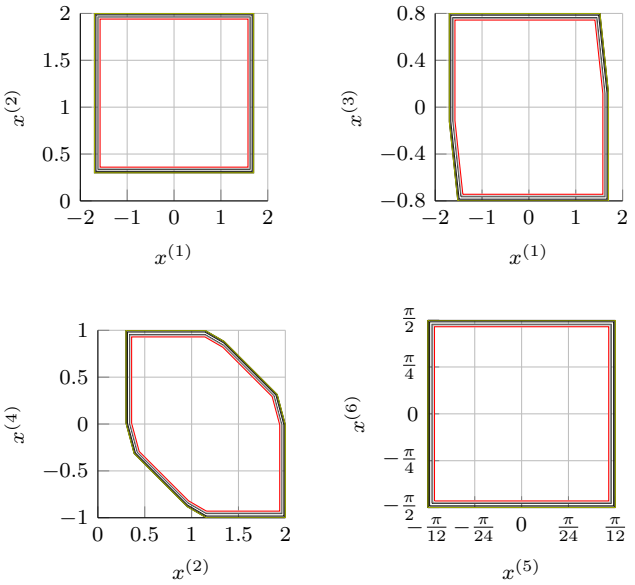


Fig. 8. Evolution of safe sets of the 6D system. The initial and final optimal safe sets are shown in green and red, respectively. In addition, the updated optimal safe sets are visualized, with a lighter gray tone corresponding to a higher number of updates.

Solving (20) initially for the offline training data takes 19 s. To enable real-time conformance updates, we slightly simplify (20), i.e., we restrict Ω and $\mathcal{Z}_{u,(\cdot)}$ to be scaled versions of the optimal initial zonotopes, as shown in Fig. 8. As a result, our update takes only 145 ms on average with a standard deviation of 8 ms. Thus, our approach can update formal safety guarantees at sampling times for nonlinear, continuous-time systems in real time.

To demonstrate the difficulty of this numerical example, we compare our results with two existing methods for computing safe sets. Because the approach in [25] has an exponential computational complexity with respect to the state space dimension, we abort the corresponding computations prematurely after 24 h. We also use the method in [72], which requires the linear system to be presented in controller canonical form. Using the corresponding publicly available code, the transformation of our initial conformant model to this form involves the inverse of a matrix whose condition number is greater than 10^6, which leads to significant numerical errors.

[1] https://mathworks.com/help/matlab/ref/ode45.html

D. Continuous-Time 12D System

To demonstrate the scalability of our approach, we consider the under-actuated, continuous-time quadrotor model proposed in [29], [73]. The system dynamics is linearized around the hover condition to obtain a linear model. The resulting twelve states are given by

- the spatial positions $\begin{bmatrix} x^{(1)} & x^{(2)} & x^{(3)} \end{bmatrix}^T \in [-3, 3]^3$ and
- their velocities $\begin{bmatrix} x^{(4)} & x^{(5)} & x^{(6)} \end{bmatrix}^T \in [-3, 3]^3$, and
- the angular positions $\begin{bmatrix} x^{(7)} & x^{(8)} \end{bmatrix}^T \in [-\pi/4, \pi/4]^2$, $x^{(9)} \in [-\pi, \pi]$ and
- their velocities $\begin{bmatrix} x^{(10)} & x^{(11)} & x^{(12)} \end{bmatrix}^T \in [-3, 3]^3$.

In addition, the four control inputs are given by

- the total normalized thrust $u^{(1)} \in [-9.81, 2.38]$ and
- the second-order derivatives of the angular positions $\begin{bmatrix} u^{(2)} & u^{(3)} & u^{(4)} \end{bmatrix}^T \in [-0.5, 0.5]^3$.

The wind is modeled by the unknown, bounded disturbance $\begin{bmatrix} w^{(4)} & w^{(5)} & w^{(6)} \end{bmatrix}^T \in [-0.05, 0.05]^3$ that affects only the three spatial velocities.

We generate training data $\{x_i, u_i, x_{i+1}\}_{i=1}^{1000}$ by using the MATLAB function ode45 to solve the system of linear differential equations and by sampling uniformly from the state and input constraint sets. In addition, we compute the stabilizing feedback matrix K using LQR-based controller synthesis [70], where the state and input weighting matrices are $Q = 10I$ and $R = I$. The fixed generator matrix $G_{\text{fixed}} \in \mathbb{R}^{12\times 52}$ in (20b) is obtained following the approach in [30], i.e., by examining the sparsity of the system matrix. The initial state x_0 and the initial input u_0 are the origin. Moreover, the desired input $\tilde{u}_k$ is uniformly sampled online from the input constraint set for all $k \in \mathbb{N}_{[0,10^5]}$.

Two-dimensional projections of the initial solution of (20) are shown in Fig. 9. Because solving (20) initially for the offline training data takes 40 min, we slightly simplify (20) analogously to Subsection V-C. As a result, our 20 updates take 1.01 s on average with a standard deviation of 85 ms. In summary, our approach quickly updates formal safety guarantees for medium-sized problems.

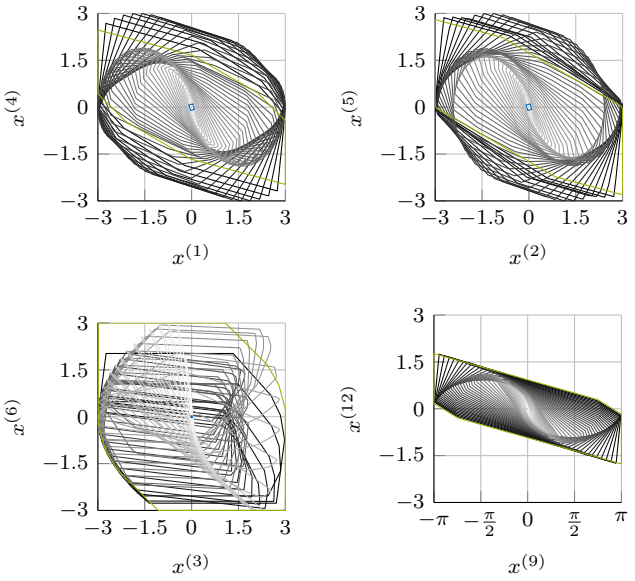


Fig. 9. Initial safe sets of the 12D system. The small safe set $\boldsymbol{\Omega}_{\text{mRPI}}$ and the optimal large safe set $\boldsymbol{\Omega}^\star$ are shown in blue and green, respectively. In addition, the reachable sets $\boldsymbol{\mathcal{R}}(\boldsymbol{k}, \boldsymbol{\Omega}^\star, \boldsymbol{\mathcal{Z}}^\star_{\boldsymbol{u},(\cdot)}, \widetilde{\boldsymbol{\mathcal{W}}})$ with $\boldsymbol{k} \in \mathbb{N}_{[\mathbf{1},\mathbf{50}]}$ are shown, with a lighter gray tone corresponding to a bigger $\boldsymbol{k}$.

VI. Conclusions

We have presented safety filters that provide formal safety guarantees for any controller. If the desired input might lead to leaving our large safe set in the future, it is modified in the least restrictive way. Unlike most other work on robust controller synthesis, we make no assumptions about the availability of the disturbance set. Thus, we perform offline set membership identification based on a finite set of available training data. Because a new measurement obtained online might invalidate the formal safety guarantees of our safety filter, fast online conformance updating is crucial. In contrast to existing work, our updates are performed in real time, even for medium-sized problems. These real-time updates are enabled by designing our update procedure to be independent of the number of measurements and by using scalable reachability analysis as well as convex optimization algorithms. We have demonstrated the effectiveness, generalizability, and scalability of our safety filter approach using four numerical examples from the literature, including a six-dimensional, nonlinear system.

References

[1] M. Alshiekh, R. Bloem, R. Ehlers, B. Könighofer, S. Niekum, and U. Topcu, "Safe reinforcement learning via shielding," in *AAAI Conference on Artificial Intelligence*, 2018, pp. 2669–2678.

[2] L. Hewing, K. P. Wabersich, M. Menner, and M. N. Zeilinger, "Learning-based model predictive control: Toward safe learning in control," *Annual Review of Control, Robotics, and Autonomous Systems*, vol. 3, no. 1, pp. 269–296, 2020.

[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] M. Althoff and J. M. Dolan, "Online verification of automated road vehicles using reachability analysis," *IEEE Transactions on Robotics*, vol. 30, no. 4, pp. 903–918, 2014.

[5] F. Gruber and M. Althoff, "Anytime safety verification of autonomous vehicles," in *IEEE Conference on Intelligent Transportation Systems*, 2018, pp. 1708–1714.

[6] A. Colombo and D. Del Vecchio, "Least restrictive supervisors for intersection collision avoidance: A scheduling approach," *IEEE Transactions on Automatic Control*, vol. 60, no. 6, pp. 1515–1527, 2015.

[7] L. Sha, "Using simplicity to control complexity," *IEEE Software*, vol. 18, no. 4, pp. 20–28, 2001.

[8] B. Könighofer, M. Alshiekh, R. Bloem, L. Humphrey, R. Könighofer, U. Topcu, and C. Wang, "Shield synthesis," *Formal Methods in System Design*, vol. 51, no. 2, pp. 332–361, 2017.

[9] N. Aréchiga and B. H. Krogh, "Using verified control envelopes for safe controller design," in *American Control Conference*, 2014, pp. 2918–2923.

[10] S. Bak, K. Manamcheri, S. Mitra, and M. Caccamo, "Sandboxing controllers for cyber-physical systems," in *IEEE/ACM Conference on Cyber-Physical Systems*, 2011, pp. 3–12.

[11] S. Bak, T. T. Johnson, M. Caccamo, and L. Sha, "Real-time reachability for verified simplex design," in *IEEE Real-Time Systems Symposium*, 2014, pp. 138–148.

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

[13] J. Wolff and M. Buss, "Invariance control design for nonlinear control affine systems under hard state constraints," in *IFAC Symposium on Nonlinear Control Systems*, 2004, pp. 555–560.

[14] M. Kimmel and S. Hirche, "Invariance control for safe human-robot interaction in dynamic environments," *IEEE Transactions on Robotics*, vol. 33, no. 6, pp. 1327–1342, 2017.

[15] A. D. Ames, X. Xu, J. W. Grizzle, and P. Tabuada, "Control barrier function based quadratic programs for safety critical systems," *IEEE Transactions on Automatic Control*, vol. 62, no. 8, pp. 3861–3876, 2017.

[16] B. T. Lopez, J.-J. E. Slotine, and J. P. How, "Robust adaptive control barrier functions: An adaptive and data-driven approach to safety," *IEEE Control Systems Letters*, vol. 5, no. 3, pp. 1031–1036, 2021.

[17] E. Garone, S. Di Cairano, and I. Kolmanovsky, "Reference and command governors for systems with constraints: A survey on theory and applications," *Automatica*, vol. 75, pp. 306–328, 2017.

[18] F. Borrelli, A. Bemporad, and M. Morari, *Predictive Control for Linear and Hybrid Systems*. Cambridge University Press, 2017.

[19] S. V. Raković and W. S. Levine, Eds., *Handbook of Model Predictive Control*, ser. Control Engineering. Birkhäuser, 2019.

[20] F. Gruber and M. Althoff, "Scalable robust output feedback MPC of linear sampled-data systems," in *IEEE Conference on Decision and Control*, 2021, pp. 2563–2570.

[21] 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.

[22] P. Cardaliaguet, "A differential game with two players and one target," *SIAM Journal on Control and Optimization*, vol. 34, no. 4, pp. 1441–1460, 1996.

[23] 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.

[24] F. Blanchini and S. Miani, *Set-Theoretic Methods in Control*, 2nd ed. Birkhäuser, 2015.

[25] 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.

[26] F. Tahir and I. M. Jaimoukha, "Low-complexity polytopic invariant sets for linear systems subject to norm-bounded uncertainty," *IEEE Transactions on Automatic Control*, vol. 60, no. 5, pp. 1416–1421, 2015.

[27] 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.

[28] 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.

[29] 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.

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

[31] F. Gruber and M. Althoff, "Computing safe sets of linear sampled-data systems," *IEEE Control Systems Letters*, vol. 5, no. 2, pp. 385–390, 2021.

[32] M. Milanese, J. Norton, H. Piet-Lahanier, and É. Walter, Eds., *Bounding Approaches to System Identification*. Springer, 1996.

[33] L. Ljung, "Perspectives on system identification," *Annual Reviews in Control*, vol. 34, no. 1, pp. 1–12, 2010.

[34] H. Roehm, J. Oehlerking, M. Woehrle, and M. Althoff, "Model conformance for cyber-physical systems: A survey," *ACM Transactions on Cyber-Physical Systems*, vol. 3, no. 3, pp. 1–26, 2019.

[35] Y. Chen and N. Ozay, "Data-driven computation of robust control invariant sets with concurrent model selection," *IEEE Transactions on Control Systems Technology*, vol. 30, no. 2, pp. 495–506, 2022.

[36] J. Berberich, A. Koch, C. W. Scherer, and F. Allgöwer, "Robust data-driven state-feedback design," in *American Control Conference*, 2020, pp. 1532–1538.

[37] S. K. Mulagaleti, A. Bemporad, and M. Zanon, "Data-driven synthesis of robust invariant sets and controllers," *IEEE Control Systems Letters*, vol. 6, pp. 1676–1681, 2022.

[38] S. Sadraddini and C. Belta, "Formal guarantees in data-driven model identification and control synthesis," in *Conference on Hybrid Systems: Computation and Control*, 2018, pp. 147–156.

[39] E. Terzi, L. Fagiano, M. Farina, and R. Scattolini, "Learning-based predictive control for linear systems: A unitary approach," *Automatica*, vol. 108, p. 108473, 2019.

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

[41] A. Girard, "Reachability of uncertain linear systems using zonotopes," in *Workshop on Hybrid Systems: Computation and Control*. Springer, 2005, pp. 291–305.

[42] 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.

[43] A. Kulmburg and M. Althoff, "On the co-NP-completeness of the zonotope containment problem," *European Journal of Control*, vol. 62, pp. 84–91, 2021.

[44] 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.

[45] 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, p. 100826, 2020.

[46] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge University Press, 2004.

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

[48] H. J. van Waarde, C. De Persis, M. K. Camlibel, and P. Tesi, "Willems' fundamental lemma for state-space systems and its extension to multiple datasets," *IEEE Control Systems Letters*, vol. 4, no. 3, pp. 602–607, 2020.

[49] S. B. Liu and M. Althoff, "Reachset conformance of forward dynamic models for the formal analysis of robots," in *IEEE/RSJ Conference on Intelligent Robots and Systems*, 2018, pp. 370–376.

[50] E. Walter and H. Piet-Lahanier, "Recursive robust minimax estimation for models linear in their parameters," in *IFAC Identification and System Parameter Estimation*, 1992, pp. 215–220.

[51] 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.

[52] A. Chalkis, I. Z. Emiris, and V. Fisikopoulos, "Practical volume estimation of zonotopes by a new annealing schedule for cooling convex bodies," in *International Congress on Mathematical Software*, 2020, pp. 212–221.

[53] L. Vandenberghe, S. Boyd, and S.-P. Wu, "Determinant maximization with linear matrix inequality constraints," *SIAM Journal on Matrix Analysis and Applications*, vol. 19, no. 2, pp. 499–533, 1998.

[54] C. Combastel, "Zonotopes and Kalman observers: Gain optimality under distinct uncertainty paradigms and robust convergence," *Automatica*, vol. 55, pp. 265–273, 2015.

[55] M. Gevers, "Identification for control: From the early achievements to the revival of experiment design," *European Journal of Control*, vol. 11, no. 4-5, pp. 335–352, 2005.

[56] P. J. Goulart, E. C. Kerrigan, and J. M. Maciejowski, "Optimization over state feedback policies for robust control with constraints," *Automatica*, vol. 42, no. 4, pp. 523–533, 2006.

[57] C. D. Meyer, *Matrix Analysis and Applied Linear Algebra*. SIAM, 2000.

[58] S. V. Raković, E. C. Kerrigan, K. I. Kouramas, and D. Q. Mayne, "Invariant approximations of the minimal robust positively invariant set," *IEEE Transactions on Automatic Control*, vol. 50, no. 3, pp. 406–410, 2005.

[59] I. Kolmanovsky and E. G. Gilbert, "Theory and computation of disturbance invariant sets for discrete-time linear systems," *Mathematical Problems in Engineering*, vol. 4, no. 4, pp. 317–367, 1998.

[60] A. Wintenberg and N. Ozay, "Implicit invariant sets for high-dimensional switched affine systems," in *IEEE Conference on Decision and Control*, 2020, pp. 3291–3297.

[61] B. Schürmann, M. Klischat, N. Kochdumper, and M. Althoff, "Formal Safety Net Control Using Backward Reachability Analysis," *IEEE Transactions on Automatic Control*, vol. 67, no. 11, pp. 5698–5713, 2022.

[62] V. M. Zavala and L. T. Biegler, "The advanced-step NMPC controller: Optimality, stability and robustness," *Automatica*, vol. 45, no. 1, pp. 86–93, 2009.

[63] B. Schürmann, A. El-Guindy, and M. Althoff, "Closed-form expressions of convex combinations," in *American Control Conference*, 2016, pp. 2795–2801.

[64] 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.

[65] X. Yang and J. K. Scott, "A comparison of zonotope order reduction techniques," *Automatica*, vol. 95, pp. 378–384, 2018.
[66] M. Althoff, "An introduction to CORA 2015," in *Workshop on Applied Verification for Continuous and Hybrid Systems*, 2015, pp. 120–151.
[67] J. Löfberg, "YALMIP : A toolbox for modeling and optimization in MATLAB," in *IEEE Symposium on Computer Aided Control Systems Design*, 2004, pp. 284–289.
[68] MOSEK Aps, "The MOSEK optimization toolbox for MATLAB manual. Version 9.2," 2021. [Online]. Available: https://docs.mosek.com/9.2/toolbox/index.html
[69] M. Herceg, M. Kvasnica, C. N. Jones, and M. Morari, "Multi-parametric toolbox 3.0," in *European Control Conference*, 2013, pp. 502–510.
[70] H. Kwakernaak and R. Sivan, *Linear Optimal Control Systems*. Wiley, 1972.
[71] P. Bouffard, "On-board model predictive control of a quadrotor helicopter: Design, implementation, and experiments," EECS Department, University of California, Berkeley, Tech. Rep. UCB/EECS-2012-241, 2012.
[72] T. Anevlavis, Z. Liu, N. Ozay, and P. Tabuada, "Controlled invariant sets: Implicit closed-form representations and applications," 2021.
[73] S. Kaynama and C. J. Tomlin, "Benchmark: Flight envelope protection in autonomous quadrotors," in *Workshop on Applied Verification for Continuous and Hybrid Systems*, 2014.

Felix Gruber is a Ph.D. candidate in Computer Science at the Technical University of Munich, Germany. In 2017, he received the Master of Science degree in electrical engineering from the Technical University of Munich, Germany, and was a visiting student researcher at the University of California, Berkeley, USA. His research interests include control theory, optimization, and reachability analysis, with applications to safety-critical systems.

Matthias Althoff is an Associate Professor in Computer Science at the Technical University of Munich, Germany. He received his diploma engineering degree in Mechanical Engineering in 2005, and his Ph.D. degree in Electrical Engineering in 2010, both from the Technical University of Munich, Germany. From 2010 to 2012 he was a Postdoctoral Researcher at Carnegie Mellon University, Pittsburgh, USA, and from 2012 to 2013 an Assistant Professor at Technische Universität Ilmenau, Germany. His research interests include the formal verification of continuous and hybrid systems, reachability analysis, planning algorithms, nonlinear control, automated vehicles, and power systems.