Scalable robust output feedback MPC of linear sampled-data systems

Felix Gruber and Matthias Althoff

Abstract— **Cyber-physical control systems typically consist of two components: a clocked digital controller and a physical plant evolving in continuous time. Clearly, the state and input constraints must be satisfied not only at, but also between sampling times of the controller. We address this issue by proposing a robust output feedback model predictive control approach for sampled-data systems, which are affected by additive disturbances and measurement noise. To guarantee robust constraint satisfaction for an infinite time horizon, we present a scalable approach to compute safe terminal sets. Based on these sets and using scalable reachability analysis and convex optimization algorithms, we construct real-time controllers that explicitly consider all online computation times. We demonstrate the usefulness of our robust control approach using a vehicle platooning benchmark from the literature.**

I. Introduction

Over the past few decades, model predictive control (MPC) has established itself as a very successful approach for controlling complex dynamical systems, both in academia and industry [1], [2]. Its considerable popularity can be attributed to its simple concept and ability to effectively handle state and input constraints [3], [4]. This is achieved by iteratively solving an optimal control problem on a receding time horizon and by applying the optimized input to the system until the next time step. When using MPC in safety-critical applications, it is crucial to ensure robustness against uncertainties. Therefore, robust MPC approaches that guarantee robust constraint satisfaction despite unknown but bounded uncertainties are required.

Initially, robust MPC has been applied to linear systems with state feedback control. Because a min-max optimization over feedback policies easily becomes impractical due to its computational complexity [5], [6], tube-based MPC approaches have been proposed [7], [8]. The key idea is to ensure that the state of the system remains within a tube surrounding the nominal trajectory satisfying the constraints. Thus, by tightening the constraints appropriately, only the disturbance-free nominal prediction model is required for online computations, whereas computationally expensive set-based operations are performed offline. As a result, tube-based MPC approaches have been successfully employed in real-time [9], [10] and various extensions of these approaches now exist [11], [12]. In addition to state feedback control, output feedback MPC approaches have been proposed because the exact measurement of the state is typically unavailable [13], [14], [15].

Most literature on robust MPC deals with discrete-time models. Thus, constraint satisfaction is only guaranteed at discrete time steps while assuming that the solution of the optimal control problem can be obtained instantaneously. However, in most cyber-physical control systems, a continuous-time physical plant is controlled by a discrete-time digital controller. Therefore, robust constraint satisfaction must also be guaranteed between discrete sampling times. Sampled-data robust MPC approaches have recently gained popularity due to their high practical relevance [16], [17]. For instance, the approach in [8] has been extended to deal with sampled-data systems in [18]. However, the unknown but bounded uncertainty is assumed to be constant between two sampling times, which is unrealistic.

Because most robust MPC approaches suffer from high computational complexity, centralized methods are typically inapplicable when handling large systems. For instance, the computation of a polytopic invariant terminal set is difficult for large systems because performing the required set operations of high-dimensional polytopes becomes intractable [19], [20]. Typically, these obstacles can only be overcome if the control problem can be decomposed into simpler ones, as in distributed MPC [21], [22]. Instead of polytopes, ellipsoids are also widely used to analyze the evolution of the uncertainty based on scalable ellipsoidal reachability analysis [23]. However, ellipsoids underapproximate typical box constraints in high dimensions very poorly [24], [25], increasing conservatism.

This paper is based on our preliminary results presented in [26], where state feedback control is considered and simple safe terminal boxes are computed. In this paper, we

- use zonotopes as an efficient set representation;
- compute zonotopic safe terminal sets to guarantee robust constraint satisfaction for an infinite time horizon;
- explicitly consider the computation time of the online optimal control problem;
- propose a real-time robust output feedback MPC approach for linear sampled-data systems; and
- demonstrate the usefulness of our control approach using a nine-dimensional benchmark from the literature with a sampling time of 150 ms.

The rest of this paper is structured as follows: In Section II, zonotopes as an efficient set representation are introduced and the control problem is formulated. Based on our reachability analysis in Section III, our robust output feedback dual-mode MPC approach is presented in Section IV, followed by a numerical example in Section V. Conclusions and suggestions for future work are finally provided in Section VI.

This work was supported by the European Commission Project justIT-SELF 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

II. Preliminaries

In this section, we introduce zonotopes as an efficient set representation. In addition, we state the control problem.

A. Set representation by zonotopes

A crucial aspect of reachability analysis is the choice of the set representation, e.g., ellipsoids [23] or polytopes [27]. We use zonotopes as an efficient set representation because typical box constraints can be exactly represented and the computational complexity of our reachability analysis reduces to $\mathcal{O}\big(n_x^3\big)$ with $n_x \in \mathbb{N}_{>0}$ denoting the state space dimension [28], [29]. In addition, zonotopes can be stored efficiently as matrices.

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

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

where $c \in \mathbb{R}^{n_\mathcal{Z}}$ is the center, $G \in \mathbb{R}^{n_\mathcal{Z} \times \eta(\mathcal{Z})}$ is the generator matrix with $\eta\left(\mathcal{Z}\right) \in \mathbb{N}$ denoting the number of generators, and the infinity norm is defined as the maximum absolute row sum. To use a more concise notation, we write $\mathcal{Z} = \langle c, G\rangle$. Based on this definition, it follows that zonotopes are convex, closed, centrally symmetric polytopes.

According to [30], the Minkowski addition of two zonotopes $\mathcal{Z}_1 = \langle c_1, G_1\rangle \subset \mathbb{R}^{n_\mathcal{Z}}$ and $\mathcal{Z}_2 = \langle c_2, G_2\rangle \subset \mathbb{R}^{n_\mathcal{Z}}$ and multiplication by a matrix $M \in \mathbb{R}^{m \times n_\mathcal{Z}}$ are

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

The simple structure of these two fundamental operations enables our reachability analysis to handle large linear systems [28], [29]. To determine whether $\mathcal{Z}_1$ is contained in $\mathcal{Z}_2$, i.e., if $\mathcal{Z}_1 \subseteq \mathcal{Z}_2$, we check whether a matrix $\Gamma \in \mathbb{R}^{\eta(\mathcal{Z}_2)\times\eta(\mathcal{Z}_1)}$ and a vector $\gamma \in \mathbb{R}^{\eta(\mathcal{Z}_2)}$ exist such that [31]

$$G_1 = G_2\Gamma \tag{1a}$$

$$c_2 - c_1 = G_2\gamma \tag{1b}$$

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

The zonotope containment problem is also known to be co-NP-complete [32]. In addition, we define the stacking of $\mathcal{Z}_1 = \langle c_1, G_1\rangle \subset \mathbb{R}^{n_{\mathcal{Z}_1}}$ and $\mathcal{Z}_2 = \langle c_2, G_2\rangle \subset \mathbb{R}^{n_{\mathcal{Z}_2}}$ by

$$\left\langle \begin{matrix} \mathcal{Z}_1 \\ \mathcal{Z}_2 \end{matrix} \right\rangle = \left\langle \begin{bmatrix} c_1 \\ c_2 \end{bmatrix}, \begin{bmatrix} G_1 \\ G_2 \end{bmatrix} \right\rangle, \tag{2}$$

where the number of generators $\eta\left(\mathcal{Z}_1\right)$ and $\eta\left(\mathcal{Z}_2\right)$ is equal.

B. Problem statement

In this paper, we consider time-invariant physical plants evolving in continuous time according to

$$\dot{x}(t) = Ax(t) + Bu(t) + w(t), \tag{3}$$

where $x(t) \in \mathbb{R}^{n_x}$ is the state and $u(t) \in \mathbb{R}^{n_u}$ is the input at time $t \in \mathbb{R}_{\geq 0}$. The additive state disturbance trajectory $w(\cdot)$ is unknown but bounded by the state disturbance set $\mathcal{W} \subset \mathbb{R}^{n_x}$, i.e., $w(t) \in \mathcal{W}$ at all times t. To use a more concise notation, we write $w(\cdot) \in \mathcal{W}$. In addition, noisy measurements

$$y(t_k) = C_D x(t_k) + v(t_k)$$

are obtained at discrete sampling times $t_k = k\Delta t$ with $k \in \mathbb{N}$ and $\Delta t \in \mathbb{R}_{>0}$. Similar to the state disturbance, the unknown but bounded output disturbance $v(t_k)$ is contained within the output disturbance set $\mathcal{V} \subset \mathbb{R}^{n_y}$ at all sampling times t_k, i.e., $v(\cdot) \in \mathcal{V}$.

The disturbed system in (3) must satisfy hard constraints on the state and input of the form

$$x(\cdot) \in \mathcal{X} \tag{4a}$$

$$u(\cdot) \in \mathcal{U}, \tag{4b}$$

where $\mathcal{X} \subset \mathbb{R}^{n_x}$ and $\mathcal{U} \subset \mathbb{R}^{n_u}$ are the state and input constraint sets, respectively. Thus, the continuous-time state and input trajectories must always remain within $\mathcal{X}$ and $\mathcal{U}$, respectively.

To formulate a meaningful sampled-data control problem, we assume that (A_D, B_D, C_D) is stabilizable and detectable, where $A_D = e^{A\Delta t}$ and $B_D = \big(\int_0^{\Delta t} e^{At}\,\mathrm{d}t\big)B$. In addition, without loss of generality, we assume that $\mathcal{W}$, $\mathcal{V}$, $\mathcal{X}$, and $\mathcal{U}$ are zonotopes containing the origin. In this paper, the control goal is to steer the system in (3) to a neighborhood of the origin while minimizing a given cost function and satisfying the constraints in (4).

III. Reachability analysis

In this section, we present a clocked digital state estimator and controller. In addition, we propose an approach to compute reachable sets for times $t_k + t$ with $t \in [0, \Delta t)$ based on a noisy measurement obtained at t_k. Subsequently, we extend this approach to compute reachable sets for the entire time horizon, i.e., for any $t \in \mathbb{R}_{\geq 0}$.

A. State estimation and control

To obtain a discrete-time state estimator, we first exactly discretize the dynamics in (3), resulting in

$$x(t_{k+1}) = A_D x(t_k) + B_D u(t_k) + w_D(t_k),$$

where the discretized state disturbance w_D is bounded by the zonotope $\mathcal{W}_D \subset \mathbb{R}^{n_x}$ [28, Eq. (3.7)]. Instead of using a computationally demanding set-membership or strip-based observer [33], [34], we use a simple Luenberger state estimator

$$\hat{x}(t_{k+1}) = A_D\hat{x}(t_k) + B_D u(t_k) + L\big(y(t_k) - C_D\hat{x}(t_k)\big),$$

where $L \in \mathbb{R}^{n_x \times n_y}$ is a user-defined stabilizing output injection matrix such that all eigenvalues of $A_D - LC_D$ are contained in the open complex unit disc. As a result, the dynamics of the state estimation error $\epsilon(t_k) = x(t_k) - \hat{x}(t_k)$ and the corresponding error sets are

$$\begin{aligned} \epsilon(t_{k+1}) &= \big(A_D - LC_D\big)\epsilon(t_k) + w_D(t_k) - Lv(t_k) \\ \mathcal{E}(t_{k+1}) &= \big(A_D - LC_D\big)\mathcal{E}(t_k) \oplus \mathcal{W}_D \oplus (-L\mathcal{V}), \end{aligned}$$

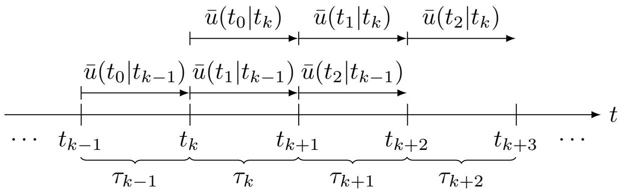

Fig. 1. Piecewise constant correction input trajectories $\bar{u}(\cdot|t_{k-1})$ and $\bar{u}(\cdot|t_k)$ for the prediction horizon $N = 3$.

where the initial error $\epsilon(t_0)$ lies within the zonotopic initial error set $\mathcal{E}(t_0) \subseteq \mathcal{X}$ containing the origin. Thus, $\epsilon(t_k) \in \mathcal{E}(t_k)$ for any $k \in \mathbb{N}$, resulting in $x(t_k) \in \hat{x}(t_k) \oplus \mathcal{E}(t_k)$ and $\hat{x}(t_k) \in x(t_k) \oplus \big(-\mathcal{E}(t_k)\big)$. We also want to mention that we do not require $\mathcal{E}(t_0)$ to be a robust positively invariant (RPI) set [19], [20], which would imply $\mathcal{E}(t_{k+1}) \subseteq \mathcal{E}(t_k)$ and significantly simplify the reachability analysis [13].

Based on the state estimate $\hat{x}(t_k)$ at t_k, our sampled-data controller provides a piecewise constant control signal [35]. Similar to [36], we use the simple, yet effective sampled-data control law

$$u(t) = \bar{u}(t_k) + K\hat{x}(t_k) \quad \text{for } t \in [t_k, t_{k+1}), \tag{5}$$

where $\bar{u}$ represents the piecewise constant correction input that is optimized online, and $K \in \mathbb{R}^{n_u \times n_x}$ is a user-defined stabilizing feedback matrix such that all eigenvalues of $A_D + B_D K$ are contained in the open complex unit disc.

In addition, $\bar{u}(\cdot|t_k)$ denotes the correction input trajectory that is optimized online starting at t_k based on $\hat{x}(t_k)$. In Fig. 1, we illustrate the piecewise constant correction input trajectories $\bar{u}(\cdot|t_{k-1})$ and $\bar{u}(\cdot|t_k)$. Analogous to the definition of $\bar{u}(\cdot|t_k)$, we denote the predictions of the future state, state estimate, and input trajectory based on $\hat{x}(t_k)$ by $x(\cdot|t_k)$, $\hat{x}(\cdot|t_k)$, and $u(\cdot|t_k)$.

B. First time interval

Based on the control law in (5), the input during $[t_k, t_{k+1})$ depends on the current state estimate $\hat{x}(t_k)$. To accommodate for the state-dependent control input in our set-based reachability analysis [28], [37], we define an augmented state $\tilde{x} \in \mathbb{R}^{n_x+n_u}$ [25]. The corresponding dynamics is

$$\underbrace{\begin{bmatrix} \dot{x}(t) \\ \dot{u}(t) \end{bmatrix}}_{\dot{\tilde{x}}(t)} = \underbrace{\begin{bmatrix} A & B \\ 0 & 0 \end{bmatrix}}_{\widetilde{A}} \underbrace{\begin{bmatrix} x(t) \\ u(t) \end{bmatrix}}_{\tilde{x}(t)} + \underbrace{\begin{bmatrix} w(t) \\ 0 \end{bmatrix}}_{\widetilde{w}(t)}, \tag{6}$$

where the piecewise constant control input $u(t)$ is given by (5). To project a zonotope of augmented states $\widetilde{\mathcal{Z}} \subset \mathbb{R}^{n_x+n_u}$ onto the space of nonaugmented states and inputs, we define the two matrices

$$\Pi_x = \begin{bmatrix} I & 0 \end{bmatrix} \in \mathbb{R}^{n_x \times (n_x+n_u)}$$
$$\Pi_u = \begin{bmatrix} 0 & I \end{bmatrix} \in \mathbb{R}^{n_u \times (n_x+n_u)},$$

where I denotes the identity matrix of appropriate dimension. For instance, the center and generator matrix of the zonotope $\Pi_x \left\langle c_{\widetilde{\mathcal{Z}}}, G_{\widetilde{\mathcal{Z}}} \right\rangle$ are obtained by deleting the last n_u rows of $c_{\widetilde{\mathcal{Z}}}$ and $G_{\widetilde{\mathcal{Z}}}$, respectively.

The solution of (6) at $t_k + t$ with $t \in [0, \Delta t)$ based on the augmented initial state $\tilde{x}(t_k)$ and state disturbance trajectory $w(\cdot)$ is denoted by $\tilde{\chi}\big(t, \tilde{x}(t_k), w(\cdot)\big) \in \mathbb{R}^{n_x+n_u}$. By considering an augmented initial state zonotope $\widetilde{\mathcal{Z}}_{\text{init}} \subset \mathbb{R}^{n_x+n_u}$ and the state disturbance set $\mathcal{W}$, the exact set of augmented states reachable at $t_k + t$ is

$$\widetilde{\mathcal{R}}_{\text{exact}}(t, \widetilde{\mathcal{Z}}_{\text{init}}, \mathcal{W}) = \big\{ \tilde{\chi}\big(t, \tilde{x}(t_k), w(\cdot)\big) \;\big|\; \tilde{x}(t_k) \in \widetilde{\mathcal{Z}}_{\text{init}}, \\ w(\cdot) \in \mathcal{W} \big\}.$$

Because reachable sets cannot be exactly computed for general linear systems [38], [39], we compute tight zonotopic over-approximations $\widetilde{\mathcal{R}}_{\text{over}} \supseteq \widetilde{\mathcal{R}}_{\text{exact}}$ according to [28]. Then, $x(t|t_k)$ is guaranteed to lie within $\Pi_x \widetilde{\mathcal{R}}_{\text{over}}(t, \widetilde{\mathcal{Z}}_{\text{init}}, \mathcal{W})$ for any $t \in [0, \Delta t)$ when choosing

$$\widetilde{\mathcal{Z}}_{\text{init}} = \left\langle \begin{matrix} 0 \\ \bar{u}(t_0|t_k) \end{matrix} \right\rangle \oplus \left\langle \begin{matrix} \hat{x}(t_k) \\ K\hat{x}(t_k) \end{matrix} \right\rangle \oplus \left\langle \begin{matrix} \mathcal{E}(t_k) \\ 0 \end{matrix} \right\rangle, \tag{7}$$

because of (2) and the following considerations: If the state estimate of the nonaugmented system in (3) at t_k is $\hat{x}(t_k)$, it follows that $x(t_k) \in \hat{x}(t_k) \oplus \mathcal{E}(t_k)$. In addition, the sampled-data input is $u(t|t_k) = \bar{u}(t_0|t_k) + K\hat{x}(t_k)$ based on (5).

C. Entire time horizon

Based on $\hat{x}(t_k)$, $\bar{u}(\cdot|t_k)$, and the previous considerations for $t_k + t$ with $t \in [0, \Delta t)$, the computation of the reachable set $\widetilde{\mathcal{R}}_{\text{hybrid}}$ for any $t_k + t$ with $t \in \mathbb{R}_{\geq 0}$ is presented in Alg. 1. Subsequently, this algorithm is briefly described.

Algorithm 1 Computation of the hybrid reachable set $\widetilde{\mathcal{R}}_{\text{hybrid}}\big(t, \hat{x}(t_k), \bar{u}(\cdot|t_k), \mathcal{W}, \mathcal{E}(\cdot)\big)$ for $t_k + t$ with $t \in \mathbb{R}_{\geq 0}$

```
Input: $t, \hat{x}(t_k), \bar{u}(\cdot|t_k), \mathcal{W}, \mathcal{E}(\cdot)$
Output: $\widetilde{\mathcal{Z}}_{\text{final}}$
1: $i \leftarrow 0$
2: $\widetilde{\mathcal{Z}}_{\text{init}} \leftarrow$ apply $\bar{u}(t_0|t_k), \hat{x}(t_k), \mathcal{E}(t_k)$ to (7)
3: while $t_{i+1} < t$ do
4:     $\mathcal{Z}_x \leftarrow \Pi_x \widetilde{\mathcal{R}}_{\text{over}}(\Delta t, \widetilde{\mathcal{Z}}_{\text{init}}, \mathcal{W})$
5:     $i \leftarrow i + 1$
6:     $\widetilde{\mathcal{Z}}_{\text{init}} \leftarrow \left\langle \begin{matrix} 0 \\ \bar{u}(t_i|t_k) \end{matrix} \right\rangle \oplus \left\langle \begin{matrix} \mathcal{Z}_x \\ K\mathcal{Z}_x \end{matrix} \right\rangle \oplus \left\langle \begin{matrix} 0 \\ -K\mathcal{E}(t_{k+i}) \end{matrix} \right\rangle$
7: end while
8: $\widetilde{\mathcal{Z}}_{\text{final}} \leftarrow \widetilde{\mathcal{R}}_{\text{over}}(t - t_i, \widetilde{\mathcal{Z}}_{\text{init}}, \mathcal{W})$
```

In line 2 of Alg. 1, we compute the augmented initial state set $\widetilde{\mathcal{Z}}_{\text{init}}$ according to (7). In lines 3 to 7, we iteratively compute reachable sets for steps of Δt until the specified time t is no longer greater than t_{i+1} for some $i \in \mathbb{N}$. The augmented initial state set $\widetilde{\mathcal{Z}}_{\text{init}}$ is updated in line 6 based on the following considerations: $\hat{x}(t_i|t_k)$ is contained in $x(t_i|t_k) \oplus \big(-\mathcal{E}(t_{k+i})\big)$ and $u(t_i|t_k)$ lies within $\bar{u}(t_i|t_k) \oplus K\big(x(t_i|t_k) \oplus \big(-\mathcal{E}(t_{k+i})\big)\big)$ for $i \in \mathbb{N}_{>0}$. Finally, in line 8, the reachable set for the specified time t is obtained.

Thus far, we have only considered reachable sets at certain points in time. Nonetheless, we must guarantee robust constraint satisfaction not only at, but also between two sampling times. Therefore, we also compute reachable sets for entire time intervals $\tau_i = [t_i, t_{i+1})$ with $i \in \mathbb{N}$. This set is given by

$$\begin{aligned}&\widetilde{\mathcal{R}}_{\text{hybrid}}\big(\tau_i, \hat{x}(t_k), \bar{u}(\cdot|t_k), \mathcal{W}, \mathcal{E}(\cdot)\big)\\ &\quad = \bigcup_{t\in\tau_i} \widetilde{\mathcal{R}}_{\text{hybrid}}\big(t, \hat{x}(t_k), \bar{u}(\cdot|t_k), \mathcal{W}, \mathcal{E}(\cdot)\big)\end{aligned}$$

and is computed according to [28, Sec. 3.2]. To use a more concise notation, we write

$$\begin{aligned}\mathcal{R}_x(\dots) &= \Pi_x \widetilde{\mathcal{R}}_{\text{hybrid}}(\dots)\\ \mathcal{R}_u(\dots) &= \Pi_u \widetilde{\mathcal{R}}_{\text{hybrid}}(\dots).\end{aligned}$$

In summary, we can efficiently compute the set of states and inputs that are reachable at arbitrary points in time and time intervals based on the state estimate $\hat{x}(t_k)$ at t_k. In the following section, we use these reachable set computations to propose an efficient robust output feedback MPC approach.

IV. Robust output feedback MPC

In this section, we present our robust output feedback dual-mode MPC algorithm considering a finite prediction horizon of $N \in \mathbb{N}_{>0}$. In the first mode, we iteratively solve an optimal control problem on a receding time horizon. We switch to the second mode when the state of the system is guaranteed to lie within a safe terminal set. In this mode, a corresponding safe terminal controller ensures robust constraint satisfaction at all future times.

First, we explicitly consider the computation time for solving the optimal control problem online. Second, we compute zonotopic safe terminal sets with corresponding safe terminal controllers offline. Third, after introducing terminal and contraction constraints, we state the optimal control problem, which is solved online. Finally, we present our robust output feedback dual-mode MPC algorithm and propose simplifications for its implementation.

A. Computation time consideration

To ensure the satisfaction of the state and input constraints in (4), we explicitly consider the nonzero computation time for solving the optimal control problem online [40]. At the sampling time $t_k = k\Delta t$, the available computation time for optimizing $\bar{u}(\cdot|t_{k-1})$ is used, as shown in Fig. 1. Then, we set

$$\bar{u}(t|t_k) = \bar{u}(t + t_1|t_{k-1}) \qquad \text{for } t \in [t_0, t_1), \tag{8}$$

and apply the input $\bar{u}(t_1|t_{k-1}) + K\hat{x}(t_k)$ to the system during τ_k while we optimize $\bar{u}(t|t_k)$ for $t \in [t_1, t_N)$. If the optimization solver requires a longer time than Δt to complete, we abort the optimization prematurely.

If the obtained correction input trajectory $\bar{u}(\cdot|t_k)$ is infeasible, we use the remainder of $\bar{u}(\cdot|t_{k-1})$ as a backup solution under the common assumption that an initial feasible solution $\bar{u}(\cdot|t_0)$ exists. In this case, we set

$$\bar{u}(t|t_k) = \begin{cases}\bar{u}(t + t_1|t_{k-1}) & \text{for } t \in [t_0, t_{N-1})\\ \emptyset & \text{for } t \in [t_{N-1}, t_N)\end{cases}. \tag{9}$$

If $\bar{u}(t_1|t_{k_\Omega-1})$ equals $\emptyset$ for some $k_\Omega \in \mathbb{N}$, we apply the inputs of the safe terminal control trajectory $\bar{u}_\Omega(\cdot|t_{k_\Omega})$ that guarantees robust constraint satisfaction at all times $t \geq t_{k_\Omega}$. In the following subsection, we present the construction of this safe terminal controller in more detail.

B. Safe terminal sets and controllers

Ensuring the satisfaction of the constraints in (4) is typically achieved using the terminal controller $u(t) = K\hat{x}(t_k)$ for $t \in \tau_k$ and constructing a polytopic terminal set that is an RPI set [19], [20]. However, underlying Minkowski additions and matrix transformations of polytopes become intractable for large systems.

Instead of constructing an RPI set, we have proposed an efficient state feedback control approach to compute large zonotopic safe terminal sets [41]. A set $\Omega(t_k) \subset \mathbb{R}^{n_x}$ is called a safe terminal set if a safe terminal control trajectory $\bar{u}_\Omega(\cdot|t_k)$ exists so that robust constraint satisfaction for an infinite time horizon is guaranteed in case $x(t_k) \in \Omega(t_k)$. Subsequently, we briefly present this two-step approach and state the main modifications that are required to incorporate output feedback control.

First, we compute a small safe terminal set $\underline{\Omega}(t_k)$ containing the origin, which is typically not invariant but can be safely steered into itself in finite time and is closely related to the minimal RPI set [42]. In particular, we determine some finite $i_k \in \mathbb{N}_{>0}$ so that $\mathcal{E}(t_{k+i_k}) \subseteq \mathcal{E}(t_k)$ and

$$\mathcal{R}_x\big(t_{i_k}, \underline{\Omega}(t_k) \oplus (-\mathcal{E}(t_k)), 0, \mathcal{W}, \mathcal{E}(\cdot)\big) \subseteq \underline{\Omega}(t_k) \tag{10a}$$

$$\mathcal{R}_x\big(\tau_i, \underline{\Omega}(t_k) \oplus (-\mathcal{E}(t_k)), 0, \mathcal{W}, \mathcal{E}(\cdot)\big) \subseteq \mathcal{X} \tag{10b}$$

$$\mathcal{R}_u\big(\tau_i, \underline{\Omega}(t_k) \oplus (-\mathcal{E}(t_k)), 0, \mathcal{W}, \mathcal{E}(\cdot)\big) \subseteq \mathcal{U}, \tag{10c}$$

where $\underline{\Omega}(t_k) = \mathcal{R}_x\big(t_{i_k}, \mathcal{X} \oplus (-\mathcal{E}(t_k)), 0, \mathcal{W}, \mathcal{E}(\cdot)\big)$ and $i \in \{0, 1, \dots, i_k - 1\}$. In contrast to RPI sets, $x(t_k + t)$ is allowed to lie outside $\underline{\Omega}(t_k)$ for some time $t \in \big(0, t_{i_k}\big)$. Nonetheless, (10b) and (10c) ensure robust constraint satisfaction during this period. Therefore, the state and input constraints in (4) are satisfied if $x(t_k) \in \underline{\Omega}(t_k)$ and $\hat{x}(t_k) \in \underline{\Omega}(t_k) \oplus (-\mathcal{E}(t_k))$, respectively.

Second, to increase the region of operation, we solve a convex optimization problem whose solution $\bar{u}_\Omega(\cdot|t_k)$ safely steers all states within the optimized large safe terminal set $\Omega(t_k)$ into $\underline{\Omega}(t_k)$. As a result, when using the safe terminal controller $\bar{u}_\Omega$ in case $x(t_k) \in \Omega(t_k)$, the satisfaction of the constraints in (4) is ensured. In the following subsections, we use these safe terminal sets and controllers to define terminal and contraction constraints.

C. Terminal constraint

Ideally, we want to add the terminal constraint

$$\mathcal{R}_x\big(t_N, \hat{x}(t_k), \bar{u}(\cdot|t_k), \mathcal{W}, \mathcal{E}(\cdot)\big) \subseteq \Omega(t_{k+N}) \tag{11}$$

to the optimal control problem solved during τ_k. Although the zonotope containment condition in (1) provides a way to solve (11) for large systems, it is computationally too expensive for most real-time applications. Thus, to speed up online computations, we decompose (1) into two linear programming problems. The first one is solved offline, and the second one is added to the constraints of the optimal control problem.

First, based on $\Omega(t_{k+N}) = \langle c_{\Omega(t_{k+N})}, G_{\Omega(t_{k+N})} \rangle$ and the reachable set $\mathcal{R}_x(t_N, 0, 0, \mathcal{W}, \mathcal{E}(\cdot)) = \langle c_{\mathcal{R}_{k+N,1}}, G_{\mathcal{R}_{k+N,1}} \rangle$, we solve the convex optimization problem

$$\begin{aligned} \underset{\Gamma_{k+N}}{\text{minimize}} \quad & \|\Gamma_{k+N}\|_\infty && \text{(12a)} \\ \text{subject to} \quad & G_{\mathcal{R}_{k+N,1}} = G_{\Omega(t_{k+N})}\Gamma_{k+N} && \text{(12b)} \\ & \|\Gamma_{k+N}\|_\infty \leq 1 && \text{(12c)} \end{aligned}$$

offline, where Γ_{k+N} is a matrix of appropriate dimensions. Second, we add the terminal constraint

$$\begin{aligned} c_{\Omega(t_{k+N})} - (c_{\mathcal{R}_{k+N,1}} + c_{\mathcal{R}_{k+N,2}}) = G_{\Omega(t_{k+N})}\gamma_{k+N} && \text{(13a)} \\ \| \begin{bmatrix} \Gamma^*_{k+N} & \gamma_{k+N} \end{bmatrix} \|_\infty \leq 1 && \text{(13b)} \end{aligned}$$

to the optimal control problem, where $\langle c_{\mathcal{R}_{k+N,2}}, 0 \rangle = \mathcal{R}_x(t_N, \hat{x}(t_k), \bar{u}(\cdot|t_k), 0, 0)$, Γ^*_{k+N} denotes the solution of (12), and γ_{k+N} is an optimization vector of appropriate dimension. Based on the superposition principle and (1), it follows that the condition in (11) is satisfied if (13) is feasible.

D. Contraction constraint

Inspired by [36], we introduce a simple contraction constraint such that convergence to a neighborhood of the origin in finite time is ensured. Before we present this constraint, we introduce the distance

$$\|\mathcal{Z}\| = \min \{\delta \in \mathbb{R}_{\geq 0} \mid \mathcal{Z} \subseteq \delta \langle 0, I \rangle \} \tag{14}$$

between the origin and zonotope $\mathcal{Z}$, where $\langle 0, I \rangle$ is the unit ball corresponding to the infinity norm. To account for different orders of magnitude of the state space dimensions, they can also be weighted accordingly, e.g., using a suitable diagonal matrix instead of I in (14).

Based on the contraction distances

$$d(t_i|t_k) = \alpha + \left\| \mathcal{R}_x(t_i, \hat{x}(t_k), \bar{u}(\cdot|t_k), \mathcal{W}, \mathcal{E}(\cdot)) \right\|,$$

where $\alpha \in \mathbb{R}_{>0}$ is a user-defined contraction parameter that is typically chosen close to zero, we add the contraction constraint

$$\sum_{i=1}^{N-1} d(t_i|t_k) - \sum_{i=1}^{N-1} d(t_i|t_{k-1}) < -\alpha \tag{15}$$

to the optimal control problem solved during τ_k. Thus, (15) ensures that $\sum_{i=1}^{N-1} d(t_i|t_k)$ is a strictly decreasing function with respect to consecutive time steps k, implying the convergence of the state trajectory $x(\cdot)$ to the origin.

As stated in Section IV-A, we use the remainder of the previous solution $\bar{u}(\cdot|t_{k-1})$ as a backup if the optimal control problem is infeasible. To be consistent with the update of $\bar{u}(\cdot|t_k)$ in (9), we set

$$d(t_i|t_k) = \begin{cases} d(t_{i+1}|t_{k-1}) & \text{for } i \in \{1, 2, \ldots, N-2\} \\ 0 & \text{for } i = N-1 \end{cases} \tag{16}$$

if the optimal control problem solved during τ_k is infeasible.

E. Optimal control problem

By combining the presented constraints, the online optimal control problem solved during the time interval τ_k is

$$\begin{aligned} \underset{\bar{u}(\cdot|t_k)}{\text{minimize}} \quad & J(\hat{x}(t_k), \bar{u}(\cdot|t_k)) && \text{(17a)} \\ \text{subject to} \quad & \text{(8), (13), and (15) are satisfied} && \text{(17b)} \\ & \text{for } i \in \{0, 1, \ldots, N-1\}: \\ & \mathcal{R}_x(\tau_i, \hat{x}(t_k), \bar{u}(\cdot|t_k), \mathcal{W}, \mathcal{E}(\cdot)) \subseteq \mathcal{X} && \text{(17c)} \\ & \mathcal{R}_u(\tau_i, \hat{x}(t_k), \bar{u}(\cdot|t_k), \mathcal{W}, \mathcal{E}(\cdot)) \subseteq \mathcal{U}, && \text{(17d)} \end{aligned}$$

where J is a convex cost function. Because (17) is a convex optimization problem with linear constraints, it can be solved efficiently by existing convex optimization algorithms [43]. To reduce online computational effort [7], [11], we tighten the constraint sets $\mathcal{X}$ and $\mathcal{U}$ in (17c) and (17d) based on the superposition principle because the time-invariant reachable sets corresponding to $\mathcal{W}$ and $\mathcal{E}(\cdot)$ can be computed offline.

F. Algorithm

Finally, we present our robust output feedback dual-mode MPC approach in Alg. 2, where we iteratively solve the optimal control problem in (17) in a moving horizon fashion until the safe terminal controller $\bar{u}_\Omega$ takes over. Subsequently, we briefly describe this algorithm.

Algorithm 2 Robust output feedback dual-mode MPC

```
 1: $k \leftarrow 1$
 2: while $d(t_1|t_{k-1}) \not\leq \alpha$ do                                ▷ 1. mode
 3:     $u(t) \leftarrow \bar{u}(t_1|t_{k-1}) + K\hat{x}(t_k)$ for $t \in \tau_k$
 4:     $(\bar{u}(\cdot|t_k), d(\cdot|t_k), r_{\text{infeasible}}) \leftarrow$ solve (17)
 5:     if $r_{\text{infeasible}}$ then
 6:         $\bar{u}(\cdot|t_k) \leftarrow$ apply $\bar{u}(\cdot|t_{k-1})$ to (9)
 7:         $d(\cdot|t_k) \leftarrow$ apply $d(\cdot|t_{k-1})$ to (16)
 8:     end if
 9:     $k \leftarrow k + 1$
10: end while
11: $k_\Omega \leftarrow k$
12: while true do                                                   ▷ 2. mode
13:     $u(t) \leftarrow \bar{u}_\Omega(t_k - t_{k_\Omega}|t_{k_\Omega}) + K\hat{x}(t_k)$ for $t \in \tau_k$
14:     $k \leftarrow k + 1$
15: end while
```

In line 2 of Alg. 2, we check if the unknown state $x(t_k)$ is guaranteed to lie within $\Omega(t_k)$ at t_k. If this is the case, we switch to the second mode and apply the inputs of the safe terminal controller $\bar{u}_\Omega$ to the system in lines 12 to 15. Otherwise, the input that is applied to the system during τ_k is updated in line 3 based on Section IV-A. In addition, the optimal control problem in (17) is solved until t_{k+1} in line 4. Its third return value $r_{\text{infeasible}}$ is a Boolean flag that is true if the optimal control problem was infeasible; otherwise, it is false. In case of an infeasible solution, $\bar{u}(\cdot|t_k)$ and $d(\cdot|t_k)$ are updated based on the feasible backup solution of the previous time step in lines 6 and 7.

Theorem 1: If an initial feasible solution $\bar{u}(\cdot|t_0)$ with corresponding $d(\cdot|t_0)$ and $\sum_{i=1}^{N-1} d(t_i|t_{-1}) = \infty$ is given at

t_1, Alg. 2 steers the disturbed system in (3) to a neighborhood of the origin while satisfying the constraints in (4). □

Proof: We must show two things: (i) The constraints in (4) are satisfied for an infinite time horizon, and (ii) the system reaches a neighborhood of the origin in finite time.

(i) This part of the proof is mainly based on extending the optimized correction input trajectory by the safe terminal control trajectory and using the previous solution as a backup in case of the infeasibility of (17). Because these ideas follow standard robust dual-mode MPC approaches [36], this part of the proof is omitted.

(ii) Based on Section IV-B, the safe terminal control trajectory $\bar{u}_\Omega(\cdot|t_k)$ steers the system to a neighborhood of the origin if $x(t_k) \in \Omega(t_k)$ with $k \in \mathbb{N}$. If

$$d(t_1|t_{k-1}) \leq \alpha, \tag{18}$$

which is the condition in Alg. 2 when we switch to use $\bar{u}_\Omega(\cdot|t_k)$, $x(t_k)$ is guaranteed to be the origin or lie within $\Omega(t_k)$ based on (14), (16), and the terminal constraint in Section IV-C. Thus, it remains to show that (18) is satisfied for some finite k.

If the optimal control problem in (17) solved during τ_k is feasible, we know that (15) is satisfied, i.e., the contraction rate of at least α is guaranteed for $\sum_{i=1}^{N-1} d(t_i|t_k)$. If it is infeasible, $d(\cdot|t_k)$ is updated according to (16), i.e., it is set equal to the remainder of the previous contraction distance trajectory $d(\cdot|t_{k-1})$. As a result, the contraction constraint in (15) is simplified to $-d(t_1|t_{k-1}) < -\alpha$. This inequality holds because the condition in line 2 of Alg. 2 would have shown that $x(t_k) \in \Omega(t_k)$. Thus, the contraction rate of α is also guaranteed in case of the infeasibility of (17). Therefore, (18) is satisfied for some finite k. ■

G. Simplifications

As mentioned in Section III-A, we do not require the initial error set $\mathcal{E}(t_0)$ to be an RPI set [19], [20], which would imply $\mathcal{E}(t_{i+1}) \subseteq \mathcal{E}(t_i)$ for any $i \in \mathbb{N}$ [13]. To avoid computing an infinite number of error sets, we exploit the fact that the set sequence $\mathcal{E}(\cdot)$ typically quickly converges to the minimal RPI set $\mathcal{E}(t_\infty)$, which can be tightly over-approximated by another RPI set $\mathcal{E}_\infty$ [44]. Thus, we can use $(1+\beta)\mathcal{E}_\infty$ as over-approximation for all $\mathcal{E}\left(t_{k+i}\right)$ with $i \in \mathbb{N}$ by computing the smallest $\beta \in \mathbb{R}_{\geq 0}$ such that $\mathcal{E}\left(t_k\right) \subseteq (1+\beta)\mathcal{E}_\infty$ for some given $k \in \mathbb{N}$. Therefore, only a finite number of error sets must be computed.

Similarly, it suffices to compute a finite number of safe terminal sets. In particular, we replace $\Omega\left(t_{k+i}\right)$ with $\Omega\left(t_k\right)$ for all $i \in \mathbb{N}$. It is sufficient to compute only $\Omega(t_N)$ when choosing $k = N$ and remaining in the first mode for the first N time steps in Alg. 2. In the following section, we use these simplifications to solve a benchmark problem from the literature.

V. Numerical Example

With the next release of our MATLAB controller synthesis toolbox AROC [45], which uses our reachability analysis toolbox CORA [37] to compute reachable sets, we will make our robust output feedback MPC algorithm public. All optimization problems are modeled using YALMIP [46] with the parameter "allownonconvex" set to 0 and solved using MOSEK [47] with default parameters. All computations are performed on a laptop with an Intel Core i7-1185G7 and 32 GB memory.

TABLE I

State, input, and disturbance bounds

$e^{(1)}, e^{(2)}, e^{(3)}$	[–10, 10] m
$\dot{e}^{(1)}, \dot{e}^{(2)}, \dot{e}^{(3)}$	[5, 5] $\frac{\mathrm{m}}{\mathrm{s}}$
$a^{(1)}, a^{(2)}, a^{(3)}$	[–8, 8] $\frac{\mathrm{m}}{\mathrm{s}^2}$
$u^{(1)}, u^{(2)}, u^{(3)}$	[–8, 8] $\frac{\mathrm{m}}{\mathrm{s}^2}$
$a^{(0)}$	[–1, 1] $\frac{\mathrm{m}}{\mathrm{s}^2}$
$v^{(1)}, v^{(3)}, v^{(5)}$	[–0.05, 0.05] m
$v^{(2)}, v^{(4)}, v^{(6)}$	[–0.05, 0.05] $\frac{\mathrm{m}}{\mathrm{s}}$
$\epsilon^{(1)}(t_0), \epsilon^{(4)}(t_0), \epsilon^{(7)}(t_0)$	[–0.5, 0.5] m
$\epsilon^{(2)}(t_0), \epsilon^{(5)}(t_0), \epsilon^{(8)}(t_0)$	[–0.5, 0.5] $\frac{\mathrm{m}}{\mathrm{s}}$
$\epsilon^{(3)}(t_0), \epsilon^{(6)}(t_0), \epsilon^{(9)}(t_0)$	[–0.5, 0.5] $\frac{\mathrm{m}}{\mathrm{s}^2}$

In this section, we show the effectiveness of our control approach on the vehicle platooning benchmark in [48]. The dynamics corresponding to the relative motion of the i^{th} following vehicle with $i \in \{1, 2, 3\}$ and its vehicle ahead is

$$\ddot{e}^{(i)} = a^{(i-1)} - a^{(i)},$$

where the relative position error $e^{(i)}$ denotes the difference between the two vehicles and a given safe reference distance. In addition, $a^{(i)}$ corresponds to the i^{th} effective acceleration described by the drivetrain dynamics

$$\dot{a}^{(i)} = -\frac{1}{T_i} a^{(i)} + \frac{1}{T_i} u^{(i)},$$

where T_i represents a time constant that is assumed to be 0.5 s for all $i \in \{1, 2, 3\}$ and $u^{(i)}$ is the i^{th} control input. In addition, the acceleration of the leading vehicle $a^{(0)}$ is assumed to be an unknown but bounded additive disturbance. In summary, the state of the platoon is described by $x = \begin{bmatrix} e^{(1)} & \dot{e}^{(1)} & a^{(1)} & e^{(2)} & \dot{e}^{(2)} & a^{(2)} & e^{(3)} & \dot{e}^{(3)} & a^{(3)} \end{bmatrix}^T$, the control input is $u = \begin{bmatrix} u^{(1)} & u^{(2)} & u^{(3)} \end{bmatrix}^T$, and the state disturbance is $w = \begin{bmatrix} 0 & a^{(0)} & 0 & \dots & 0 \end{bmatrix}^T$. Because vehicle-to-vehicle communication is assumed, a central controller with stabilizing state feedback matrix K can be designed [48].

The output matrix $C_D \in \mathbb{R}^{6 \times 9}$ is chosen to be an identity matrix, where every third row is eliminated, i.e., the effective accelerations of all three following vehicles cannot be measured directly. Thus, the output disturbance is $v = \begin{bmatrix} v^{(1)} & v^{(2)} & \dots & v^{(6)} \end{bmatrix}^T$. In addition, the state, input, disturbance, and initial state estimation error bounds are shown in Table I.

To enable short solver computation times, we use the quadratic cost function $J = \sum_{i=1}^{N-1} L\big(\bar{x}(t_i|t_k), \bar{u}(t_i|t_k)\big) + V\big(\bar{x}(t_N|t_k)\big)$ in (17a), where $L\big(\bar{x}, \bar{u}\big) = \bar{x}^T \bar{x} + 10 \bar{u}^T \bar{u}$, $V(\bar{x}) = \bar{x}^T \bar{x}$, and $\bar{x}(t_i|t_k)$ equals $\mathcal{R}_x\big(t_i, \hat{x}(t_k), \bar{u}(\cdot|t_k), 0, 0\big)$. Thus, the convex optimization problem in (17) is a quadratic program [43]. The prediction horizon is $N = 20$, and the sampling time is $\Delta t = 150\,\mathrm{ms}$. The applied correction input during τ_0 is 0, the contraction parameter used in (15) is $\alpha = 10^{-3}$, and the stabilizing output injection matrix L is obtained by [49, Eq. 16]. In addition, the unknown nonaugmented initial system state is given by $x(t_0) = \begin{bmatrix} -9\,\mathrm{m} & 4\,\frac{\mathrm{m}}{\mathrm{s}} & 7\,\frac{\mathrm{m}}{\mathrm{s}^2} & 9\,\mathrm{m} & -4\,\frac{\mathrm{m}}{\mathrm{s}} & 7\,\frac{\mathrm{m}}{\mathrm{s}^2} & 3\,\mathrm{m} & 3\,\frac{\mathrm{m}}{\mathrm{s}} & 0 \end{bmatrix}^T$.

In Fig. 2, we show projections of sets and random trajectories onto relative position and velocity error dimensions. In addition, we visualize both trajectories $\bar{u}(\cdot)$ and $u(\cdot)$ in Fig. 3. Because the maximum computation time for solving the optimization problem in (17) is always less than Δt, we never have to abort the online optimization prematurely.

In comparison, more than 5 min are required to perform a single Minkowski addition of two nine-dimensional axis-aligned unit boxes represented in halfspace representation using the toolbox MPT3 [50]. Since this computation corresponds to the simplest case possible, it is obvious that polytopic methods are unsuitable for handling large systems.

VI. Conclusions and Future Work

We have proposed a robust output feedback dual-mode MPC approach for linear sampled-data systems. Using scalable reachability analysis and convex optimization algorithms, zonotopic safe terminal sets that ensure robust constraint satisfaction at all times after they are entered are obtained. Based on these sets, real-time controllers that steer the system to a neighborhood of the origin despite unknown but bounded disturbances are computed. In contrast to existing work, a sampling time of 150 ms is achieved for a nine-dimensional system without exploiting the sparsity of the system matrix. In the future, we plan to apply our scalable methods to learning-based MPC.

References

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

[2] S. J. Qin and T. A. Badgwell, "A survey of industrial model predictive control technology," *Control Engineering Practice*, vol. 11, no. 7, pp. 733–764, 2003.

[3] F. Borrelli, A. Bemporad, and M. Morari, *Predictive Control for Linear and Hybrid Systems*, 1st ed. Cambridge University Press, 2017.

[4] S. V. Raković and W. S. Levine, Eds., *Handbook of Model Predictive Control*, 1st ed., ser. Control Engineering. Birkhäuser, 2019.

[5] J. H. Lee and Z. Yu, "Worst-case formulations of model predictive control for systems with bounded parameters," *Automatica*, vol. 33, no. 5, pp. 763–781, 1997.

[6] P. O. M. Scokaert and D. Q. Mayne, "Min-max feedback model predictive control for constrained linear systems," *IEEE Transactions on Automatic Control*, vol. 43, no. 8, pp. 1136–1142, 1998.

[7] L. Chisci, J. A. Rossiter, and G. Zappa, "Systems with persistent disturbances: Predictive control with restricted constraints," *Automatica*, vol. 37, no. 7, pp. 1019–1028, 2001.

[8] D. Q. Mayne, M. M. Seron, and S. V. Raković, "Robust model predictive control of constrained linear systems with bounded disturbances," *Automatica*, vol. 41, no. 2, pp. 219–224, 2005.

[9] R. Gonzalez, M. Fiacchini, T. Alamo, J. L. Guzman, and F. Rodríguez, "Online robust tube-based MPC for time-varying systems: A practical approach," *International Journal of Control*, vol. 84, no. 6, pp. 1157–1170, 2011.

[10] M. N. Zeilinger, D. M. Raimondo, A. Domahidi, M. Morari, and C. N. Jones, "On real-time robust model predictive control," *Automatica*, vol. 50, no. 3, pp. 683–694, 2014.

[11] A. Richards and J. How, "Robust stable model predictive control with constraint tightening," in *American Control Conference*, 2006, pp. 1557–1562.

[12] S. V. Raković, B. Kouvaritakis, M. Cannon, C. Panos, and R. Findeisen, "Parameterized tube model predictive control," *IEEE Transactions on Automatic Control*, vol. 57, no. 11, pp. 2746–2761, 2012.

[13] D. Q. Mayne, S. V. Raković, R. Findeisen, and F. Allgöwer, "Robust output feedback model predictive control of constrained linear systems: Time varying case," *Automatica*, vol. 45, no. 9, pp. 2082–2087, 2009.

[14] M. Kögel and R. Findeisen, "Robust output feedback MPC for uncertain linear systems with reduced conservatism," in *IFAC World Congress*, 2017, pp. 10 685–10 690.

[15] F. D. Brunner, M. A. Müller, and F. Allgöwer, "Enhancing output-feedback MPC with set-valued moving horizon estimation," *IEEE Transactions on Automatic Control*, vol. 63, no. 9, pp. 2976–2986, 2018.

[16] M. Farina and R. Scattolini, "Tube-based robust sampled-data MPC for linear continuous-time systems," *Automatica*, vol. 48, no. 7, pp. 1473–1476, 2012.

[17] F. Blanchini, D. Casagrande, G. Giordano, and U. Viaro, "Robust constrained model predictive control of fast electromechanical systems," *Journal of the Franklin Institute*, vol. 353, no. 9, pp. 2087–2103, 2016.

[18] F. A. C. C. Fontes, S. V. Raković, and I. V. Kolmanovsky, "Rigid tube model predictive control for linear sampled-data systems," in *IFAC World Congress*, 2017, pp. 9840–9845.

[19] 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.

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

[21] P. D. Christofides, R. Scattolini, D. Muñoz de la Peña, and J. Liu, "Distributed model predictive control: A tutorial review and future research directions," *Computers and Chemical Engineering*, vol. 51, pp. 21–41, 2013.

[22] J. M. Maestre and R. R. Negenborn, *Distributed model predictive control made easy*, 1st ed., ser. Intelligent Systems, Control and Automation: Science and Engineering. Springer, 2014, vol. 69.

[23] A. A. Kurzhanskiy and P. Varaiya, "Ellipsoidal techniques for reachability analysis of discrete-time linear systems," *IEEE Transactions on Automatic Control*, vol. 52, no. 1, pp. 26–38, 2007.

[24] 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.

[25] 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.

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

[27] A. Chutinan and B. H. Krogh, "Computational techniques for hybrid system verification," *IEEE Transactions on Automatic Control*, vol. 48, no. 1, pp. 64–75, 2003.

[28] M. Althoff, "Reachability analysis and its application to the safety assessment of autonomous cars," Doctoral dissertation, Technical University of Munich, 2010. [Online]. Available: https://mediatum.ub.tum.de/doc/1287517/

[29] ——, "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.

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

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

[32] A. Kulmburg and M. Althoff, "On the co-NP-completeness of the zonotope containment problem," *European Journal of Control*, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0947358021000856

[33] V. T. H. Le, C. Stoica, T. Alamo, E. F. Camacho, and D. Dumur, "Zonotopic guaranteed state estimation for uncertain systems," *Automatica*, vol. 49, no. 11, pp. 3418–3424, 2013.

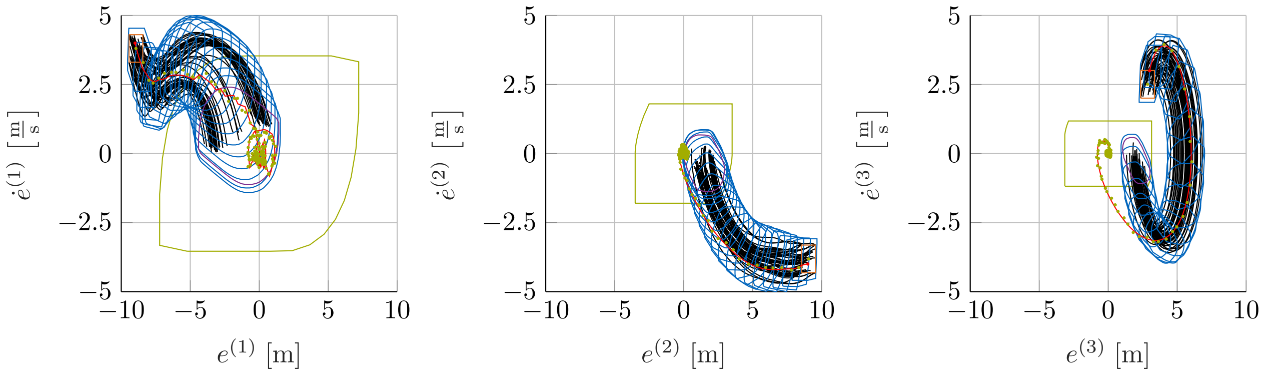

Fig. 2. Projections of sets and trajectories onto relative position and velocity error dimensions. The initial reachable sets corresponding to $\bar{u}(\cdot|t_0)$ for all N time intervals and for the last time point are shown in blue and purple, respectively. In addition, 50 random trajectories for the initial solution are shown in black, and the actual trajectory for $t \in [0, 20]$ s is shown in red. Moreover, the red and green dots represent the unknown initial state $x(t_0)$ and the state estimates along the actual trajectory, respectively. In addition, the orange and green zonotopes correspond to the initial error set $\mathcal{E}(t_0)$ and the safe terminal set $\Omega(t_N)$.

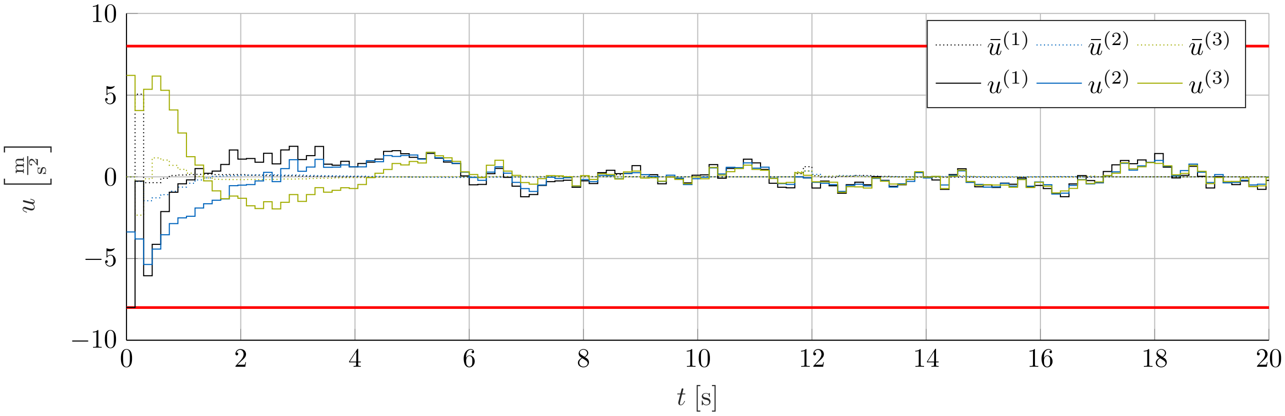

Fig. 3. Input trajectories. In addition, the bounds of the input constraint set $\mathcal{U}$ are shown in red. At time step $k = 78$, i.e., at $t = 11.7$, our robust output feedback dual-mode MPC controller switches to the second mode, which ensures robust constraint satisfaction at all future times.

[34] M. Althoff and J. J. Rath, “Comparison of guaranteed state estimators for linear time-invariant systems,” *Automatica*, vol. 130, p. 109662, 2021.

[35] F. H. Clarke, Y. S. Ledyaev, E. D. Sontag, and A. I. Subbotin, “Asymptotic controllability implies feedback stabilization,” *IEEE Transactions on Automatic Control*, vol. 42, no. 10, pp. 1394–1407, 1997.

[36] J. M. Bravo, T. Alamo, and E. F. Camacho, “Robust MPC of constrained discrete-time nonlinear systems based on approximated reachable sets,” *Automatica*, vol. 42, no. 10, pp. 1745–1751, 2006.

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

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

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

[40] 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.

[41] 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.

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

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

[44] 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.

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

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

[47] MOSEK Aps, “The MOSEK optimization toolbox for MATLAB manual. Version 9.2,” 2021. [Online]. Available: https://docs.mosek.com/9.2/toolbox/index.html

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

[49] W. Tang, Z. Wang, Y. Wang, T. Raissi, and Y. Shen, “Interval estimation methods for discrete-time linear time-invariant systems,” *IEEE Transactions on Automatic Control*, vol. 64, no. 11, pp. 4717–4724, 2019.

[50] M. Herceg, M. Kvasnica, C. N. Jones, and M. Morari, “Multi-parametric toolbox 3.0,” in *European Control Conference*, 2013, pp. 502–510.