Nonlinear Analysis: Hybrid Systems 36 (2020) 100854

Contents lists available at ScienceDirect

Nonlinear Analysis: Hybrid Systems

journal homepage: www.elsevier.com/locate/nahs

Finite horizon constrained control and bounded-error estimation in the presence of missing data ☆

Kwesi Rutledge [a,*], Sze Zheng Yong [b], Necmiye Ozay [a]

[a] Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor, MI 48109, USA
[b] School for Engineering of Matter, Transport and Energy, Arizona State University, Tempe, AZ 85287, USA

ARTICLE INFO

Article history:
Received 12 February 2019
Received in revised form 9 August 2019
Accepted 11 December 2019
Available online 28 January 2020

Keywords:
Robust estimators
Bounded-error estimation
Missing data
Invariance control

ABSTRACT

In this paper, we propose an optimization-based design technique for constrained control and bounded-error state estimation for affine systems in the presence of intermittent measurements. We treat the affine system as a switched system where the measurement equation switches between two modes based on whether a measurement exists or is missing, and model potential missing data patterns with a finite-length language that constrains the feasible mode sequences. Then, we introduce a novel property, equalized recovery, that generalizes the equalized performance property and that allows us to tolerate missing observations. By utilizing Q-parametrization, we show that a finite horizon optimal estimator/controller can be constructed using time-based and prefix-based approaches, where the latter implicitly estimates the specific missing data pattern (i.e., mode sequence), within the given language, according to the prefix observed so far. We illustrate with numerical examples that the proposed approaches can provide desirable performance guarantees.

1. Introduction

Feedback control and state estimation are central to enabling advanced and smart technologies for dynamical systems such as autonomous vehicles, industrial robots, etc. While controller and estimator design techniques are abundant, many of them rely on the common assumption that regularly sampled sensor measurements are reliably available. However, as systems become increasingly integrated and distributed, such access to sensory data can no longer be taken for granted in the control and decision-making loops of the system. For example, in networked control systems where sensory data is transmitted over unreliable communication networks, measurement packets may be dropped [1,2] or they can experience denial of service attacks [3], whereas for many autonomous systems that rely on a myriad of sensors and perception algorithms for state information such as position and velocity, said information may be temporarily missing due to sensor glitches, occlusions, or classification errors in the perception algorithms [4]. Therefore, there is a need for control and estimation algorithms that are robust to missing data.

Safety-criticality of these systems also require certain invariance properties in the form of state constraints that hold indefinitely. These state constraints can take the form of a small enough tracking error in control problems [5] or small enough estimation error in estimation problems [6], where exceeding a certain error bound is deemed unsafe. Invariance

☆ No author associated with this paper has disclosed any potential or pertinent conflicts which may be perceived to have impending conflict with this work. For full disclosure statements refer to https://doi.org/10.1016/j.nahs.2020.100854.

* Corresponding author.
E-mail address: krutledg@umich.edu (K. Rutledge).

https://doi.org/10.1016/j.nahs.2020.100854
1751-570X/

becomes particularly challenging when the controllers and estimators do not have regular access to measurements. The goal of this paper is to develop a methodology for the design of controllers and estimators that guarantee that certain state constraints are satisfied even in the face of missing measurement events. Our key insight is to relax invariance conditions by allowing the system or error states to satisfy a slightly larger bound during missing data events as long as the safety constraints are not violated and the states come back to the original level at the end. We call this property *equalized recovery*, as a generalization of equalized performance proposed in the context of bounded error estimation that requires guaranteeing a uniform bound [6].

In particular, we propose an optimization-based method to synthesize controllers and estimators that provide equalized recovery guarantees for discrete-time linear systems with intermittent measurements. We model the system with intermittent measurements as a switched system with the mode signal representing whether the measurement is available or missing. Instead of the commonly used probabilistic models for intermittent measurements [2], we model the feasible missing data patterns using a finite language, where each word in the language corresponds to a potential mode signal. At run-time, although we know that the mode signal belongs to this finite language, we do not know which word in the language is being realized since the missing data pattern is observed online. Earlier work [7], defines a worst-case word for the given language and designs an estimator that is robust against this worst-case missing data pattern. Here we propose to adapt the filter/controller gains based on the prefix of the missing data pattern observed thus far; hence, significantly improving achievable recovery levels. The design of these controller/estimator gains are enabled by Q-parametrization, which is a technique used to recast the optimal control design for affine systems as a convex programming problem [8]. Although, in general, imposing additional structure on filter/controller gains in Q-parametrization-based design leads to non-convex problems, one of our main contributions is to show that the structure imposed by prefix dependency of the gains still leads to a convex problem. Therefore, the proposed controllers and estimators not only provide significantly improved recovery levels, but also can be synthesized efficiently.

Additionally, we investigate the relationship between equalized recovery for estimation problems and detectability for linear systems with data loss as presented in [9], and show that equalized recovery is slightly more general than detectability. Finally, the effectiveness of our proposed estimator and controller designs are demonstrated in simulation on several autonomous driving scenarios, including adaptive cruise control and lane keeping. We also use a formation control problem with multiple agents to demonstrate scalability.

Preliminary results for designing equalized recovery estimators have been presented in [7,10], where the former paper considers the worst-case language and the latter introduces a prefix-based approach to reduce conservativeness. The current paper combines the results in [7,10] and extends them to control synthesis problems with missing data, while providing a more comprehensive treatment of the problem together with full proofs. The rest of the paper is organized as follows. We start by presenting some notation and preliminary results in Section 2. The problems under consideration are formally stated in Section 3. Section 4 states the main results on how to synthesize estimators and controllers that satisfy the required equalized recovery levels. Some implementation details and connections to detectability/stabilizability are given in Section 5, while illustrative examples are given in Section 6 before the paper is concluded in Section 7. Most of the proofs are deferred to the Appendix to ease the exposition.

1.1. Literature review

Control and estimation problems for systems subject to missing data or intermittent measurements have been extensively considered in the context of networked control systems [1,2,11] and more recently for security problems involving denial of service attacks [3]. For missing and intermittent observations modeled by probability distributions, extensions of the Kalman filter have been proposed (e.g., [2,12,13]) to estimate the system state. Similarly, in this setting of probabilistic data loss models, stabilizing controllers or controllers minimizing a quadratic cost have been studied (e.g., [1,14,15]). Nonetheless, the probabilistic nature of these approaches is not directly applicable for safety-critical applications, where hard bounds on the estimation or tracking errors are often necessary.

Another approach for modeling missing data patterns is to use a characterization of the set of all plausible missing data patterns. A simple characterization of this sort is the so-called (m, k) firmness [16,17] that indicates that for any k consecutive measurements, at least m are available. A more general set description can be obtained using automata [9] or finite languages [7,10,18] to represent the set of all feasible missing data patterns. Jungers et al. [9] study observability and controllability like properties for discrete-time linear systems subject to data loss, and characterize conditions on the system and the automata representing the missing data pattern for these properties to hold. On the other hand, the controller and estimator design problem is not considered in [9] and theoretical analysis focuses on the idealized setting with no process or measurement noise. In this paper, we use the language-based representation as in our earlier work [7,10,18] and focus on control and estimator designs, which guarantee constraints on the states or errors are satisfied, for systems subject to missing measurements and various types of noise. Another related, yet complementary line of work is on designing measurement schedules (i.e., when to measure and when not to measure if there is a budget on the number of measurements) together with controls [19,20], which differs from our setting in that we assume the missing measurements are chosen adversarially from the set of feasible patterns.

As we are interested in enforcing constraints on the control or estimation error, our results are related to disturbance rejection, set-valued observers, or ℓ_∞ filtering [6,21–24], though these approaches cannot readily handle missing data. Of

particular interest to our approach is an intuitive property for state estimators called equalized performance that ensures that the estimation error does not increase at each step (e.g., [6,25]). Instead, we allow a bounded increase in the error during missing data events as long as the error recovers back to its original level at the end.

From a computational standpoint, our controller and estimator synthesis approach builds upon Q-parametrization to reduce the non-convex measurement feedback controller/estimator design problem to a convex optimization form via a nonlinear change of variables [8]. In particular, we restrict the controllers and design variables in the estimators to be affine in the measurements with memory (i.e., we allow the current action to depend on past measurements). However, the main difference is that, instead of memory depending only on the output measurement history, the memory of the controllers and filters we design also depends on the prefix of the missing data pattern seen so far (i.e., on the discrete-state history), which results in significantly improved performance. As an alternative to measurement feedback one can use disturbance feedback [3,18], which is equivalent to measurement feedback only in special cases [26]. Existing work using disturbance feedback with potentially missing data [3,18] does not consider dependence on the discrete-state history and our prefix-based parametrization can be used in disturbance feedback controllers/filters as well to improve their performance. Moreover, as a minor difference from the literature, both disturbance feedback and measurement feedback with Q-parametrization are mostly used for optimal control while we use the latter to enforce constraints in a worst-case setting using tools from robust optimization, a problem stated as a future direction in [3]. Finally, the prefix dependence can be interpreted as essentially performing estimation at the discrete-level akin to estimators in hidden mode hybrid systems [27,28], however in our case the mode is observed (we know whether the measurement is missing or not at each time) but we are trying to estimate the mode-sequence (the missing data pattern). The prefix-based parametrization we propose is also closely related to path-dependent controllers proposed in [24] in the context of disturbance attenuation and stability for Markov jump linear systems, with the difference being that the problems in [24] lead to linear matrix inequalities in the controller gains due to different control objectives, whereas in our setting the problems are non-convex in the filter/controller gains and a nonlinear transformation is needed.

2. Notation and preliminaries

We denote the set of real numbers by $\mathbb{R}$ and the set of binary numbers by $\mathbb{B}$. Throughout this work, the norm $\|\cdot\|$ is the infinity norm, i.e., for a vector $v \in \mathbb{R}^n$, $\|v\| \doteq \max_{i=1,\dots,n} |v_i|$. The symbol $\otimes$ represents the Kronecker product, I_k represents the identity matrix of size k, $0_{k\times m}$ represents the $k \times m$ zero matrix, $\mathbb{1}_k$ denotes a k dimensional vector of ones. The subscripts are dropped when the dimension of the matrix is clear from the context. The operator $\text{diag} : \mathbb{R}^n \to \mathbb{R}^{n\times n}$ maps a vector $v \in \mathbb{R}^n$ to the $n \times n$ diagonal matrix with the elements of v on its main diagonal. For matrices and vectors, the inequalities $\geq$ are always taken element-wise. For a (block) vector v, v_k and $v_{i:j}$ denote its kth entry, and its sub-vector consisting of entries from ith to jth, respectively.

We call any finite set Σ an *alphabet*. In particular, we use $\Sigma = \mathbb{B}$ to represent missing data patterns. Any Σ-valued signal $q = q(t_0)q(t_0+1)\dots q(t_0+N)$ is called a *word*. The symbol Σ^* denotes the set of all finite-length words, whereas Σ^T and $\Sigma^{[T]}$ denote the set of all words with length equal to T or with length up to T that are formed by elements in Σ, respectively. For a word $q \in \Sigma^*$, its length is denoted by $|q|$. For $i \leq |q|$, we use $q_{[1:i]}$ to denote the length i prefix of q. For example, if $q = q(t_0)q(t_0+1)\dots q(t_0+N)$, then $q_{[1:i]} = q(t_0)q(t_0+1)\dots q(t_0+i-1)$. Finally, the set of all non-empty prefixes of q is denoted by $\mathit{Pref}(q)$. As a concrete example, the word $q = \{1001\} \in \mathbb{B}^{[4]}$ has length $|q| = 4$, $q_{[1:3]} = 100$ is a prefix, $q_{[2:3]} = 00$ is a subword but not a prefix, and $\mathit{Pref}(q) = \{1, 10, 100, 1001\}$. An arbitrary set $\mathcal{L}$ of words formed from a given alphabet Σ is called a language over Σ. We overload the Pref operator and use it for languages as $\mathit{Pref}(\mathcal{L}) \doteq \cup_{q\in\mathcal{L}} \mathit{Pref}(q)$.

2.1. Properties of block triangular matrices

In this section, we present some properties of block triangular matrices that are used to develop optimization results later in the paper.

Definition 1 (*Leading Block Principal Submatrix*)**.** The i-th *leading principal block submatrix* of a $l \times p$ block matrix $X \in \mathbb{R}^{al\times bp}$, denoted by $\mathcal{BM}_i(X)$, is the $l \times p$ block matrix:

$$\mathcal{BM}_i(X) = X([1:il], [1:ip])$$

for all $i \in [1, \min(a, b)]$, where $X([1:il], [1:ip])$ indicates the submatrix formed by all entries of matrix X that are in both the first il rows and the first ip columns.

Several useful properties of the leading principal block matrix operator $\mathcal{BM}_i(\cdot)$ for block lower triangular matrices are stated next. Proofs of these statements can be found in Appendix A.

Lemma 1. *Let $W, X \in \mathbb{R}^{ap\times bq}$, $Y \in \mathbb{R}^{bq\times cr}$ and $Z \in \mathbb{R}^{as\times as}$. The following properties hold:*

1. $\mathcal{BM}_i(W + X) = \mathcal{BM}_i(W) + \mathcal{BM}_i(X)$;

2. *If X and Y are $p \times q$ and $q \times r$ block lower triangular, respectively, then $\mathcal{BM}_i(XY) = \mathcal{BM}_i(X)\mathcal{BM}_i(Y)$;*
3. *If Z is nonsingular and $s \times s$ block lower triangular, then $\mathcal{BM}_i(Z^{-1}) = (\mathcal{BM}_i(Z))^{-1}$,*

for all $i \in [1, min(a, b, c)]$.

Proposition 1. *Let $\bar{C}^{(1)}$ and $\bar{C}^{(2)}$ be $p \times n$ block lower triangular matrices that share the same j-th leading block principal submatrix:*

$$\mathcal{BM}_j(\bar{C}^{(1)}) = \mathcal{BM}_j(\bar{C}^{(2)}).$$

Also let $F^{(1)}$, $F^{(2)}$ be $n \times p$ block lower triangular matrices and let S be an $n \times n$ block lower triangular matrix, all with compatible block sizes. Define, for $i \in \{1, 2\}$,

$$Q^{(i)} \doteq F^{(i)}(I - \bar{C}^{(i)}SF^{(i)})^{-1}. \tag{1}$$

Then,

$$\mathcal{BM}_j(F^{(1)}) = \mathcal{BM}_j(F^{(2)}) \in \mathbb{R}^{jn \times jp}$$

if and only if

$$\mathcal{BM}_j(Q^{(1)}) = \mathcal{BM}_j(Q^{(2)}) \in \mathbb{R}^{jn \times jp}.$$

Proposition 2. *Consider the following pairs of matrices $(\bar{C}^{(1)}, \bar{C}^{(2)})$ and $(Q^{(1)}, Q^{(2)})$ that share the same j-th principal block leading submatrix amongst each pair*

$$\mathcal{BM}_j(\bar{C}^{(1)}) = \mathcal{BM}_j(\bar{C}^{(2)}),$$
$$\mathcal{BM}_j(Q^{(1)}) = \mathcal{BM}_j(Q^{(2)})$$

and consider two vectors $f^{(1)}$ and $f^{(2)}$ and a block lower triangular matrix S. Define, for $i \in \{1, 2\}$:

$$r^{(i)} = (I + Q^{(i)}\bar{C}^{(i)}S)f^{(i)}. \tag{2}$$

Then, the vectors $f^{(1)}$ and $f^{(2)}$ satisfy:

$$f_k^{(1)} = f_k^{(2)} \quad \forall k \in [1, jn]$$

if and only if the first jn entries of the vector $r^{(1)}$ is identical to that of $r^{(2)}$:

$$r_k^{(1)} = r_k^{(2)} \quad \forall k \in [1, jn].$$

3. Problem setup

As alluded to in the introduction, the goal of this paper is to design feedback control or estimation mechanisms that are robust to missing measurements. This section describes the system model considered, including the missing data patterns. Then, we formally state the problem.

3.1. Model description

We consider discrete-time affine systems with state update and measurement equations defined as:

$$\begin{aligned} x(t+1) &= Ax(t) + Bu(t) + w(t) + k, \quad w(t) \in \mathcal{W}, \\ y(t) &= \begin{cases} Cx(t) + v(t), & q(t) = 1, \\ \emptyset, & q(t) = 0, \end{cases} \quad v(t) \in \mathcal{V}, \end{aligned} \tag{3}$$

where A, B, C, k are known system matrices, $x(t) \in \mathcal{X} \subseteq \mathbb{R}^n$ is the continuous state, $u(t) \in \mathcal{U} \subseteq \mathbb{R}^m$ is the input, $w(t) \in \mathcal{W} \subseteq \mathbb{R}^n$ is the process noise, $y(t) \in \mathcal{Y} \subseteq \mathbb{R}^p \cup \{\emptyset\}$ is the output measurements of the system, $q(t) \in \mathbb{B}$ is the discrete state/mode of the system, with $q(t) = 1$ denoting that the measurement vector is available and $q(t) = 0$ denoting that the measurement vector is not available (i.e., "missing"), and $v(t) \in \mathcal{V} \subseteq \mathbb{R}^p$ is the measurement noise. The noise terms $w(t)$ and $v(t)$ are unknown but bounded, and their bounds are known (i.e., $\mathcal{W} = \{w \in \mathbb{R}^n \mid \|w\| \leq \eta_w\}$ and $\mathcal{V} = \{v \in \mathbb{R}^p \mid \|v\| \leq \eta_v\}$). Moreover, we assume that the control input $u(t)$ is bounded (i.e., $\mathcal{U} = \{u \in \mathbb{R}^m \mid \|u\| \leq \eta_u\}$).

Remark 1. We note that the methods developed in this paper can be extended to work with systems that have time-varying matrices in the place of A, B, C, f. However, this case is omitted for clarity of notation. Similarly, the sets $\mathcal{W}$, $\mathcal{V}$ and $\mathcal{U}$ can be arbitrary polytopes but for simplicity, we constrain them to be hypercubes in the rest of the paper.

The performance of any controller or estimator designed for the system in (3) clearly depends on how much information is received, and yet the evolution of the discrete state $q(t)$ is not included in (3). To model this, we introduce a constraint on the evolution of $q(t)$. If the evolution of the discrete state is not constrained at all, one possibility is $q(t) = 0$ for all times, and there is no measurement for feedback. However, in many applications, it is possible to have a priori information about admissible missing data patterns (e.g., based on a network device's specification sheet or the knowledge about communication protocols that are packet-drop tolerant). To describe the evolution of the discrete state $q(t)$, we introduce the missing data language $\mathcal{L}$:

Definition 2 (*Missing Data Language*)**.** A **missing data language** $\mathcal{L} \subseteq \mathbb{B}^T$ is a set of words q, called a **missing data pattern**, that describes the possible trajectories of $q(t)$ in the system (3). A mode signal $q = q(t_0), q(t_0+1), \ldots, q(t_0+T-1)$ is said to **satisfy** the missing data language $\mathcal{L}$ if $q \in \mathcal{L}$.

Remark 2. Other works in the literature have considered representing missing data patterns or sequences with concepts such as (m, k)-firmness [16,17] or a bound on the number of consecutive missing data packets. The representation that we describe here is general enough that it can express both of these concepts.

3.2. Problem statement

In what follows, we describe several constrained control and estimation problems for the system in (3). Designing a controller that enforces certain state constraints or an estimator that guarantees that estimation error remains bounded is particularly challenging when the controller or estimator does not have access to all measurements for all times. Therefore, we introduce a relaxed invariance-type objective that we call equalized recovery. Finally, we present a formal unified problem statement.

An observer or an estimator $\mathcal{O} : (\mathcal{Y} \times \mathcal{U} \times \mathbb{B})^* \mapsto \mathcal{X}$ maps the measured input/output sequence and the missing data sequence to an estimate $\hat{x}$ of the state. In particular, we will consider Luenberger-like estimator structures of the form:

$$\begin{aligned} \hat{x}(t+1) &= A\hat{x}(t) + Bu(t) - u_e(t) + k, \\ \hat{y}(t) &= C\hat{x}(t), \end{aligned} \tag{4}$$

where the injection term $u_e(t)$ is the design variable. The constrained estimation problem aims to impose constraints on the estimation error $e(t) = x(t) - \hat{x}(t)$ by appropriately designing $u_e(t)$.

Another problem of interest is that of synthesizing a controller $\mathcal{C} : (\mathcal{Y} \times \mathbb{B})^* \mapsto \mathcal{U}$, where the design variable is $u(t)$ in (3). The goal is to impose constraints on the state of the closed-loop system, while it is required to respect the constraints on the input imposed by the set $\mathcal{U}$. It is possible to pose a similar problem for tracking control, where there is a given desired trajectory $x_d(t_0), x_d(t_0+1), \ldots, x_d(t_0+T)$ and its corresponding $u_d(t_0), u_d(t_0+1), \ldots, u_d(t_0+T-1)$ such that $x_d(t+1) = Ax_d(t) + Bu_d(t) + k$, and the objective is to guarantee constraints on the tracking error $\xi(t) \doteq x(t) - x_d(t)$. One can also consider imposing constraints on affine functions of the states.

Both estimation and control synthesis problems can be mapped (with slight modifications) to a generic constrained control problem on a unified system:

$$\begin{aligned} \xi(t+1) &= A\xi(t) + B_\xi u_\xi(t) + w(t) + k_\xi, \quad w(t) \in \mathcal{W}_\xi, \\ y_\xi(t) &= \begin{cases} C\xi(t) + v(t), & q(t) = 1, \\ \emptyset, & q(t) = 0, \end{cases} \quad v(t) \in \mathcal{V}_\xi, \end{aligned} \tag{5}$$

where the transformed state $\xi(t)$, the transformed output $y_\xi(t) \in \mathcal{Y}_\xi$ and the transformed input $u_\xi(t) \in \mathcal{U}_\xi$, as well as B_ξ, k_ξ, $\mathcal{U}_\xi$, and $\mathcal{Y}_\xi$ represent different signals, matrices and sets depending on the problem of interest (for the sake of completeness, they are provided in Appendix B). The proper objective for both problems is then to design a feedback law for $u_\xi(t)$ as a function of all previous outputs $\{y_\xi(\tau)\}_{\tau=t_0}^{t}$ and discrete states $\{q(\tau)\}_{\tau=t_0}^{t}$ such the states of the system (5) satisfies certain constraints.

Finally, as mentioned earlier, when the measurements can be missing as in (5), it is not reasonable to expect that the constraints hold invariantly. For instance in an estimation or tracking problem, it might be reasonable to allow a larger bound during missing data events. To capture this, we define a new type of constraint that relaxes invariance.

Definition 3 (*Equalized Recovery*)**.** A discrete-time dynamical system as in (5) is said to achieve an equalized recovery level M_1 with recovery time T and intermediate level $M_2 \geq M_1$ at time t_0 if for any initial state with $\|\xi(t_0)\| \leq M_1$, we have $\|\xi(t)\| \leq M_2$ for all $t \in [t_0, t_0+T]$ and $\|\xi(t_0+T)\| \leq M_1$.

Equalized recovery expresses a form of boundedness for the trajectories of the system and can be viewed as a form of weak "invariance", where instead of enforcing the set $\mathcal{X}_1 \doteq \{x \mid \|x\| \leq M_1\}$ being invariant, we relax the invariance condition and allow the states to be in $\mathcal{X}_2 \doteq \{x \mid \|x\| \leq M_2\}$ as long as they recover back to the set $\mathcal{X}_1$. Equalized recovery's interpretation as weak invariance is somewhat related to multi-set invariance discussed in [29] for switched systems. For estimation problems, in the special case where $M_1 = M_2$ and $T = 1$, equalized recovery reduces to equalized performance [6,25,30], which essentially states that $\|\xi(t)\| \leq M_1$ should be invariant with ξ being the estimation error.

Remark 3. Instead of imposing equalized recovery on the state $\xi(t)$, it also possible to consider equalized recovery on a subset or a linear combination of the state, i.e., $z(t) = L\xi(t)$, if desired. The proposed synthesis solution in Section 4 can be slightly modified to account for this in a straightforward manner. Furthermore, in addition to the infinity norm, we can also consider other "set templates" such as zonotopes with minor modifications to our proposed solution.

Given all these elements, the problem we are interested in can be formally stated as follows:

Problem 1. Consider a missing data language $\mathcal{L} \subseteq \mathbb{B}^T$, and a system of the form (5), whose mode signal $q(t)$, $t \in [t_0, t_0 + T - 1]$ satisfies the missing data language $\mathcal{L}$. Given the recovery level M_1, intermediate level $M_2 \geq M_1$, and recovery time T, find a feedback law $\Gamma : (\mathcal{Y}_\xi \times \mathbb{B})^* \mapsto \mathcal{U}_\xi$ such that the system achieves an equalized recovery level M_1 with recovery time T and intermediate level M_2.

4. Synthesis of a prefix-based feedback

This section addresses the feedback synthesis problem in Problem 1 by developing convex optimization problems that can construct affine feedback laws. We start with a commonly used time-based affine feedback law and show how language constraints can be integrated in this. Then, we present a new feedback structure that updates the gains based on the prefix of the missing data pattern seen so far and show that prefix-based feedback laws generalize the time-based ones and, moreover, that the synthesis of prefix-based affine feedback laws can be reduced to a convex optimization problem. The proofs of the main results are provided in Appendix C.

4.1. Time-based feedback laws and their limitations

A common structure for feedback laws, which we call *time-based*, takes the following form [8]:

$$u_\xi(t) = f(t) + \sum_{\tau=t_0}^{t} F_{(t,\tau)} y_\xi(\tau). \tag{6}$$

Note that if there is a single missing data pattern $q^{(*)}$ (i.e., the language $\mathcal{L}$ has only one word), one could interpret the system in (5) as a linear time-varying system instead of a switched system, by defining a time-varying measurement matrix $C(t) = 0$ when $q^{(*)}(t) = 0$, and $C(t) = C$ when $q^{(*)}(t) = 1$, which essentially sets the output to 0 when it is missing. If we define $u_\xi \doteq \begin{bmatrix} u_\xi(t_0)^\top, & u_\xi(t_0+1)^\top, & \cdots & u_\xi(t_0+T-1)^\top \end{bmatrix}^\top$ associated with a word $q^{(*)}$, the feedback laws in (6) can be written in matrix–vector form as follows:

$$u_\xi = f^{(*)} + F^{(*)} y_\xi,$$

where

$$f^{(*)} \doteq \begin{bmatrix} f(t_0)^\top & f(t_0+1)^\top & \cdots & f(t_0+T-1)^\top \end{bmatrix}^\top \tag{7}$$

and

$$F^{(*)} \doteq \begin{bmatrix} F_{(t_0,t_0)} & 0 & \cdots & 0 \\ F_{(t_0+1,t_0)} & F_{(t_0+1,t_0+1)} & & \vdots \\ \vdots & & \ddots & 0 \\ F_{(t_0+T-1,t_0)} & F_{(t_0+T-1,t_0+1)} & \cdots & F_{(t_0+T-1,t_0+T-1)} \end{bmatrix}. \tag{8}$$

The design of feedback laws of the form (6), or equivalently finding $(F^{(*)}, f^{(*)})$ that guarantees certain convex constraints on the state and input, is in general a non-convex problem due to the states of the closed-loop system being a non-convex function of the gains $(F^{(*)}, f^{(*)})$. Nevertheless, a non-linear change of variables, namely Q-parametrization, is used in [8] to render the design of such feedback laws a convex problem.

Our previous work [7] introduces a method for representing any given language $\mathcal{L}$ with a more difficult language $\mathcal{L}^*$ that satisfies $|\mathcal{L}^*| = 1$. We do this by introducing a partial order for words $q^{(1)}, q^{(2)} \in \mathbb{B}^T$ for arbitrary T as:

$$q^{(1)} \preceq q^{(2)} \quad \iff \quad (\forall i \in [1, T])(q^{(1)}_{[i]} = 0 \implies q^{(2)}_{[i]} = 0).$$

With this partial order, one can derive a word that is uniquely "harder" or "more missing" than any in the given language $\mathcal{L}$, in the sense of being the least upper bound for the set $\mathcal{L}$. We refer to this least upper bound as the worst-case word q^* and the worst-case language is then given by $\mathcal{L}^* \doteq \{q^*\}$. Solving for gains $(F^{(*)}, f^{(*)})$ associated with $\{q^*\}$ that guarantees an equalized recovery level, then, guarantees the same equalized recovery level for any missing data pattern in $\mathcal{L}$ when these gains are used for feedback [7].

However, the timed-based solution based on the use of the worst-case language $\mathcal{L}^*$ has some limitations. Consider the following language $\mathcal{L} = \{q^{(1)}, q^{(2)}\}$ with $q^{(1)} = 1011$ and $q^{(2)} = 1101$. In this case, the worst-case language $\mathcal{L}^*$ can be identified by performing a bitwise AND of $q^{(1)}$ and $q^{(2)}$: $\mathcal{L}^* = \{1001\}$.

Note that in the above example with the language $\mathcal{L} = \{q^{(1)}, q^{(2)}\}$, the discrete state's value at time $t = t_0 + 1$ directly describes the word in $\mathcal{L}$ that is being executed. i.e., if $q(t_0 + 1) = 1$, then we know that trajectory $q^{(2)}$ of the discrete state is occurring; otherwise, $q^{(1)}$ is occurring. So, if we make that identification at time t we use a set of feedback gains that are specific to an individual word after the second time step instead of the feedback gains designed with $\mathcal{L}^*$. In such a scheme, the feedback would only need to be open-loop on one time instance. Using a new set of gains that were specific to a single word in $\mathcal{L}$ instead of the worst-case word q^* would obviously reduce the recovery level in this case and across a broad number of other examples. On the other hand, time-based feedback ignores the observed mode sequence so far and tries to be robust against all words in the language in an open-loop (in the discrete mode) fashion.

The above might indicate that one could simply synthesize a feedback gain for each word q in the language $\mathcal{L}$ and choose the proper gain at runtime to arrive at a less conservative solution, but the selection of a proper gain can be tricky even in the simple example that was shown above. Ultimately, we cannot select the exact gain that we need in such a solution because we cannot know the word that is being executed at the beginning of the time horizon. For instance, in the example language above, it is impossible to discern which word is occurring after receiving the measurement at time t_0 of $q(t_0) = 1$ because 1 is a prefix of both words in $\mathcal{L}$. This motivates a new type of prefix-based feedback structure that we introduce in the next subsection.

4.2. Prefix-based feedback laws

To overcome the limitations of the time-based feedback laws, the feedback laws for the controllers/estimators should have some understanding of the currently observed sequence of the discrete state, which we capture by the following feedback structure,

$$u_\xi(q_{[t_0:t]}) = f(t, q_{[t_0:t]}) + \sum_{\tau=t_0}^{t} F_{(t,\tau,q_{[t_0:t]})} y_\xi(\tau), \tag{9}$$

We call this a *prefix-based* feedback law. Similarly, estimators (4) and controllers with injection terms or outputs in the form (9) are called *prefix-based estimators* and *prefix-based controllers*, respectively.

While the time-based solutions use the available (non-missing) output measurement history, $\{y_\xi(\tau)\}_{\tau=t_0}^{t}$, for feedback, prefix-based feedback laws use both the output history and the discrete-state history, $\{y_\xi(\tau)\}_{\tau=t_0}^{t}$ and $\{q(\tau)\}_{\tau=t_0}^{t}$. By its definition, it essentially also performs estimation at the discrete-level (or online model detection) to detect which missing data pattern in $\mathcal{L}$ is active and adapts the filter gains accordingly. Whereas, the time-based estimator/controller is agnostic to the missing data pattern and tries to be robust rather than adaptive. The following proposition formally captures the fact that prefix-based estimators/controllers are more general than time-based estimators/controllers.

Proposition 3. *For any time-based feedback law for the dynamical system in (5) with missing data pattern given by a fixed-length language $\mathcal{L}$, identical performance can be obtained using a prefix-based feedback law.*

Proof. Let the transformed input term for the time-based feedback law be

$$u_\xi(t) = \bar{f}(t) + \sum_{\tau=t_0}^{t} \bar{F}_{(t,\tau)} y_\xi(\tau). \tag{10}$$

Define the gains of the prefix-based feedback law's transformed input term in (9) as $f(t, \lambda) \doteq \bar{f}(t)$, $F_{(t,\tau,\lambda)} \doteq \bar{F}_{(t,\tau)}$ for all $t \in [t_0, t_0 + T - 1]$, $\tau \in [t_0, t]$ and for all $\lambda \in \bigcup_{q\in\mathcal{L}} Pref(q)$. Then the two feedback laws are equivalent. □

In order to design prefix-based feedback gains, we associate with each word $q^{(i)} \in \mathcal{L}$ a pair of gain matrices $(F^{(i)}, f^{(i)})$ defined as in (7)–(8). However, since at run-time we do not know which word $q^{(i)}$ is active, we enforce some constraints on the gain matrices of words sharing prefixes as follows:

$$(p \in Pref(q^{(i)}) \cap Pref(q^{(j)})) \implies \begin{array}{l} \left(\mathcal{BM}_{|p|}(F^{(i)}) = \mathcal{BM}_{|p|}(F^{(j)})\right) \\ \quad \wedge \left((f^{(i)})_{1:|p|m} = (f^{(j)})_{1:|p|m}\right), \end{array} \quad \forall q^{(i)}, q^{(j)} \in \mathcal{L}. \tag{11}$$

Moreover, we utilize the constrained optimal control approach in [8] to jointly solve for all $(F^{(i)}, f^{(i)})$, which involves the use of Q-parametrization to convert the non-convex problem into a convex one. However, it is well known that Q-parametrization does not generally lead to convex problems when additional structure on the gains $F^{(i)}$ and $f^{(i)}$ are imposed. One of our main results is thus to show that the prefix dependency of the gains in (11) still leads to a convex problem when using Q-parametrization. More precisely, in the following theorem, we will present a bijection relationship between the constraints (11) on the gain matrices/vectors of prefix-based feedback of the form in (9) with the parametrization in (1) and (2). Hence, the convexity of the corresponding Q-parametrization is preserved when imposing the prefix dependency constraints.

Theorem 1. *Given a prefix-based estimator/controller with the transformed input term (9), we associate with it block matrices $\{(F^{(i)}, f^{(i)})\}_{i=1}^{|\mathcal{L}|}$ formed from the filter gains, where for all $q^{(i)} \in \mathcal{L}$, the (j,k)-block entry $F^{(i)}_{jk}$ of $F^{(i)}$ is defined as*

$$F^{(i)}_{jk} \doteq F_{(t_0+j-1,t_0+k-1,q^{(i)}_{[1:j]})} \tag{12}$$

$\forall k \in [1,j]$, $\forall j \in [1,T]$, and $F^{(i)}_{jk} = 0$ otherwise; and the j-th block entry of the feedforward term $f^{(i)}$ is defined as

$$f^{(i)}_{j} \doteq f(t_0 + j - 1, q^{(i)}_{[1:j]}) \tag{13}$$

$\forall j \in [1,T]$. Let S and $\bar{C}^{(i)}$ be as:

$$\begin{aligned} \bar{C}^{(i)} &\doteq \begin{bmatrix} diag(q^{(i)}) \otimes C & 0_{pT\times n} \end{bmatrix}, \\ S &\doteq \begin{bmatrix} 0 & 0 & \cdots & 0 \\ B_\xi & 0 & \cdots & 0 \\ AB_\xi & B_\xi & \cdots & 0 \\ \vdots & \vdots & \ddots & 0 \\ A^{T-1}B_\xi & A^{T-2}B_\xi & \cdots & B_\xi \end{bmatrix}. \end{aligned} \tag{14}$$

Then, Eqs. (1) and (2) define a bijection such that any estimator $\{(F^{(i)}, f^{(i)})\}_{i=1}^{|\mathcal{L}|}$ is paired with one and only one element in the polyhedral set:

$$\mathcal{Q}(\mathcal{L}) \doteq \left\{ \{(Q^{(i)}, r^{(i)})\}_{i=1}^{|\mathcal{L}|} \middle| \begin{array}{ll} Q^{(i)} \text{ is block lower diagonal}, & \forall i, \\ (p \in Pref(q^{(i)}) \cap Pref(q^{(j)})) \implies & \begin{array}{l} \left(\mathcal{BM}_{|p|}(Q^{(i)}) = \mathcal{BM}_{|p|}(Q^{(j)})\right) \\ \wedge \left((r^{(i)})_{1:|p|m} = (r^{(j)})_{1:|p|m}\right), \end{array} \forall q^{(i)}, q^{(j)} \in \mathcal{L} \end{array} \right\}. \tag{15}$$

Using the above theorem, a necessary and sufficient condition for the existence of prefix-based estimators and controllers that solve Problem 1 can then be formulated as follows:

Theorem 2 (*Estimator and Controller Synthesis with Missing Data (Prefix-Based)*)**.** *There exists a prefix-based estimator/controller (i.e., $\{(F^{(i)}, u_0^{(i)})\}_{i=1}^{|\mathcal{L}|}$) that satisfies equalized recovery with parameters $(M_1, M_2, \mathcal{L})$ if and only if the following robust linear programming problem is feasible:*

$$\text{Find} \quad \left\{(Q^{(i)}, r^{(i)})\right\}_{i=1}^{|\mathcal{L}|} \in \mathcal{Q}(\mathcal{L}) \tag{16a}$$

$$\text{subject to} \quad \begin{array}{l} \forall (\|w\| \le \eta_w, \|v\| \le \eta_v, \|\xi(t_0)\| \le M_1): \\ \quad \|u_\xi + u_d\| \le \eta_u, \|\xi^{(i)}\| \le M_2 \text{ and } \|\begin{bmatrix} 0_{n\times nT} & I_n \end{bmatrix} \xi^{(i)}\| \le M_1, \end{array} \quad \forall i \in [1, |\mathcal{L}|], \tag{16b}$$

where

$$\begin{aligned} \xi^{(i)} &= (H + SQ^{(i)}\bar{C}^{(i)}H)w + SQ^{(i)}N^{(i)}v + (I + SQ^{(i)}\bar{C}^{(i)})(J\xi(t_0) + H\tilde{k}) + Sr, \\ u_\xi &= Q^{(i)}\bar{C}^{(i)}Hw + Q^{(i)}N^{(i)}v + Q^{(i)}\bar{C}^{(i)}(J\xi(t_0) + H\tilde{k}) + r, \end{aligned} \tag{17}$$

$$\bar{C}^{(i)} \text{ and } S \text{ are defined in (14), } \mathcal{Q}(\mathcal{L}) \text{ is as defined in (15),} \tag{18}$$

$$N^{(i)} = diag(q^{(i)}) \otimes I, \ \ \tilde{k} = \mathbb{1}_T \otimes k_\xi, \tag{19}$$

$$J \doteq \begin{bmatrix} I_n \\ A \\ \vdots \\ A^{T-1} \\ A^T \end{bmatrix}, \ H = \begin{bmatrix} 0 & 0 & \cdots & 0 \\ I_n & 0 & \cdots & 0 \\ A & I_n & \cdots & 0 \\ \vdots & \vdots & \ddots & 0 \\ A^{T-1} & A^{T-2} & \cdots & I_n \end{bmatrix}. \tag{20}$$

Remark 4. The above feasibility problem can be modified to minimize the intermediate level M_2 subject to a given equalized recovery level M_1 and given missing data language $\mathcal{L}$, which can be easily shown to be a robust linear program over the decision variables $\left\{(Q^{(i)}, r^{(i)})\right\}_{i=1}^{|\mathcal{L}|}$ and M_2.

Since the feasibility problem in (21) contains semi-infinite constraints due to the "for all" quantifier on the uncertain terms, the problem is not readily solvable. However, as in [7], techniques from robust optimization and duality [31,32] can be applied to obtain a linear programming (LP) problem with only finitely many linear constraints. In particular, we have the following theorem:

Theorem 3 (*Robustified Estimator and Controller Synthesis with Missing Data (Prefix-Based)*)**.** *The feasibility of prefix-based finite horizon affine estimators and controllers that solve Problem 1 is equivalent to the feasibility of the following linear*

optimization problem:

$$
\begin{aligned}
&\textit{Find} && \left\{(Q^{(i)}, r^{(i)})\right\}_{i=1}^{|\mathcal{L}|} \in \mathcal{Q}(\mathcal{L}),\ \left\{(\Pi_1^{(i)}, \Pi_2^{(i)}, \Pi_3^{(i)})\right\}_{i=1}^{|\mathcal{L}|}\\
&\textit{subject to} && \forall i \in [1, |\mathcal{L}|],\ q^{(i)} \in \mathcal{L},\\
&&& \Pi_1^{(i)} \geq 0,\ \Pi_2^{(i)} \geq 0,\ \Pi_3^{(i)} \geq 0,\\
&&& \Pi_1^{(i)} \begin{bmatrix} \eta_w \mathbb{1} \\ \eta_v \mathbb{1} \\ M_1 \mathbb{1} \end{bmatrix} \leq M_2 \mathbb{1} - \begin{bmatrix} I \\ -I \end{bmatrix} (Sr^{(i)} + (I + SQ^{(i)}\bar{C}^{(i)})H\tilde{k}),\\
&&& \Pi_2^{(i)} \begin{bmatrix} \eta_w \mathbb{1} \\ \eta_v \mathbb{1} \\ M_1 \mathbb{1} \end{bmatrix} \leq M_1 \mathbb{1} - \begin{bmatrix} I \\ -I \end{bmatrix} \begin{bmatrix} 0_{n \times nT} & I_n \end{bmatrix} (Sr^{(i)} + (I + SQ^{(i)}\bar{C}^{(i)})H\tilde{k}),\\
&&& \Pi_3^{(i)} \begin{bmatrix} \eta_w \mathbb{1} \\ \eta_v \mathbb{1} \\ M_1 \mathbb{1} \end{bmatrix} \leq \eta_u \mathbb{1} - \begin{bmatrix} I \\ -I \end{bmatrix} (r^{(i)} + Q^{(i)}\bar{C}^{(i)}H\tilde{k} + u_d),\\
&&& \Pi_1^{(i)} P_\eta = \begin{bmatrix} I \\ -I \end{bmatrix} G^{(i)},\ \Pi_2^{(i)} P_\eta = \begin{bmatrix} I \\ -I \end{bmatrix} \begin{bmatrix} 0_{n \times nT} & I_n \end{bmatrix} G^{(i)},\ \Pi_3^{(i)} P_\eta = \begin{bmatrix} I \\ -I \end{bmatrix} \tilde{G}^{(i)},
\end{aligned}
\tag{21}
$$

where $\tilde{k}$, J, H and $N^{(i)}$ are as defined in (19) and (20), $\bar{C}^{(i)}$ and S are as defined in (14), $\mathcal{Q}(\mathcal{L})$ is as defined in (15), and

$$
\begin{aligned}
G^{(i)} &= \begin{bmatrix} (I + SQ^{(i)}\bar{C}^{(i)})H & SQ^{(i)}N^{(i)} & (I + SQ^{(i)}\bar{C}^{(i)})J \end{bmatrix},\\
\tilde{G}^{(i)} &= \begin{bmatrix} Q^{(i)}\bar{C}^{(i)}H & QN^{(i)} & Q^{(i)}\bar{C}^{(i)}J \end{bmatrix},\\
P_\eta &= \begin{bmatrix} I & 0 & 0 \\ -I & 0 & 0 \\ 0 & I & 0 \\ 0 & -I & 0 \\ 0 & 0 & I \\ 0 & 0 & -I \end{bmatrix}.
\end{aligned}
$$

Moreover, if (21) is feasible, then we may invert the mappings in (1) and (2) to obtain:

$$F^{(i)} \doteq (I + Q^{(i)}\bar{C}^{(i)}S)^{-1}Q^{(i)}, \tag{22}$$

$$f^{(i)} \doteq (I + Q^{(i)}\bar{C}^{(i)}S)^{-1}r^{(i)}, \tag{23}$$

and we can establish that

1. Each $F^{(i)}$ is block lower triangular;
2. For all $\lambda \in Pref(q^{(i)}) \cap Pref(q^{(j)})$, we have $\mathcal{BM}_{|\lambda|}(F^{(i)}) = \mathcal{BM}_{|\lambda|}(F^{(j)})$ and $f_k^{(i)} = f_k^{(j)}$ for all $k \in [1 : |\lambda|n]$;
3. A prefix-based estimator or controller solving Problem 1 is defined by

$$u_\xi(\lambda) = f(t, \lambda) + \sum_{\tau=t_0}^{t} F_{(t,\tau,\lambda)} y_\xi(\tau), \tag{24}$$

where $\lambda \in \bigcup_{q^{(i)} \in \mathcal{L}} Pref(q^{(i)})$, $t \doteq t_0 + |\lambda| - 1$, and the matrices $F_{(t,\tau,\lambda)}$ and $f(t, \lambda)$ are defined according to (12) and (13).

5. Discussions

In this section, we discuss how the proposed finite-horizon estimators and controllers can be implemented. We also highlight the relation between equalized recovery and more familiar notions of detectability and stabilizability.

5.1. Implementation strategies

Assuming that the optimization problems proposed in the previous section have a feasible solution, there are multiple scenarios in which the estimator or controller can be applied. First, if the problem under consideration is one that is finite horizon, we can directly use the obtained gains. Second, if the missing data pattern repeats itself with a period of T time-steps, then the same gains can be used with period T since they guarantee that the recovery period M_1 is reached at the end of the period.

Alternatively, the gains can be used in conjunction with an estimator that guarantees equalized performance or a controller that guarantees forward invariance of M_1 level when there is no missing data. In particular, if we consider

languages $\mathcal{L}$ with words that start with a $q(t) = 0$, then we can switch from the equalized performance estimator/invariance controller to equalized recovery one whenever a missing measurement occurs and revert back to the equalized performance estimator/invariance controller after the recovery time T.

In addition, instead of using hypercubes as set templates due to our use of infinity norms, we could also use more flexible set templates, e.g., zonotopes. The incorporation of such set templates may be done using linear constraints as described in [33].

5.2. Relationship to detectability and stabilizability

In particular, we state the results in terms of the recent detectability definition for linear systems with data loss events [9]. Similar results also hold true for stabilizability. Consider a system:

$$\begin{aligned} x(t+1) &= Ax(t), \\ y(t) &= \begin{cases} Cx(t), & q(t) = 1, \\ \emptyset, & q(t) = 0. \end{cases} \end{aligned} \tag{25}$$

This is the same as the system model in (3) with input and noise terms omitted. We denote a system of the form (25) subject to a missing data language $\mathcal{L} \subseteq \mathbb{B}^\omega$ consisting of infinite words (hence the superscript ω) as $(A, C, \mathcal{L})$. Detectability of such a system is defined as follows [9]:

Definition 4 (*Missing Data Detectability*)**.** The system $(A, C, \mathcal{L})$ is said to be **detectable** if for any (infinite-length) $q \in \mathcal{L}$ and any initial state $x_0 \in \mathbb{R}^n$ with $y(t, x_0, q) = 0$ for all $t \in \mathbb{N}$, it holds that $x(t, x_0, q) \to 0$ as $t \to \infty$, where $y(t, x_0, q)$ and $x(t, x_0, q)$ are the output and state, respectively, at time t when the initial state is x_0.

In the following, we show that the existence condition for an equalized recovery estimator is a strict superset of the missing data detectability property defined above. First, we prove that the detectability of a system with missing data implies that an equalized recovery estimator exists and then, we provide an example where an equalized recovery estimator exists even when the system is not detectable. The proofs of these propositions are given in Appendix D.

Proposition 4. *If a system in the form of (25) with a missing data language $\mathcal{L}$, is detectable according to Definition 4, then for any recovery level $M_1 > 0$, there exist an intermediate level $M_2 \geq M_1$ and a recovery time $T \in \mathbb{N}$ such that there exists an estimator that achieves equalized recovery for the estimation error with these parameters for the missing data language $\mathcal{L}' = \{q \mid |q| = T,\ q \in Pref(\mathcal{L})\}$.*

Moreover, equalized recovery is slightly more general than detectability as stated next.

Proposition 5. *The set of discrete-time systems in the form (25) for which an estimator exists whose state estimation error satisfies equalized recovery is a strict superset of all detectable discrete-time systems with missing data.*

6. Examples

In this section, three different examples are used to illustrate important properties of the theory presented above. First, the improvement of prefix-based estimators over our previous work's time-based estimation is visualized by revisiting an example from [7]. Second, prefix-based controllers are utilized to discuss the problem of guaranteeing safety of a lane-keeping system when its sensors have missing data events. Finally, the scalability of these methods is shown with a multi-agent tracking problem where a group of followers attempts to follow a leader down a narrow passageway.

6.1. Estimator synthesis

In vehicle safety systems, there are many state estimation problems that must be solved within a finite time horizon (e.g., a vehicle that would like to understand where other vehicles are on the road before merging into a lane or exiting the highway). In these situations, a vehicle's safety (and the safety of its occupants) relies on execution of a maneuver within a time limit, T, because if a vehicle waits too long, it may miss its opportunity to continue towards its destination or the lane may end. One method of guaranteeing safety during such a maneuver is to provide bounds on how "good" the estimates of the other vehicles in the environment are during the time window. If the maneuver can be executed without entering any of the sets defined by our bounded error estimates of the other vehicles, then the vehicle can be guaranteed to be safe.

The Adaptive Cruise Controller is a driver assistance system which controls the acceleration of the user's car (the ego car) with two objectives: (i) to maintain a desired speed, if there is no vehicle in front of it, and (ii) to maintain a safe following distance, if there is a vehicle in front of it. Typically implemented with a radar or computer vision system, the

Table 1
Constants used in the automatic cruise control (ACC) example.

m	1370 kg	T_s	0.5 s
$\bar{k}_0$	7.58 N	η_w	0.1
$\bar{k}_1$	9.9407 N s/m	η_v	0.05

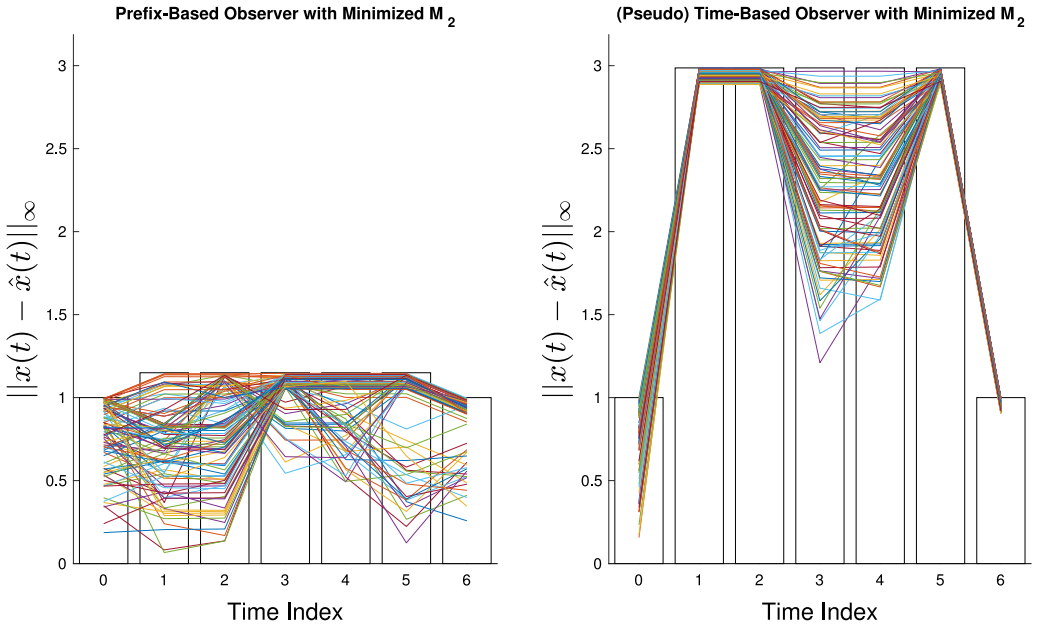

Fig. 1. Estimation error levels achieved by prefix-based (left) and time-based (right) estimators for the adaptive cruise control system. The minimum M_2 value for which equalized recovery is feasible, with $M_1 = 1$ and $T = 6$, is found by solving the robust linear program for the prefix-based and time-based feedback laws. The optimal M_2 that the prefix-based feedback can guarantee is $M_2 = 1.1498$ while the optimal M_2 that the time-based feedback can provide is $M_2 = 2.9864$.

adaptive cruise controller is a hybrid controller, but when considering the estimation error system we may analyze only the following linear system:

$$\begin{aligned} \xi(t+1) &= A\xi(t) + u_e(t) + Ew(t), \qquad w(t) \in \{w \in \mathbb{R}^3 \mid \|w\| \le \eta_w\}, \\ y_\xi(t) &= \begin{cases} C\xi(t) + v(t), & q(t) = 1, \\ \emptyset, & q(t) = 0, \end{cases} \quad v(t) \in \{v \in \mathbb{R}^2 \mid \|v\| \le \eta_v\}, \end{aligned} \tag{26}$$

where $\xi(t) = [\Delta v_e(t), \Delta h(t), \Delta v_L(t)]^T$ is the estimation error state, consisting of the error in the speed v_e of the ego vehicle, headway h, and speed v_L of the lead vehicle. Each matrix in (26) is defined as

$$A = \begin{bmatrix} e^{-\kappa T_s} & 0 & 0 \\ \frac{e^{-\kappa T_s} - 1}{\kappa} & 1 & T_s \\ 0 & 0 & 1 \end{bmatrix}, \quad C = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}, \quad E = \begin{bmatrix} 0 \\ \frac{T_s^2}{2} \\ T_s \end{bmatrix},$$

with $\kappa \triangleq \bar{k}_1/m$. Considering the parameter values selected in Table 1, one can completely define the system above. Then, when given a missing data language $\mathcal{L}$ and a set of parameters M_1 and M_2, we can pose this in the estimator synthesis form of Problem 1.

We will discuss the results of synthesis when considering the following language:

$$\begin{aligned} \mathcal{L}_1 &= \left\{ q \in \mathbb{B}^6 \;\middle|\; (q_{[1]} = q_{[6]} = 1) \wedge (\exists_1 i \in [2,5] \text{ s.t. } q_{[i]} = 0) \right\} \\ &= \left\{ 101111,\ 110111,\ 111011,\ 111101,\ 111111 \right\}, \end{aligned}$$

where the symbol $\exists_1 i$ indicates that there may exist up to one such element i. For this language, the worst-case language $\mathcal{L}_1^*$ can be found to be $\mathcal{L}_1^* = \{100001\}$.

For the language $\mathcal{L}_1$ and the recovery level $M_1 = 1$, the minimal value of the intermediate level M_2 is found using Remark 4 and Theorem 3. In contrast, another solution to the problem can be obtained using time-based feedback, the worst-case language $\mathcal{L}_1^*$, and Theorem 3 via a robust optimization similar to that of (21). From the intuition described in Section 4, we expect that the prefix-based estimator can in general guarantee tighter estimation error bounds than the

time-based estimator. A comparison is shown and discussed in more detail within Fig. 1 for the Adaptive Cruise Control example. In both cases, the system is initialized with random initial conditions and disturbances that satisfy the specified sets.

The time-based estimator's optimization required 1999 free variables and 0.2489 s to solve when using Gurobi [34], while the prefix-based estimator's optimization required 7993 free variables and 0.3637 s to solve.

6.2. Controller synthesis

Consider an automatic lane-keeping system. In such a system, one can imagine that the system has an estimate (with bounded error) of the vehicle's lateral position with respect to the center of the lane. Estimates might come from a computer vision system or other noisy sensors and thus can be subject to glare or misidentification of lane boundaries. This is where missing data events can arise.

In this system, instead of making estimates about where the vehicle is, we will try to control it so that it remains near the center of the lane. The simplified lane keeping model is defined by the following double integrator system:

$$\begin{aligned} \xi(t+1) &= \begin{bmatrix} 0 & 1 \\ 0 & -20 \end{bmatrix} \xi(t) + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u(t) + \begin{bmatrix} 1 \\ 0 \end{bmatrix} w(t), \;\; w(t) \in \mathcal{W} \doteq \{w \in \mathbb{R} \mid |w| \leq 0.05\} \\ y_\xi(t) &= \begin{cases} \xi(t) + v(t), & q(t) = 1, \\ \emptyset, & q(t) = 0, \end{cases} \qquad\qquad v(t) \in \mathcal{V} \doteq \{v \in \mathbb{R}^2 \mid \|v\| \leq 0.1\}, \end{aligned} \tag{27}$$

where the state $\xi(t) = [x_c(t), \dot{x}_c(t)]^T$ consists of the deviation $x_c(t)$ from the centerline of the lane and the lateral velocity $\dot{x}_c(t)$, and the control input $u(t)$ is the lateral force applied through steering.

Again, the trajectory of $q(t)$ is defined by a missing data language $\mathcal{L}$ which would come from the properties of the roads that the autonomous vehicle operates on as well as the specifications of its sensors. With all of the above information, the problem of guaranteeing safety can be posed as follows: Given that the vehicle state starts near the center of the lane with near zero lateral velocity, can we design a prefix-based feedback controller such that the vehicle never deviates outside of the lane boundaries despite missing data events from $\mathcal{L}$?

The formal interpretation of such a problem would be that given some specification of the initial position (i.e., our M_1 value), the lane width, along with the disturbance sets $\mathcal{W}$ and $\mathcal{V}$ and the missing data model $\mathcal{L}$, find a feedback law (9) such that the decision variable M_2 is minimized. If the minimal M_2 is below the lane width value, we can guarantee that the system will not deviate from the center of the lane when using our controller during a missing data event from $\mathcal{L}$.

Consider the system in (27), with the following missing data language:

$$\mathcal{L}_2 = \left\{ q \in \mathbb{B}^{12} \;\middle|\; \exists_1 i \in [1,\, 10] \text{ s.t. } \begin{array}{l} (q_{[i]} = 0) \wedge \\ (q_{[i+1]} = 0) \wedge \\ ((j \neq i) \wedge (j \neq i+1) \implies q_{[j]} = 1) \end{array} \right\}.$$

Let the initial state of the lane keeping system be within an infinity norm ball of radius $M_1 = 0.3$. Given that the process disturbances come from the set $\mathcal{W} = \{w \in \mathbb{R} \mid |w| \leq 0.05\}$ and the measurement disturbances come from the set $\mathcal{V} = \{v \in \mathbb{R}^2 \mid \|v\|_\infty \leq 0.1\}$, we synthesize a controller that minimizes the worst case deviation from the center of the lane. Note that the disturbance set $\mathcal{V}$ is overly conservative (most sensors can detect the lane boundaries of a lane to a precision of 0.01 m), but this is meant to simply show one of the many possible settings that the controller synthesis framework can handle.

In Fig. 2, we illustrate how some of the trickier initial conditions are handled. The figure contains multiple trajectories. Each trajectory begins with the same state on the boundary of the M_1 norm ball and experiences identical disturbances from the sets $\mathcal{W}$ and $\mathcal{V}$. The aspect that changes between each trajectory is the missing data pattern (which leads to different prefix-based feedback). One can see that for some of the hardest missing data patterns, the deviation from the center of the lane gets very close to the edge of our guarantee set (i.e., the boxes of width M_2), but always recovers to the proper level in the end.

This synthesis problem contained 46,773 free variables and was solved after 2.5420 s with Gurobi [34].

6.3. Controller synthesis: Formation control

Finally, we consider the problem of coordinating the movement of a fleet of agents through an obstacle filled environment. Precisely controlling formations of controlled agents has become very important across multiple domains including space exploration and disaster relief. For example, when using microsatellites to obtain many spacially distributed observations during orbit, maintenance of specific formations allows the devices to spend less fuel while achieving their mission [35]. In these contexts and many others, carefully controlled formations of the fleet make it easier to achieve a task and decrease the probability of collision with obstacles (e.g., space debris, other satellites).

Unfortunately, while navigating through obstacle-filled fields, localization methods that are based on a relative navigation sensing system or computer vision algorithms may experience missing measurements. In the following

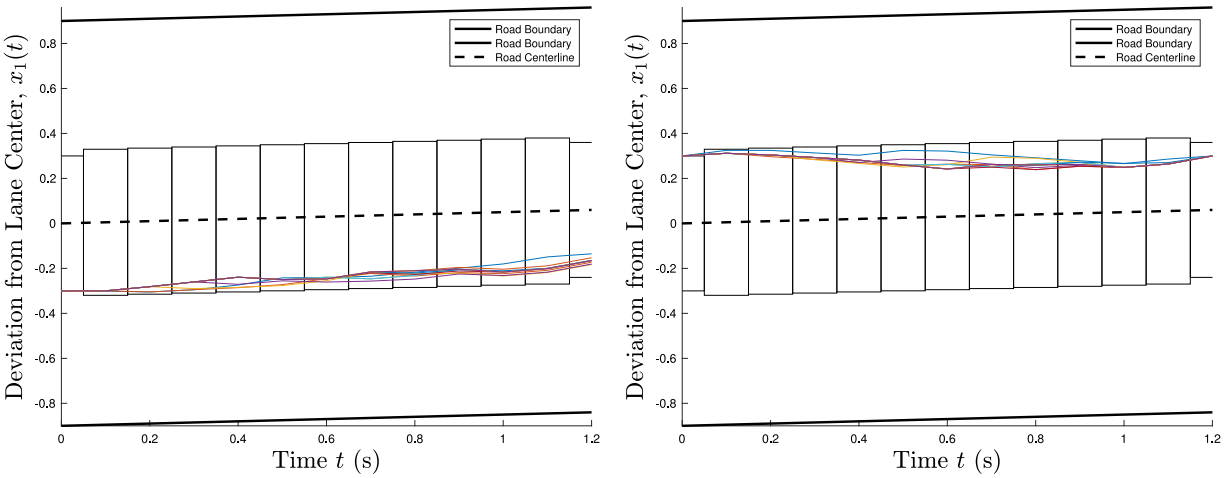

Fig. 2. Consider any one of the panels above. In each panel, multiple trajectories of the lane keeping system are visualized, where each trajectory is initialized at the same state on the $M_1 = 0.3$ hypercube's boundary and experiences the exact same disturbance (a carefully chosen, maximum norm disturbance). The only thing that varies across each trajectory is the missing data pattern σ. Thus, what causes the trajectories to diverge is how the prefix-based controller handles these missing data events when they happen. Regardless of the missing data pattern, it is shown that these adversarily chosen initial conditions and disturbances can still be guaranteed to return to the desired level M_1 and the system achieves equalized performance.

example, we show how a prefix-based control strategy for formations of such agents can be used to guarantee safe navigation of a narrow channel while maintaining a formation close to the desired one.

Consider the problem of designing a controller for a set of $n_r \cdot n_c$ agents with single-integrator dynamics moving in a two-dimensional plane. The agents seek to align themselves in n_r rows and n_c columns behind an uncontrolled lead agent (forming a grid). The rows are defined such that there are n_c agents in each row and the space between agents in a given row is always 2 m. Furthermore, the rows are organized in the y-direction such that they are evenly distributed between $\ell_y + 1$ and $\ell_y - 1$ where ℓ_y is the y-component of the leader's position.

The lead agent is moving with unknown, but bounded actions within the set $\mathcal{W} = \{w \in \mathbb{R}^2 \mid \|w\| \leq 1.5\}$. The following agents' movement can be defined in terms of error states in the x and y directions, $e_x^{(i,j)}(t) \doteq x^{(i,j)}(t) - \bar{x}^{(i,j)}(t)$ and $e_y^{(i,j)}(t) \doteq y^{(i,j)}(t) - \bar{y}^{(i,j)}(t)$, representing the difference in the (i,j)-th follower's x- and y- positions from its desired formation (i.e., grid) position.

In this work, the formation is maintained by every first agent in the row (any agent with $j = 1$ observing/sensing the states of the leader) and all other agents observing/sensing the states of the agent before it (agent j measures the position of agent $j-1$). When the desired states are known to be a constant offset from the leader's position or positions of other agents, the system may be written as follows:

$$e_x^{(i,j)}(t+1) = \begin{cases} (u_x^{(i,j)}(t) - w_x(t)) \cdot \Delta t, & j = 1, \\ (u_x^{(i,j)}(t) - u_x^{(i,j-1)}(t)) \cdot \Delta t, & \text{otherwise}, \end{cases}$$

$$e_y^{(i,j)}(t+1) = \begin{cases} (u_y^{(i,j)}(t) - w_y(t)) \cdot \Delta t, & j = 1, \\ (u_y^{(i,j)}(t) - u_y^{(i,j-1)}(t)) \cdot \Delta t, & \text{otherwise}, \end{cases}$$

$$y_{e,i}(t) = \begin{cases} \begin{bmatrix} e_x^{(i,j)}(t) \\ e_y^{(i,j)}(t) \end{bmatrix} + v_i(t), & q(t) = 1, \\ \emptyset, & q(t) = 0, \end{cases}$$

where the discretization step Δt is 0.1 s, the measurement disturbances $v_i(t)$ come from the set $\mathcal{V} \doteq \{v \in \mathbb{R}^2 \mid \|v\| \leq 0.5\}$ for all i and the input of the (i,j)-th agent as instantaneous speed in the x or y direction is written as $u_x^{(i,j)}(t)$ and $u_y^{(i,j)}(t)$, respectively.

In Figs. 3 and 4, the followers (black drones) are behind the leader (maize and blue drone) as the leader moves from left to right through a narrow passage, where localization information is sometimes dropped according to the language $\mathcal{L}_3$:

$$\mathcal{L}_3 = \left\{ \begin{matrix} 11111111, & 01111111, \\ 00111111, & 10111111, \\ 01011111, & 11011111 \end{matrix} \right\}.$$

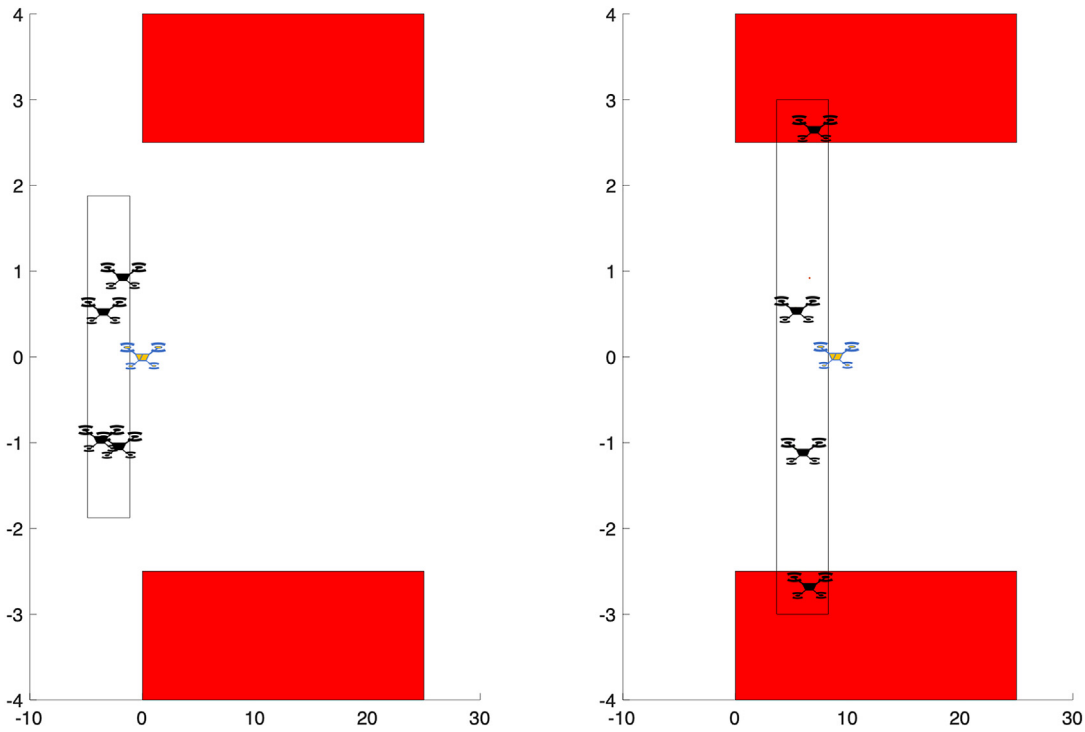

Fig. 3. A time-based controller using the worst-case language could not guarantee that the followers would safely exit the channel. It can guarantee that followers (black drones) will remain in the black outline defined by $M_2 = 2$ which overlaps with the red wall (thus collisions may happen).

Table 2
Analysis of the control synthesis time when the number of controlled agents increases.

Number of followers	Solver time (s)
4 (2 × 2 grid formation)	10.9708
9 (3 × 3 grid formation)	37.0226
16 (4 × 4 grid formation)	113.0512

The worst-case language in this case is $\mathcal{L}_3^* = \{00011111\}$. A time-based controller using $\mathcal{L}_3^*$ in this channel would have to adopt an open-loop strategy for the first three time steps and could not guarantee that the followers would avoid collision (see Fig. 3), but the prefix-based controller maintains the state error within a safe bound throughout the channel (see Fig. 4).

The 2 × 2 version of this problem contained 1519 free variables and was solved in 10.9708 s with Gurobi [34]. To illustrate the scalability of our control synthesis method, the solution time for this problem is compared in Table 2, where the number of following agents is varied. These simulations show that our proposed approach does involve longer computation times when the number of states are increased. Note, however, that our control synthesis is done offline and the computation time is thus not a critical limiting factor.

7. Conclusions

In this work, we presented a method for synthesizing bounded-error estimators and constrained controllers for affine systems that provide equalized recovery guarantees even in the presence of missing data, where the missing data patterns are constrained by a finite-length language. Our proposed solution leveraged Q-parametrization as well as some additional structure in our problem to provide the required inputs for estimation or control. We presented both time-based and prefix-based solutions. While the time-based solution is robust to the worst-case missing data pattern, the prefix-based solution implicitly estimates the specific missing data pattern (i.e., mode sequence), within the given language, based on observed history of missing data pattern so far and provides less conservative results. Since both the synthesis problems are reduced to linear programs, such controllers and estimators can be synthesized very efficiently.

Our current results are for finite horizon problems. Our future work includes developing similar estimators and controllers for infinite horizon problems where the missing data pattern is given by an automaton that also marks the

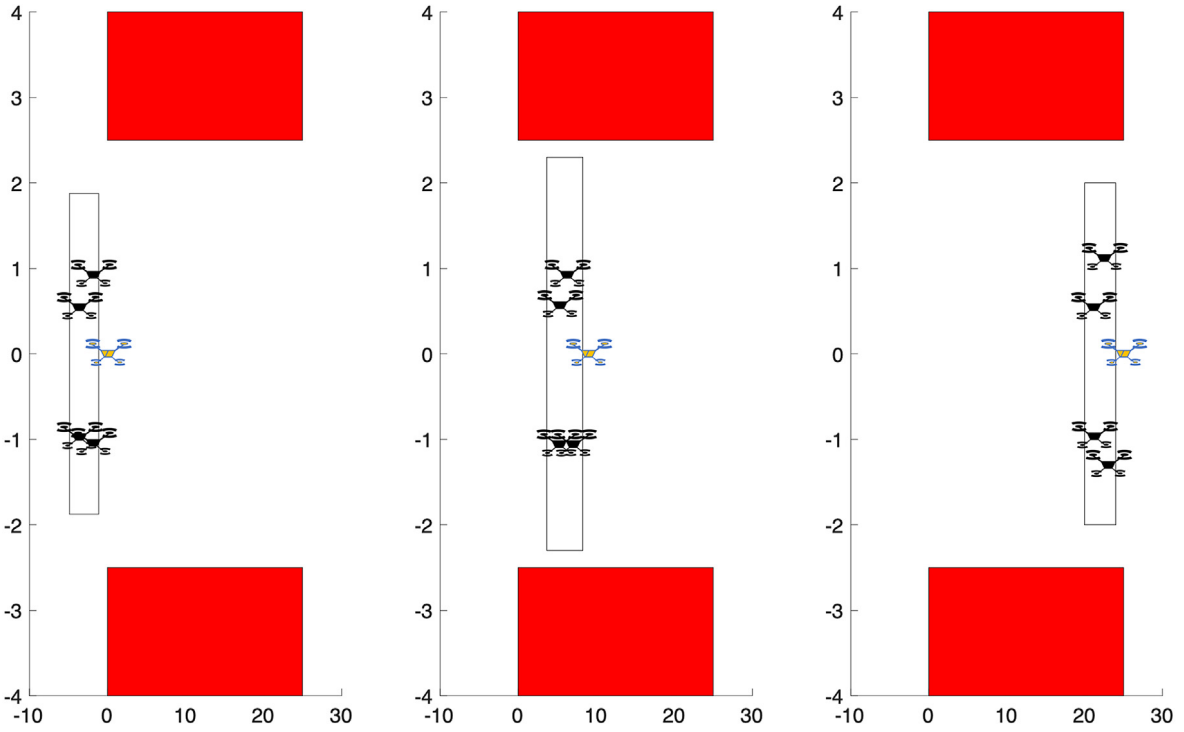

Fig. 4. While in the channel, the system experiences missing data events according to language $\mathcal{L}_3$, but a prefix-based controller can guarantee that the followers (black drones) will travel through the channel without colliding with either wall ($M_2 = 1.3$ and the black outline does not ever touch the red wall).

states where recovery level should be achieved. The main difficulty in this case arises from identifying systems and automata for which a finite memory controller or estimator will be enough to achieve required performance levels.

Acknowledgments

This work is partially supported by the National Science Foundation, USA Graduate Research Fellowship Grant Number DGE 1256260, an Early Career Faculty grant from NASA's Space Technology Research Grants Program, and National Science Foundation, USA grant ECCS-1553873.

Appendix A. Proofs of properties of block lower triangular matrices

Proof of Lemma 1. The first property is a trivial consequence of matrix addition, while the second property follows directly from multiplication of two block lower triangular matrices. Finally, the third property can be observed from the identity for partitioned matrix inversion of block lower triangular matrices. □

Proof of Proposition 1. Let $\bar{C}_0 \doteq \mathcal{BM}_j(\bar{C}^{(1)}) = \mathcal{BM}_j(\bar{C}^{(2)})$. We prove both the sufficient and necessary directions:

Sufficiency: Suppose that $\mathcal{BM}_j(F^{(1)}) = \mathcal{BM}_j(F^{(2)}) = \tilde{F}$. Using the fact that $\bar{C}^{(i)}$, S and $F^{(i)}$ are block lower triangular (and hence, $Q^{(i)}$ is also block lower triangular) for $i \in \{1, 2\}$ as well as Lemma 1, we have:

$$
\begin{aligned}
&\mathcal{BM}_j(Q^{(1)}) \\
&= \mathcal{BM}_j[F^{(1)}(I - \bar{C}^{(1)}SF^{(1)})^{-1}] \\
&= \mathcal{BM}_j(F^{(1)})(\mathcal{BM}_j(I) - \mathcal{BM}_j(\bar{C}^{(1)})\mathcal{BM}_j(S)\mathcal{BM}_j(F^{(1)}))^{-1} \\
&= \tilde{F}(\mathcal{BM}_j(I) - \bar{C}_0\mathcal{BM}_j(S)\tilde{F})^{-1} \\
&= \mathcal{BM}_j(F^{(2)})(\mathcal{BM}_j(I) - \mathcal{BM}_j(\bar{C}^{(2)})\mathcal{BM}_j(S)\mathcal{BM}_j(F^{(2)}))^{-1} \\
&= \mathcal{BM}_j[F^{(2)}(I - \bar{C}^{(2)}SF^{(2)})^{-1}] \\
&= \mathcal{BM}_j(Q^{(2)}).
\end{aligned}
$$

Necessity: Suppose that $\mathcal{BM}_j(Q^{(1)}) = \mathcal{BM}_j(Q^{(2)}) = \tilde{Q}$. First, we note the (strictly) block lower triangular properties of $\bar{C}^{(i)}$, S, $Q^{(i)}$ and $F^{(i)}$. It was shown in [8] that we can solve for $F^{(i)}$, for $i \in \{1, 2\}$ from (1) as:

$$F^{(i)} = (I + Q^{(i)}\bar{C}^{(i)}S)^{-1}Q^{(i)}.$$

Then, using the fact that $\bar{C}^{(i)}$, S, $Q^{(i)}$ and $F^{(i)}$ are block lower triangular for $i \in \{1, 2\}$ and Lemma 1, we find that:

$$\begin{aligned}
&\mathcal{BM}_j(F^{(1)}) \\
&= \mathcal{BM}_j[(I + Q^{(1)}\bar{C}^{(1)}S)^{-1}Q^{(1)}] \\
&= (\mathcal{BM}_j(I) + \mathcal{BM}_j(Q^{(1)})\mathcal{BM}_j(\bar{C}^{(1)})\mathcal{BM}_j(S))^{-1}\mathcal{BM}_j(Q^{(1)}) \\
&= (\mathcal{BM}_j(I) + \tilde{Q}\bar{C}_0\mathcal{BM}_j(S))^{-1}\tilde{Q} \\
&= (\mathcal{BM}_j(I) + \mathcal{BM}_j(Q^{(2)})\mathcal{BM}_j(\bar{C}^{(2)})\mathcal{BM}_j(S))^{-1}\mathcal{BM}_j(Q^{(2)}) \\
&= \mathcal{BM}_j[(I + Q^{(2)}\bar{C}^{(2)}S)^{-1}Q^{(2)}] \\
&= \mathcal{BM}_j(F^{(2)}). \quad \square
\end{aligned}$$

Proof of Proposition 2. The proof is similar to Proposition 1. First, consider the forward direction of the proof (sufficiency):

$$\begin{aligned}
r^{(1)}_{1:jn} &= [(I + Q^{(1)}\bar{C}^{(1)}S)u^{(1)}_0]_{1:jn} \\
&= \mathcal{BM}_j(I + Q^{(1)}\bar{C}^{(1)}S)[u^{(1)}_0]_{1:jn} \\
&= (\mathcal{BM}_j(I) + \mathcal{BM}_j(Q^{(1)})\mathcal{BM}_j(\bar{C}^{(1)})\mathcal{BM}_j(S))[u^{(1)}_0]_{1:jn} \\
&= (\mathcal{BM}_j(I) + \mathcal{BM}_j(Q^{(2)})\mathcal{BM}_j(\bar{C}^{(2)})\mathcal{BM}_j(S))[u^{(2)}_0]_{1:jn} \\
&= \mathcal{BM}_j(I + Q^{(2)}\bar{C}^{(2)}S)[u^{(2)}_0]_{1:jn} \\
&= r^{(2)}_{1:jn},
\end{aligned}$$

where we again applied Lemma 1 and the fact that $\bar{C}^{(i)}$, S and $Q^{(i)}$ are block lower triangular for $i \in \{1, 2\}$. The proof of the opposite direction (necessity) is similar. $\square$

Appendix B. Matrices and sets for estimator and controller synthesis problems

For completeness' sake, we provide the corresponding matrices and sets for the estimator and controller synthesis problems in this section.

In the context of estimator synthesis for the system with missing data in (3) using the estimator structure in (4), the state estimation error system for the state estimation error given by $\xi(t) \doteq x(t) - \hat{x}(t)$ can be found to be of the form in (5) with $B_\xi \doteq I$, $u_\xi(t) \doteq u_e(t)$, $k_\xi \doteq 0$, $\mathcal{U}_\xi \doteq \mathbb{R}^m$, and the transformed output $y_\xi(t) \doteq y(t) - C\hat{x}(t)$ when $q(t) = 1$ and $y_\xi(t) \doteq \emptyset$ when $q(t) = 0$, where $\hat{x}(t)$ is known signal that can computed using (4).

On the other hand, the controller synthesis problem for the system with missing data in (3) is one with the system dynamics of the form in (5) with $B_\xi \doteq B$, $k_\xi \doteq k$, $u_\xi(t) \doteq u(t)$, $y_\xi(t) \doteq y(t)$ and $\xi(t) \doteq x(t)$, as well as $\mathcal{U}_\xi \doteq \mathcal{U}$. Moreover, for the tracking control problem with a given desired trajectory $x_d(t_0), x_d(t_0 + 1), \ldots, x_d(t_0 + T)$ and associated desired inputs $u_d(t_0), u_d(t_0 + 1), \ldots, u_d(t_0 + T - 1)$, the corresponding system dynamics takes the form in (5) with $B_\xi \doteq B$, $k_\xi \doteq 0$, $u_\xi(t) \doteq u(t) - u_d(t)$, $y_\xi(t) \doteq y(t) - Cx_d(t)$ and $\xi(t) \doteq x(t) - x_d(t)$, as well as $\mathcal{U}_\xi \doteq \{u_\xi(t) \in \mathbb{R}^m \mid u_\xi(t) + u_d(t) \in \mathcal{U}\}$.

Appendix C. Proofs of main results

Proof of Theorem 1. This follows directly from Propositions 1 and 2 as well as the invertibility of the mappings (1) and (2). $\square$

Proof of Theorem 2. For a given prefix-based feedback law $\{(F^{(i)}, u^{(i)}_0)\}_{i=1}^{|\mathcal{L}|}$, the transformed state trajectory $\xi^{(i)} = [\xi^{(i)}(t_0)^\intercal, \ldots, \xi^{(i)}(t_0 + T)^\intercal]^\intercal$ under the i-th missing data pattern can be written as a nonlinear function of $\{(F^{(i)}, u^{(i)}_0)\}_{i=1}^{|\mathcal{L}|}$ just by plugging in the transformed input term (9). After applying a change of variables via the mapping in Theorem 1, we can express $\xi^{(i)}$ as a linear function of $\{(Q^{(i)}, r^{(i)})\}_{i=1}^{|\mathcal{L}|} \in \mathcal{Q}(\mathcal{L})$ as in (17). Since the equalized recovery condition can also be written as linear constraints in $\xi^{(i)}$ that should hold for all initial states satisfying M_1 bound and for all possible noise values, problem (21) is a robust linear program, whose feasibility is equivalent to the existence of the desired estimator/controller. Finally, the gains of the prefix-based feedback law are obtained by applying the inverse of the mapping in Theorem 1. $\square$

Proof of Theorem 3. In this proof, we will convert the semi-infinite constraints in Theorem 2 into linear constraints by leveraging the robust optimization approach in [31,32]. Since we will repeat the same robustification process for each $i \in [1, |\mathcal{L}|]$ in (16), we will drop the dependence on i in the following.

We first rewrite the infinity norm expressions in (16) as linear inequalities and substitute the expression for ξ in terms of w, v, $\xi(t_0)$ and r using the equations in Theorem 2, as follows:

$$\|\xi\| \leq M_2 \Rightarrow \begin{bmatrix} I \\ -I \end{bmatrix} \eta = \begin{bmatrix} I \\ -I \end{bmatrix} (G \begin{bmatrix} w \\ v \\ \xi(t_0) \end{bmatrix} + Sr + (I + SQ\bar{C})H\tilde{f}) \leq M_2 \mathbb{1}, \tag{C.1}$$

$$\|R_T\xi\| \le M_1 \Rightarrow \begin{bmatrix} I \\ -I \end{bmatrix} R_T\eta = \begin{bmatrix} I \\ -I \end{bmatrix} R_T(G\begin{bmatrix} w \\ v \\ \xi(t_0) \end{bmatrix} + Sr + (I + SQ\bar{C})H\tilde{f}) \le M_1\mathbb{1}, \tag{C.2}$$

$$\|u_\xi + u_d\| \le \eta_u \Rightarrow \begin{bmatrix} I \\ -I \end{bmatrix}(u_\xi + u_d) = \begin{bmatrix} I \\ -I \end{bmatrix}(\tilde{G}\begin{bmatrix} w \\ v \\ \xi(t_0) \end{bmatrix} + r + Q\bar{C}H\tilde{f} + u_d) \le \eta_u\mathbb{1}, \tag{C.3}$$

$$\begin{matrix} \|w\| \le \eta_w, \\ \|v\| \le \eta_v, \\ \|\xi(t_0)\| \le M_1, \end{matrix} \Rightarrow \begin{bmatrix} I & 0 & 0 \\ -I & 0 & 0 \\ 0 & I & 0 \\ 0 & -I & 0 \\ 0 & 0 & I \\ 0 & 0 & -I \end{bmatrix} \begin{bmatrix} w \\ v \\ \xi(t_0) \end{bmatrix} \le \begin{bmatrix} \eta_w\mathbb{1} \\ \eta_v\mathbb{1} \\ M_1\mathbb{1} \end{bmatrix}, \tag{C.4}$$

where $G \doteq \begin{bmatrix}(I + SQ\bar{C})H & SQN & (I + SQ\bar{C})J\end{bmatrix}$, $\tilde{G} \doteq \begin{bmatrix}Q\bar{C}H & QN & Q\bar{C}J\end{bmatrix}$ and $R_T \doteq \begin{bmatrix}0_{n\times nT} & I_n\end{bmatrix}$. Then, leveraging the robust optimization approach in [31,32], we can convert the constraints of (C.1) and (C.2) "for all disturbances w, v and $\xi(t_0)$", i.e., subject to (C.4), into linear constraints with dual matrix variables Π_1, Π_2 and Π_3 for each $i \in [1, |\mathcal{L}|]$. □

Appendix D. Detectability related proofs

D.1. Proof of Proposition 4

In this proof, we use an unobservable subspace argument to show that for initial conditions starting in two important subsets of the state space, equalized recovery parameters can be found and then by a direct sum argument we can conclude that such parameters can be found for any initial condition in the state space of a detectable system.

First, by assumption, any initial condition in the unobservable subspace $x_0 \in \mathcal{U}_0(q)$ asymptotically converges to zero. Convergence to zero implies that there exists a time $t^{(1)}$ such that the zero estimator (the estimator that always returns the zero vector) satisfies equalized recovery for time horizon $t^{(1)}$, any recovery level M_1, and a finite intermediate level M_2.

Next, note that if there exists a time t' such that $y(t', x_0, q)$ (the output at time t' caused by initial condition x_0) is different from the output of any other initial condition, then equalized recovery is feasible. The estimator that achieves equalized recovery has time horizon t', any recovery level, and a finite intermediate level that depends on the dynamics. By definition, all initial conditions x_0 in the orthogonal complement of the unobservable subspace $x_0 \in \mathcal{U}_0(q)^\perp$ of (25) have such a time $t^{(2)}$ or else contradict the definition of $\mathcal{U}_0(q)^\perp$.

Finally, because any initial condition must be in the direct sum of $\mathcal{U}_0(q)$ and $\mathcal{U}_0(q)^\perp$, the proof is complete. For any initial condition $x_0 = x_0' + x_0''$ where $x_0' \in \mathcal{U}_0(q)$ and $x_0'' \in \mathcal{U}_0(q)^\perp$, equalized recovery is satisfied with time horizon $t^{(1)} + t^{(2)}$, a finite intermediate level, and a recovery level dependent on the choice of $t^{(1)}$ and $t^{(2)}$. □

D.2. Proof of Proposition 5

We show that equalized recovery is indeed a superset of detectability by providing a simple example. Consider the following scalar linear system:

$$\begin{aligned} x(t+1) &= x(t), \\ y(t) &= 0. \end{aligned}$$

It is trivial to see that this system does not satisfy detectability according to Definition 4 because $x(t, x_0, q)$ for any $x_0 \neq 0$ does not tend to zero. However, equalized recovery is trivially satisfied for any T, M_1 and $M_2 \ge M_1$. □

References

[1] W. Zhang, M.S. Branicky, S.M. Phillips, Stability of networked control systems, IEEE Control Syst. 21 (1) (2001) 84–99.
[2] B. Sinopoli, L. Schenato, M. Franceschetti, K. Poolla, M.I. Jordan, S.S. Sastry, Kalman filtering with intermittent observations, IEEE Trans. Automat. Control 49 (9) (2004) 1453–1464.
[3] S. Amin, A.A. Cárdenas, S.S. Sastry, Safe and secure networked control systems under denial-of-service attacks, in: R. Majumdar, P. Tabuada (Eds.), Hybrid Systems: Computation and Control, Springer Berlin Heidelberg, Berlin, Heidelberg, 2009, pp. 31–45.
[4] K. Habib, ODI Resume Report on Investigation PE 16-007 concerning Tesla automatic vehicle control systems, Tech. rep., NHTSA Office of Defects Investigation, 2016.
[5] F. Blanchini, Set invariance in control, Automatica 35 (11) (1999) 1747–1767.
[6] F. Blanchini, M. Sznaier, A convex optimization approach to synthesizing bounded complexity ℓ_∞ filters, IEEE Trans. Automat. Control 57 (1) (2012) 216–221.
[7] K.J. Rutledge, S.Z. Yong, N. Ozay, Optimization-based design of bounded-error estimators robust to missing data, in: Analysis and Design of Hybrid Systems ADHS, IFAC-PapersOnLine (2018) ADHS Full Papers.
[8] J. Skaf, S.P. Boyd, Design of affine controllers via convex optimization, IEEE Trans. Automat. Control 55 (11) (2010) 2476–2487.

[9] R.M. Jungers, A. Kundu, W.P.M.H. Heemels, Observability and controllability analysis of linear systems subject to data losses, IEEE Trans. Automat. Control 63 (10) (2018) 3361–3376.
[10] K.J. Rutledge, S.Z. Yong, N. Ozay, Prefix-based Bounded-error Estimation with Intermittent Observations, in: Proc. Amer. Control Conf., 2019, pp. 4320–4325.
[11] A. Bemporad, M. Heemels, M. Johansson, et al., Networked Control Systems, Vol. 406, Springer, 2010.
[12] Z. Jin, C. Ko, R.M. Murray, Estimation for nonlinear dynamical systems over packet-dropping networks, in: 2007 American Control Conference, 2007, pp. 5037–5042, http://dx.doi.org/10.1109/ACC.2007.4283063.
[13] G. Battistelli, A. Benavoli, L. Chisci, State estimation with remote sensors and intermittent transmissions, Systems Control Lett. 61 (1) (2012) 155–164, http://dx.doi.org/10.1016/j.sysconle.2011.10.005, URL http://www.sciencedirect.com/science/article/pii/S016769111100260X.
[14] P. Seiler, R. Sengupta, Analysis of communication losses in vehicle control problems, in: American Control Conference, vol. 2, 2001, pp. 1491–1496.
[15] K. Gatsis, M. Pajic, A. Ribeiro, G.J. Pappas, Opportunistic control over shared wireless channels, IEEE Trans. Automat. Control 60 (12) (2015) 3140–3155.
[16] N. Jia, Y.-Q. Song, R.-Z. Lin, Analysis of networked control system with packet drops governed by (m,k)-firm constraint, in: 6th IFAC International Conference on Fieldbus Systems and their Applications, IFAC Proc. Vol. 38 (2) (2005) 63–70, http://dx.doi.org/10.3182/20051114-2-MX-3901.00010.
[17] F. Felicioni, N. Jia, Y.-Q. Song, F. Simonot-Lion, Impact of a (m,k)-firm data dropouts policy on the quality of control, in: 6th IEEE International Workshop on Factory Communication Systems, in: Factory Communication Systems, 2006 IEEE International Workshop on, IEEE, Torino, Italy, 2006, pp. 353–359.
[18] S.M. Hassaan, Q. Shen, S.Z. Yong, Bounded-Error Estimator Design with Missing Data Patterns via State Augmentation, in: Proc. Amer. Control Conf., 2019, pp. 447–452.
[19] A. Colombo, M. Bahraini, P. Falcone, Measurement scheduling for control invariance in networked control systems, in: 2018 IEEE Conference on Decision and Control (CDC), IEEE, 2018, pp. 3361–3366.
[20] A. Aspeel, D. Dasnoy, R.M. Jungers, B. Macq, Optimal intermittent measurements for tumor tracking in x-ray guided radiotherapy, in: Medical Imaging 2019: Image-Guided Procedures, Robotic Interventions, and Modeling, vol. 10951, International Society for Optics and Photonics, 2019, pp. 109510C.
[21] D. Bertsekas, I. Rhodes, Recursive state estimation for a set-membership description of uncertainty, IEEE Trans. Automat. Control 16 (2) (1971) 117–128.
[22] M. Milanese, A. Vicino, Optimal estimation theory for dynamic systems with set membership uncertainty: an overview, Automatica 27 (6) (1991) 997–1009.
[23] J.S. Shamma, K.-Y. Tu, Set-valued observers and optimal disturbance rejection, IEEE Trans. Automat. Control 44 (2) (1999) 253–264.
[24] J. Lee, G. Dullerud, Optimal disturbance attenuation for discrete-time switched and Markovian jump linear systems, SIAM J. Control Optim. 45 (4) (2006) 1329–1358, http://dx.doi.org/10.1137/050627538, arXiv:https://doi.org/10.1137/050627538.
[25] F. Blanchini, M. Sznaier, A convex optimization approach to fixed-order controller design for disturbance rejection in SISO systems, IEEE Trans. Automat. Control 45 (4) (2000) 784–789, http://dx.doi.org/10.1109/9.847123.
[26] P.J. Goulart, E.C. Kerrigan, Output feedback receding horizon control of constrained systems, Internat. J. Control 80 (1) (2007) 8–20.
[27] D. Del Vecchio, A partial order approach to discrete dynamic feedback in a class of hybrid systems, in: International Workshop on Hybrid Systems: Computation and Control, Springer, 2007, pp. 159–173.
[28] R. Verma, D. Del Vecchio, Safety control of hidden mode hybrid systems, IEEE Trans. Automat. Control 57 (1) (2012) 62–77, http://dx.doi.org/10.1109/TAC.2011.2150370.
[29] N. Athanasopoulos, K. Smpoukis, R.M. Jungers, Invariant sets analysis for constrained switching systems, IEEE Control Syst. Lett. 1 (2) (2017) 256–261, http://dx.doi.org/10.1109/LCSYS.2017.2714840.
[30] O. Mickelin, N. Ozay, R.M. Murray, Synthesis of correct-by-construction control protocols for hybrid systems using partial state information, in: American Control Conference, 2014, pp. 2305–2311.
[31] A. Ben-Tal, L. El Ghaoui, A. Nemirovski, Robust Optimization, Princeton University Press, 2009.
[32] D. Bertsimas, D.B. Brown, C. Caramanis, Theory and applications of robust optimization, SIAM Rev. 53 (3) (2011) 464–501.
[33] S. Sadraddini, R. Tedrake, Linear Encodings for Polytope Containment Problems, 2019.
[34] Gurobi Optimization LLC, Gurobi optimizer reference manual, 2019, http://www.gurobi.com.
[35] M. Tillerson, L. Breger, J.P. How, Distributed coordination and control of formation flying spacecraft, in: Proc. Amer. Control Conf., vol. 2, 2003, pp. 1740–1745.