2022 IEEE 61st Conference on Decision and Control (CDC)
December 6-9, 2022. Cancún, Mexico

System-Level Recurrent State Estimators for Affine Systems Subject to Data Losses Modeled by Automata

Syed M. Hassaan and Sze Zheng Yong

2022 IEEE 61st Conference on Decision and Control (CDC) | 978-1-6654-6761-2/22/$31.00 ©2022 IEEE | DOI: 10.1109/CDC51059.2022.9992499

Abstract— **This paper proposes a robust output feedback state estimator for uncertain/bounded-error affine systems subject to data losses modeled by an automaton. Specifically, by introducing a novel property known as recurrent recovery, where the estimation errors are required to be recurrent to some minimum recovery levels at each node of the data loss automata, we design a robust estimator design that guarantees that the state estimation errors remain bounded in a recurrent manner despite worst-case realizations of process and sensor noise/uncertainties in addition to missing data. Our design can directly deal with infinite-horizon missing data specifications modeled by automata by recasting the problem into multiple finite-horizon problems of varying lengths, which results in an optimization-based approach with only a finite number of constraints. Moreover, our design is built upon system-level parameterization and for this purpose, we propose a novel affine output feedback strategy that also contributes to the literature of finite-horizon optimal control.**

I. Introduction

State estimation plays a crucial role in many engineering and control applications since system states may not be directly measured by sensors or the sensor measurements may be noisy. A great deal of current state estimators heavily rely on the availability of the sensor measurements to properly operate. However, as cyber-physical systems such as autonomous vehicles, smart buildings, power grids, etc. become increasingly integrated and networked, significant packet drops or missing data may be inevitable and this may lead to performance deterioration or even instability/malfunctions. Thus, the possibility of missing/lossy data needs to be addressed in estimator designs.

Literature Review. State estimation for networked systems that are susceptible to packet drops has been primarily investigated using a probabilistic and Bayesian filtering framework, where intermittent data is often assumed to follow some known probability distributions, e.g., [1], [2], [3], [4], [5]. Moreover, these methods are usually focused on obtaining the best average/expected estimates as opposed to achieving best worst-case or robust estimation errors when knowledge of the probability distributions is unavailable or if the missing data phenomenon is not stochastic, e.g., due to sensor glitches, occlusions, or denial of service attacks [6].

On the other hand, set-valued estimators have been proposed to consider the worst-case/robust estimation performance, e.g., [7], [8], [9]. In the context of systems with missing observations, [10], [11] modeled the feasible missing data patterns with a fixed-length language and proposed finite-horizon affine estimators to satisfy an equalized recovery property, which implies that within a finite time horizon, especially for times when observations may go missing, the estimation error can have a more relaxed upper bound, but by the end of the horizon must recover the initial upper bound. However, this approach does not directly apply for infinite horizon problems or missing data models with time-varying lengths such as (m, k)-firmness [12] that indicates that at least m out of k consecutive measurements are available, or more general formal specification using automata [13].

Moreover, similar to our prior work [10], [11], our estimator design will rely on finite-horizon optimal control. Thus, another set of relevant literature pertains to optimal control parameterizations such as output feedback, e.g., [14], output error (or disturbance) feedback, e.g., [15], and system-level synthesis, e.g., [16], [17]. In particular, while the first two parameterizations have been explored for estimation/control with missing data, to our best knowledge, designs based on system-level synthesis have not been investigated. Moreover, [16] has only considered linear state and output feedback laws, while [17] has recently considered affine state feedback; thus, to our knowledge, system-level affine output feedback designs have also not been studied in the literature.

Contributions. In this paper, we propose a system-level synthesis-based robust state estimator for affine systems subject to data losses modeled by automata as described in more detail in [13]. In particular, we introduce a novel property known as *recurrent recovery* that extends the notions of equalized recovery [10], [18] and equalized performance [19], where the estimation errors are required to be recurrent at each node of the data loss automaton. Using this notion, we then present an estimator design that guarantees recurrent recovery of the estimation errors for uncertain/bounded-error affine systems with missing data despite worst-case realizations of noise and uncertainties. Our design relies on translating the infinite-length missing data signal that satisfy an automaton into multiple finite-horizon problems with varying horizon lengths, resulting in optimization problems with only a finite number of constraints that can be solved using off-the-shelf solvers. Moreover, our design is built upon system-level parameterization-based affine output feedback that is on its own a novel contribution to the literature of finite-horizon optimal control and we further show that missing data specifications can be somewhat easily encoded as simple constraints on the system response matrices. Finally, we demonstrate the usefulness of our proposed robust estimator design in simulation despite missing data modeled by an automaton representing (m,k)-firmness.

II. Problem Formulation

Notations. $\|\cdot\|$ denotes the ∞-norm, and $\mathbb{1}$, 0 and I are used to represent a vector of ones, a matrix of zeros and an identity matrix of appropriate dimensions, respectively, whereas $\mathbb{1}_n$ is used for a vector of ones of length n and I_n is used for an $n \times n$ identity matrix.

S. M. Hassaan is with the School for Engineering of Matter, Transport and Energy, Arizona State University, Tempe, AZ, USA (shassaan@asu.edu) and S. Z. Yong is with the Department of Mechanical and Industrial Engineering, Northeastern University, Boston, MA, USA (s.yong@neu.edu).

This work was supported in part by NSF grant CNS-1943545 and NASA Early Career Faculty grant 80NSSC21K0071.

A. System Dynamics and Data Loss Model

System Dynamics: We consider a discrete-time linear time-varying system subject to process noise and output noise. The model of the system dynamics is described as follows:

$$\begin{aligned} x_{k+1} &= Ax_k + Bu_k + Ww_k, \\ y_k &= \begin{cases} Cx_k + Vv_k, & q_k = 1, \\ \emptyset, & q_k = 0, \end{cases} \end{aligned} \tag{1}$$

where $x_k \in \mathbb{R}^n$, $u_k \in \mathbb{R}^m$ and $w_k \in \mathbb{R}^{n_w}$ represent the system states, inputs and process noise respectively at time k. $y_k \in \mathbb{R}^p$ and $v_k \in \mathbb{R}^{n_w}$ denote the model output obtained by the sensors and the measurement noise at time k, respectively, and q_k represents whether the sensor data is available ($q_k = 1$) or missing/absent ($q_k = 0$). We assume that w_k and v_k are bounded with $\|w_k\| \leq \eta_w$ and $\|v_k\| \leq \eta_v$ for all k. The system matrices A, B, C, W, V, η_w and η_v are all known. Without loss of generality, we assume that the initial time is $k = 0$.

With regards to the missing data specification, instead of using probabilistic models, we consider models that are constrained by some temporal logic specifications. An example specification is (m, k)-firmness, which specifies that within k consecutive time steps, there are at most m instances of data loss. Another specification could be that there can be at most l consecutive packet losses, where l is known. Moreover, an obvious assumption, to ensure that the system is observable, is that $q_k \neq 0 \ \forall k$. To model such types of missing data specifications, researchers in [13] proposed the use of automata, which we will adopt in this work. Further, for the system dynamics described in (1), [13] covers in detail the observability conditions that need to hold.

Definition 1 (Data Loss/Missing Data Automaton)**.** *An automaton is defined as a triplet $\mathcal{A} = \{E, S, \mathcal{E}\}$ where S is a set of* nodes *(representing discrete states/modes) of cardinality $|S| = N$, $E \in \{0, 1\}^{N \times N}$ is a* transition matrix*, where $E(s, s') = 1$, $s, s' \in S$ represents an allowed transition and 0 otherwise, and $\mathcal{E}$ contains the* events *associated with each allowed transition.*

To give an idea of construction of an automaton based on some specification, we provide a simple (m, k)-firmness based automaton below:

Example 1. *A missing data specification of $(1, 3)$-firmness, i.e., within a window of 3 consecutive time steps, data from at most 1 step is missing, can be captured by the automaton in Figure 1. The nodes correspond to the "state" of the data memory for the past three time steps. For instance, $s_1 = 111$ represents the memory state that the data for the past 3 time steps, i.e., at time steps $t-2$, $t-1$ and t, are available and $s_2 = 110$ represents the case when the data for $t-2$ and $t-1$ are available/present while the data for t is missing/lost, with the edge connecting the two states representing the* event *'0'$\in \mathcal{E}$ that the data at time t is missing/not received.*

Next, we label/designate certain nodes and paths of the automaton that will be useful later for our estimator design.

Definition 2 (Recurrent Node and Recurrent Node Set)**.** *Any node in an automaton $\mathcal{A}$ (cf. Definition 1) with an* out-degree *(i.e., the number of outgoing edges) that is strictly greater than 1 is called a* recurrent node*, denoted as s_R. A set containing all recurrent nodes defined in $\mathcal{A}$ is called a* recurrent node set $S_R \subseteq S$. *In the special case where $\mathcal{A}$ consists of only one loop, either if it is a single node with a self-loop or a set of nodes in a single loop, the recurrent node set is then $S_R = S$.*

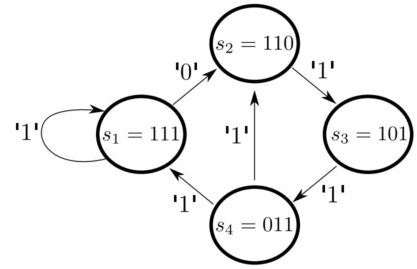

Fig. 1: Automaton for $(1, 3)$-firm signal.

Example 2. *For the automaton $\mathcal{A}$ given in Figure 1, its recurrent node set is then $S_R = \{s_1, s_4\}$, since both s_1 and s_4 have two outgoing edges.*

Definition 3 (Direct Path and Path Set)**.** *A finite sequence of nodes $d_a^b = \{\{s(t)\}_{t=1}^{T_a^b} \in S^N \mid E(s(t), s(t+1)) = 1\}$ of length T_a^b of an automaton $\mathcal{A}$ (cf. Definition 1) is called a* direct path *(between recurrent nodes) if the following conditions hold: (1) $s(1) = s_a \in S_R$, i.e., the first node in the sequence is a recurrent node, (2) $s(T_a^b) = s_b \in S_R$, i.e., the last node in the sequence is a recurrent node, and (3) $s(t) \notin S_R$ for all $1 < t < T_a^b$, i.e., none of the intermediate nodes in the sequence is a recurrent node.*
A path set $\mathcal{D}$ is then a set containing all possible direct paths in an automaton $\mathcal{A}$.

Example 3. *For the automaton $\mathcal{A}$ given in Figure 1, there are four possible direct paths: $d_1^1 = \{s_1, s_1\}$, $d_1^4 = \{s_1, s_2, s_3, s_4\}$, $d_4^1 = \{s_4, s_1\}$, and $d_4^4 = \{s_4, s_2, s_3, s_4\}$. Moreover, the path set is then $\mathcal{D} = \{d_1^1, d_1^4, d_4^1, d_4^4\}$.*

For simplicity, we assume the following:

Assumption 1. *The missing data signal is initialized at a node that is on a direct path of the automaton and the automaton does not have a terminal node (i.e., no deadlocks).*

Note that this assumption is without any loss of generality since it can be relaxed by designating the initial and terminal nodes as recurrent nodes and including the corresponding direct paths to the path set $\mathcal{D}$.

In the following proposition, we introduce certain properties of the direct paths and the path set:

Proposition 1 (Properties of Direct Path and Path Set)**.** *For every direct path $d_a^b \in \mathcal{D}$, the following are true:*
1) No two direct paths have overlapping transitions.
2) The combination of all paths $d_a^b \in \mathcal{D}$ in the path set forms the automaton.

Proof. To prove 1), suppose there are two paths $d_a^b, d_c^d \in \mathcal{D}$ that have at least one overlapping transition. Since all of the intermediate nodes are non-recurrent, it is possible only if $s_a = s_c$ and $s_b = s_d$. Hence $d_a^b = d_c^d$.

For 2), suppose there is a path $d_a^b \notin \mathcal{D}$ formed from a sequence of transitions. The only conditions for which any node on the path does not lie on any direct path of the automaton are: (i) $s_a \notin S_R$, or (ii) $s_b \notin S_R$. However, these are not possible by Assumption 1. Since there is no valid condition that results in a $d_a^b \notin \mathcal{D}$, hence all direct paths in path set can be combined to form the whole automaton. ■

As a result of item 1) in Proposition 1, each non-recurrent node of the automaton appears in exactly one direct path. And item 2) in Proposition 1 implies that the missing data signal q_k always lies on some direct path of the automaton.

B. Recurrent Recovery

Next, in view of the goal of this paper to maintain estimation error bounds below prescribed bounds despite data losses/packet drops, we prescribe *recovery levels* representing the maximum estimation error bound to each *recurrent node* $s \in S_R$ (cf. Definition 2) and define the property of *recurrent recovery* for direct paths (cf. Definition 3), where the estimation error must start from and *recover* to the *recovery levels*. Formally, this is defined as follows:

Definition 4 (Path Recurrent Recovery). *Given a direct path $d_a^b \in \mathcal{D}$ of length T_a^b in an automaton-based missing-data model $\mathcal{A}$ with* recovery levels *(i.e., estimation error bounds) of the initial recurrent node at time step* 0 *and final recurrent node at time step $k_{T_a^b} = T_a^b$ being μ_a and μ_b, respectively, an estimator achieves* path recurrent recovery *if for any initial estimation error $\tilde{x}_0$ satisfying $\|\tilde{x}_0\| \leq \mu_a$, the final estimation error satisfies $\|\tilde{x}_k\| \leq \mu_k$ for all $0 \leq k \leq T_a^b$ and $\|\tilde{x}_{T_a^b}\| \leq \mu_b$, where $\tilde{x}_k \triangleq x_k - \hat{x}_k$ is the estimation error with $\hat{x}_k$ being the state estimate, and μ_k for all $0 < k < T_a^b$ are the intermediate error bounds/recovery levels associated with the non-recurrent nodes $s(k) \in S \setminus S_R$ on the direct path d_a^b.*

Note that if μ_a and μ_b in the above definition are equal, then the definition coincides with *equalized recovery* that was defined in [10], [18]; hence *path recurrent recovery* is a slight generalization of *equalized recovery*.

Now, the overall *recurrent recovery* property of an estimator can be established by extending the path recurrent recovery property to all possible paths in $\mathcal{D}$.

Definition 5 (Recurrent Recovery). *An estimator is said to have the property of* recurrent recovery *in the presence of missing data modeled by an automaton $\mathcal{A}$ if it achieves the* path recurrent recovery *property in Definition 4 for all direct paths $d_a^b \in \mathcal{D}$ of $\mathcal{A}$.*

C. Problem Statement

With the system and the missing-data model introduced, and equipped by the above definitions, the problem we seek to solve in this paper can be cast as follows:

Problem 1 (Estimator Design with Missing Data). *Given the system dynamics* (1) *and a missing data model specified by an automaton $\mathcal{A}$, design an optimal state estimator with estimate $\hat{x}_k$ and estimation error $\tilde{x}_k = x_k - \hat{x}_k$ that minimizes a cost $J(\{\mu_i\}_{i=1}^N)$ (representing the aggregate estimation errors for the nodes of $\mathcal{A}$) subject to the property of recurrent recovery in Definition 5 being satisfied, where μ_i are finite estimation error bounds associated with the corresponding nodes $s_i \in S$.*

Consequently, from the solution of Problem 1, if the initial node is known and the initial estimation error is below the corresponding estimation error bound, then the estimation error of the estimator is guaranteed to always satisfy the estimation error bounds corresponding to the nodes that the system is in. Finally, if the initial node is unknown or if the initial estimation is high, we can initialize the estimation process with any asymptotic estimator until the current node is inferred and the estimation error is below the associated estimation error bound.

III. System-Level Recurrent Recovery State Estimator Design

In this section, we propose an estimator synthesis approach that is inspired by system-level synthesis in [16], [17] to solve Problem 1. We do this by first designing a solution without taking into consideration any data losses (i.e., the perfect observation case), followed by the design modifications to cater to the problem involving data losses. Note that the proposed system-level state estimator for the perfect observation case is obtained by first translating the problem into an equivalent finite-horizon optimal control problem with affine output feedback law, which itself is an extension of the approaches in [16] that only considered linear output feedback laws and in [17] that considered affine state feedback laws, and is thus a contribution of this paper. On the other hand, our treatment of the case with missing data/dropped packet is an extension of our prior works [20], [10], [21], [11], [18] that considered only fixed-length languages to the case where the formal specifications are infinite-length and modeled by a missing data automaton.

A. Perfect Observation Case with Fixed Time Horizon

We first focus on designing an estimator with a fixed horizon T when there is no missing data (i.e., the perfect observation case) of the following form:

$$\begin{aligned} \hat{x}_{k+1} &= A\hat{x}_x + Bu_k - u_{e,k}, \\ \hat{y} &= C\hat{x}_k, \end{aligned} \tag{2}$$

where $\hat{x}_k \in \mathbb{R}^n$ is the estimate of x_k and $u_{e,k} \in \mathbb{R}^n$ is an affine causal output feedback term defined as:

$$u_{e,k} = \nu_k + \sum_{i=0}^{k} M_{(k,i)}\tilde{y}_k, \tag{3}$$

where $M_{(k,i)} \in \mathbb{R}^{n\times p}$ and $\nu_k \in \mathbb{R}^n$ are design gains, and $\tilde{y}_k = y_k - \hat{y}_k$ is the output error between the system and the estimator outputs. With the system defined in (1) and the estimator (2), we can define the estimation error as $\tilde{x}_k \triangleq x_k - \hat{x}_k$ and obtain the corresponding estimation error system:

$$\begin{aligned} \tilde{x}_{k+1} &= A\tilde{x}_x + B_e u_{e,k} + Ww_k, \\ \tilde{y}_k &= C\tilde{x}_k + Vv_k, \end{aligned} \tag{4}$$

where $B_e = I_n$. Thus, we have converted the optimal estimator design problem to a finite-horizon optimal control problem with output feedback for the estimation error system with $u_{e,k}$ as the "control" input with an affine output feedback law in (3) with feedback gains $M_{(k,i)}$ and ν_k.

Next, we define $\tilde{\mathbf{x}}$, $\mathbf{u}_e$, $\tilde{\mathbf{y}}$, $\mathbf{w}$, and $\mathbf{v}$ as the time-concatenated estimation error states, estimator "inputs", output errors, and process and measurement noises over the horizon T, respectively, and $\boldsymbol{\delta}$ as the initial state x_0 concatenated with the disturbance sequence $\mathbf{w}$:

$$\begin{gathered} \tilde{\mathbf{x}} = \begin{bmatrix}\tilde{x}_0^\top & \tilde{x}_1^\top & \dots & \tilde{x}_T^\top\end{bmatrix}^\top, \mathbf{u}_e = \begin{bmatrix}u_{e,0}^\top & u_{e,1}^\top & \dots & u_{e,T-1}^\top\end{bmatrix}^\top, \\ \tilde{\mathbf{y}} = \begin{bmatrix}\tilde{y}_0^\top & \tilde{y}_1^\top & \dots & \tilde{y}_{T-1}^\top\end{bmatrix}^\top, \mathbf{w} = \begin{bmatrix}w_0^\top & w_1^\top & \dots & w_{T-1}^\top\end{bmatrix}^\top, \\ \mathbf{v} = \begin{bmatrix}v_0^\top & v_1^\top & \dots & v_{T-1}^\top\end{bmatrix}^\top, \boldsymbol{\delta} = \begin{bmatrix}x_0^\top & \mathbf{w}^\top\end{bmatrix}^\top. \end{gathered}$$

The dynamics can then be compactly defined as trajectories over the horizon T:

$$\begin{aligned} \tilde{\mathbf{x}} &= \mathbf{A}\tilde{\mathbf{x}} + \mathbf{B}\mathbf{u}_e + \mathbf{D}\boldsymbol{\delta}, \\ \tilde{\mathbf{y}} &= \mathbf{C}\tilde{\mathbf{x}} + \mathbf{V}\mathbf{v}, \end{aligned} \tag{5}$$

with $\mathbf{A} = \begin{bmatrix} 0 & 0 \\ I_T \otimes A & 0 \end{bmatrix}$, $\mathbf{B} = \begin{bmatrix} 0 \\ I_T \otimes B_e \end{bmatrix} = \begin{bmatrix} 0 \\ I_{nT} \end{bmatrix}$, $\mathbf{D} = \begin{bmatrix} I_n & 0 \\ 0 & I_T \otimes W \end{bmatrix}$, $\mathbf{C} = [I_T \otimes C \;\; 0]$, and $\mathbf{V} = I_T \otimes V$, where $\otimes$ is the Kronecker product. Moreover, the estimator "inputs" can be written as:

$$\mathbf{u}_e = \mathbf{M}\tilde{\mathbf{y}} + \boldsymbol{\nu} \tag{6}$$

with $\mathbf{M} = \begin{bmatrix} M_{0,0} & 0 & \dots & 0 \\ M_{1,0} & M_{1,1} & \dots & 0 \\ \vdots & \vdots & \ddots & 0 \\ M_{T-1,0} & M_{T-1,1} & \dots & M_{T-1,T-1} \end{bmatrix}$ and $\boldsymbol{\nu} = \begin{bmatrix} \nu_0 \\ \nu_1 \\ \vdots \\ \nu_{T-1} \end{bmatrix}$.

Next, inspired by [17] to obtain the case with affine feedback term in (3), we further define expanded estimation error states $\bar{\mathbf{x}} = \begin{bmatrix} \mathbb{1}_{pT}^\top & \tilde{\mathbf{x}}^\top \end{bmatrix}^\top$, output errors $\bar{\mathbf{y}} = \begin{bmatrix} \mathbb{1}_{pT}^\top & \tilde{\mathbf{y}}^\top \end{bmatrix}^\top$, and disturbance $\bar{\boldsymbol{\delta}} = \begin{bmatrix} \mathbb{1}_{pT}^\top & \boldsymbol{\delta}^\top \end{bmatrix}^\top$, and represent the expanded dynamics and estimator "inputs" in (5) and (6) as follows:

$$\begin{aligned} \bar{\mathbf{x}} &= \bar{A}\bar{\mathbf{x}} + \bar{B}\mathbf{u}_e + \bar{D}\bar{\boldsymbol{\delta}}, \\ \bar{\mathbf{y}} &= \bar{C}\bar{\mathbf{x}} + \bar{V}\mathbf{v}, \\ \mathbf{u}_e &= \bar{M}\bar{\mathbf{y}}, \end{aligned} \tag{7}$$

with $\bar{A} = \begin{bmatrix} 0 & 0 \\ 0 & \mathbf{A} \end{bmatrix}$, $\bar{B} = \begin{bmatrix} 0 \\ \mathbf{B} \end{bmatrix}$, $\bar{D} = \begin{bmatrix} I_{pT} & 0 \\ 0 & \mathbf{D} \end{bmatrix}$, $\bar{C} = \begin{bmatrix} I_{pT} & 0 \\ 0 & \mathbf{C} \end{bmatrix}$, $\bar{V} = \begin{bmatrix} 0 \\ \mathbf{V} \end{bmatrix}$, and $\bar{M} = [\bar{\nu} \;\; \mathbf{M}]$, where we decompose the affine term $\boldsymbol{\nu}$ as follows: $\boldsymbol{\nu} = \bar{\nu}\mathbf{1}_{p(T-1)}$.

Then, analogous to the output feedback case for system-level parameterization in [16], we can find the finite-horizon system response $\{\boldsymbol{\Phi}_{xx}, \boldsymbol{\Phi}_{ux}, \boldsymbol{\Phi}_{xy}, \boldsymbol{\Phi}_{uy}\}$ from perturbations (δ_x, δ_y), with $\delta_x = \bar{D}\bar{\boldsymbol{\delta}}$ and $\delta_y = \bar{V}\mathbf{v}$, to the expanded estimation error state and estimator "input" $(\bar{x}, \mathbf{u}_e)$ via the following relation:

$$\begin{bmatrix} \bar{\mathbf{x}} \\ \mathbf{u}_e \end{bmatrix} = \begin{bmatrix} \boldsymbol{\Phi}_{xx} & \boldsymbol{\Phi}_{xy} \\ \boldsymbol{\Phi}_{ux} & \boldsymbol{\Phi}_{uy} \end{bmatrix} \begin{bmatrix} \delta_x \\ \delta_y \end{bmatrix}, \tag{8}$$

where, by substituting $\mathbf{u}_e = \bar{M}\bar{\mathbf{y}}$ into the expanded dynamics in (7) and noticing that $I - \bar{A} - \bar{B}\bar{M}\bar{C}$ is invertible since $\bar{M}$ is proper, we can find the system response in (8) as

$$\begin{aligned} \boldsymbol{\Phi}_{xx} &= (I - \bar{A} - \bar{B}\bar{M}\bar{C})^{-1}, \\ \boldsymbol{\Phi}_{xy} &= (I - \bar{A} - \bar{B}\bar{M}\bar{C})^{-1}\bar{B}\bar{M}, \\ \boldsymbol{\Phi}_{ux} &= \bar{M}\bar{C}(I - \bar{A} - \bar{B}\bar{M}\bar{C})^{-1}, \\ \boldsymbol{\Phi}_{uy} &= (\bar{M}\bar{C}(I - \bar{A} - \bar{B}\bar{M}\bar{C})^{-1}\bar{B} + I)\bar{M}, \end{aligned} \tag{9}$$

where $\boldsymbol{\Phi}_{uy}$, $\boldsymbol{\Phi}_{xx}$ and $\boldsymbol{\Phi}_{xy}$ have the following structure:

$$\boldsymbol{\Phi}_{xx} = \begin{bmatrix} I_{pT} & 0 \\ \Phi_{xx,1} & \Phi_{xx,2} \end{bmatrix}, \boldsymbol{\Phi}_{xy} = \begin{bmatrix} 0 & 0 \\ \Phi_{xy,1} & \Phi_{xy,2} \end{bmatrix}, \quad \boldsymbol{\Phi}_{uy} = [\Phi_{uy,1} \;\; \Phi_{uy,2}]. \tag{10}$$

Equipped by the above, we now present a lemma for finding the observer gain $\bar{M}$ such that the two system responses (7) and (8) are equivalent. It is based on a finite-horizon variant of the system-level parameterization in [16].

Lemma 1 (System-Level Parameterization)**.** *Consider the estimation error system over a horizon T with output feedback gain $\bar{M}$ defining the output feedback term as $\mathbf{u}_e = \bar{M}\bar{\mathbf{y}} = \mathbf{M}\tilde{\mathbf{y}} + \boldsymbol{\nu}$. The following statements are true:*

1) *The affine subspace defined by*

$$\begin{bmatrix} I_\tau - \bar{A} & -\bar{B} \end{bmatrix} \begin{bmatrix} \boldsymbol{\Phi}_{xx} & \boldsymbol{\Phi}_{xy} \\ \boldsymbol{\Phi}_{ux} & \boldsymbol{\Phi}_{uy} \end{bmatrix} = [I_\tau \;\; 0], \tag{11}$$

$$\begin{bmatrix} \boldsymbol{\Phi}_{xx} & \boldsymbol{\Phi}_{xy} \\ \boldsymbol{\Phi}_{ux} & \boldsymbol{\Phi}_{uy} \end{bmatrix} \begin{bmatrix} I_\tau - \bar{A} \\ -\bar{C} \end{bmatrix} = \begin{bmatrix} I_\tau \\ 0 \end{bmatrix}, \tag{12}$$

parameterizes all system responses (8) with $\tau \triangleq pT + n(T+1)$.

2) *For any block lower triangular matrices $\{\boldsymbol{\Phi}_{xx}, \boldsymbol{\Phi}_{ux}, \boldsymbol{\Phi}_{xy}, \boldsymbol{\Phi}_{uy}\}$ that satisfy the structure in (10), the feedback gain $\bar{M} \triangleq [\bar{\nu} \;\; \mathbf{M}] = \boldsymbol{\Phi}_{uy} - \boldsymbol{\Phi}_{ux}\boldsymbol{\Phi}_{xx}^{-1}\boldsymbol{\Phi}_{xy}$ (and $\boldsymbol{\nu} = \bar{\nu}\mathbb{1}_{p(T-1)}$) achieves the desired system response.*

Proof. For the first part, taking the left hand side of (11) column-wise and substituting $\boldsymbol{\Phi}_{xx}$, $\boldsymbol{\Phi}_{xy}$, $\boldsymbol{\Phi}_{ux}$ and $\boldsymbol{\Phi}_{uy}$ with their expressions in (9) yield

$$\begin{aligned} &(I - \bar{A})\boldsymbol{\Phi}_{xx} - \bar{B}\boldsymbol{\Phi}_{ux} \\ &= (I - \bar{A})(I - \bar{A} - \bar{B}\bar{M}\bar{C})^{-1} - \bar{B}\bar{M}\bar{C}(I - \bar{A} - \bar{B}\bar{M}\bar{C})^{-1} \\ &= (I - \bar{A} - \bar{B}\bar{M}\bar{C})(I - \bar{A} - \bar{B}\bar{M}\bar{C})^{-1} = I, \\ &(I - \bar{A})\boldsymbol{\Phi}_{xy} - \bar{B}\boldsymbol{\Phi}_{uy} \\ &= (I - \bar{A})(I - \bar{A} - \bar{B}\bar{M}\bar{C})^{-1}\bar{B}\bar{M} \\ &\quad - \bar{B}(I + \bar{M}\bar{C}(I - \bar{A} - \bar{B}\bar{M}\bar{C})^{-1}\bar{B})\bar{M}, \\ &= (I - \bar{A} - \bar{B}\bar{M}\bar{C})(I - \bar{A} - \bar{B}\bar{M}\bar{C})^{-1}\bar{B}\bar{M} - \bar{B}\bar{M} = 0. \end{aligned}$$

Similar steps can be used on (12) to obtain the right hand side from the left hand side.

For the second part of the lemma, if a system response problem is solved with $\{\boldsymbol{\Phi}_{xx}, \boldsymbol{\Phi}_{xy}, \boldsymbol{\Phi}_{ux}, \boldsymbol{\Phi}_{uy}\}$ while (11) and (12) hold, we obtain

$$\begin{aligned} &\boldsymbol{\Phi}_{uy} - \boldsymbol{\Phi}_{ux}\boldsymbol{\Phi}_{xx}^{-1}\boldsymbol{\Phi}_{xy}, \\ &= (\bar{M}\bar{C}(I - \bar{A} - \bar{B}\bar{M}\bar{C})^{-1}\bar{B} + I)\bar{M} \\ &\quad - \bar{M}\bar{C}(I - \bar{A} - \bar{B}\bar{M}\bar{C})^{-1}\bar{B}\bar{M} = \bar{M}. \end{aligned}$$

■

Since our output injection term in (3) must be causal, we need the following conditions to ensure causality in $\bar{M}$ obtained from the second part of Lemma 1:

Proposition 2 (Causality Condition for $\bar{\boldsymbol{u}}$)**.** *The output injection term in (7) is causal if both $\Phi_{xy,2}$ in $\boldsymbol{\Phi}_{xy}$ and $\Phi_{uy,2}$ in $\boldsymbol{\Phi}_{uy}$ are block lower triangular.*

Proof. Expanding $\mathbf{u}_e$ from (7) while also using (10):

$$\begin{aligned} \mathbf{u}_e &= \bar{M}\bar{\mathbf{y}} = (\boldsymbol{\Phi}_{uy} - \boldsymbol{\Phi}_{ux}\boldsymbol{\Phi}_{xx}^{-1}\boldsymbol{\Phi}_{xy}) \begin{bmatrix} \mathbb{1}_{pT} \\ \tilde{\mathbf{y}} \end{bmatrix}, \\ &= \Phi_{uy,1}\mathbb{1}_{pT} + \Phi_{uy,2}\tilde{\boldsymbol{y}} - \boldsymbol{\Phi}_{ux}\boldsymbol{\Phi}_{xx}^{-1} \begin{bmatrix} 0 \\ \Phi_{xy,1}\mathbb{1}_{pT} + \Phi_{xy,2}\tilde{\boldsymbol{y}} \end{bmatrix}. \end{aligned}$$

Since $\Phi_{xy,2}$ and $\Phi_{uy,2}$ are to be multiplied with $\tilde{\boldsymbol{y}}$, so to ensure causality, both $\Phi_{xy,2}$ and $\Phi_{uy,2}$ need to be block lower triangular. ■

To our best knowledge, system-level parameterization for affine output feedback has not been considered in the literature, so the above results are also a contribution to the finite-horizon optimal control literature on system-level synthesis.

Then, the recurrent recovery problem for a finite horizon T can be obtained via the following theorem:

Theorem 1 (Perfect Case Recurrent Recovery)**.** *For the case when there is no output data loss within a fixed time horizon T, i.e., when $q_k = 1\ \forall k \in [0, T]$, there exists an affine estimator that achieves path recurrent recovery over a finite horizon T with given initial and final recovery levels μ_0 and μ_T, respectively, if the following is feasible:*

$$\min_{\boldsymbol{\Phi}_{xx}, \boldsymbol{\Phi}_{xy}, \boldsymbol{\Phi}_{ux}, \boldsymbol{\Phi}_{uy}, \{\mu_k\}_{k=1}^{T-1}} J(\{\mu_k\}_{k=1}^{T-1})$$

subject to $\forall(\|\mathbf{w}\| \leq \eta_w, \|\mathbf{v}\| \leq \eta_v, \|\tilde{x}_0\| \leq \mu_0)$:

$$\|\tilde{x}_k\| \leq \mu_k,\ \forall k \in \{0, \dots, T\}, \tag{13a}$$

$$\tilde{\boldsymbol{x}} = [0 \;\; I]\,\bar{\mathbf{x}}, \tag{13b}$$

Proposition 2, Equations (8), (11), *and* (12) *hold,* (13c)

$\Phi_{xx}, \Phi_{xy}, \Phi_{ux}, \Phi_{uy}$ *satisfy the structure in* (10). (13d)

If the optimization problem is feasible, the output feedback estimator gains can be extracted via the following:

$$\begin{aligned} \bar{M} &= [\bar{\nu}\ \mathbf{M}] = \mathbf{\Phi}_{uy} - \mathbf{\Phi}_{ux}\mathbf{\Phi}_{xx}^{-1}\mathbf{\Phi}_{xy}, \\ \boldsymbol{\nu} &= \bar{\nu}\mathbb{1}_{pT}. \end{aligned} \tag{14}$$

Proof. Firstly, the constraints in (13a) come directly from the definition of path recurrent recovery in Definition 4. Furthermore, the estimation error $\tilde{\boldsymbol{x}}$ can be extracted from (8) as in (13b), while Equations (11) and (12) define the affine subspace (system-level parameterization) that parameterizes all system responses in (8). Moreover, we require that the recovery levels $\tilde{x}_k$ be satisfied for all possible realizations of the process and measurement noises that are bounded by η_w and η_v, respectively, as well all initial estimation errors bounded by the μ_0. Finally, since the aim is to optimize the estimation error bounds, the cost function is chosen as a function of the error bounds $\{\mu_k\}_{k=1}^{T-1}$ and the feedback gains can be obtained by item 2) of Lemma 1. ■

Note that if the initial and final recovery levels in the above problem are set to be the same, i.e., $\mu_0 = \mu_T$, then the path satisfies the *equalized recovery* property that was considered in [10], [11], [18], [20], [21]. Further, since the problem in (13) has semi-infinite (i.e., 'for all') constraints, we leverage duality in robust optimization [22] to obtain an equivalent optimization problem with a finite number of constraints.

Proposition 3. *The robust estimator design that achieves path recurrent recovery via Theorem 1 when there is no missing data is equivalent to the following problem:*

$$\min_{\mathbf{\Phi}_{xx},\mathbf{\Phi}_{xy},\mathbf{\Phi}_{ux},\mathbf{\Phi}_{uy},\{\mu_k\}_{k=1}^{T-1},\Pi_1,\Pi_2,\Pi_3} J(\{\mu_k\}_{k=1}^{T-1})$$

subject to: $\Pi_1 \geq 0, \Pi_2 \geq 0, \Pi_3 \geq 0, \mu \geq 0,$ (15a)

$$[\Pi_1\ \ \Pi_2\ \ \Pi_3]\,(I_3 \otimes \begin{bmatrix} I \\ -I \end{bmatrix}) = \begin{bmatrix} I \\ -I \end{bmatrix} \mathbf{G} \begin{bmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & \mu_0\mathbb{1}_n \end{bmatrix}, \tag{15b}$$

$$[\Pi_1\ \ \Pi_2\ \ \Pi_3] \begin{bmatrix} \eta_w\mathbb{1}_{2n_wT} \\ \eta_v\mathbb{1}_{2n_vT} \\ \mathbb{1}_{2n} \end{bmatrix} \leq \begin{bmatrix} \mu \\ \mu \end{bmatrix} - \begin{bmatrix} I \\ -I \end{bmatrix} \mathbf{h}, \tag{15c}$$

Proposition 2, Equations (11) *and* (12) *hold,* (15d)

$\Phi_{xx}, \Phi_{xy}, \Phi_{ux}, \Phi_{uy}$ *satisfy the structure in* (10). (15e)

where $\mathbf{G} \triangleq [0\ \ I] \left[\mathbf{\Phi}_{xx} \begin{bmatrix} 0 \\ 0 \\ I_T \otimes W \end{bmatrix}\ \ \mathbf{\Phi}_{xy}\bar{V}\ \ \mathbf{\Phi}_{xx} \begin{bmatrix} 0 \\ I_n \\ 0 \end{bmatrix} \right]$, $\mathbf{h} \triangleq [0\ \ I]\,\mathbf{\Phi}_{xx} \begin{bmatrix} \mathbb{1}_{pT} \\ 0 \\ 0 \end{bmatrix}$, *and* $\mu \triangleq [\mu_0\mathbb{1}_n^\top, \cdots, \mu_T\mathbb{1}_n^\top]^\top$.

Proof. It can be shown that the estimation error dynamics in (8) with its bound can be written as $-\mu \leq \tilde{\mathbf{x}} = \mathbf{G}\left[\mathbf{w}^\top\ \mathbf{v}^\top\ \tilde{x}_0^\top\right]^\top + \mathbf{h} \leq \mu$ or compactly as $\begin{bmatrix} I \\ -I \end{bmatrix} (\mathbf{G}\left[\mathbf{w}^\top\ \mathbf{v}^\top\ \tilde{x}_0^\top\right]^\top + \mathbf{h}) \leq \begin{bmatrix} \mu \\ \mu \end{bmatrix}$, while the uncertainty set can be written as $(I_3 \otimes \begin{bmatrix} I \\ -I \end{bmatrix}) \begin{bmatrix} \mathbf{w} \\ \mathbf{v} \\ \tilde{x}_0 \end{bmatrix} \leq \begin{bmatrix} \eta_w\mathbb{1}_{2n_wT} \\ \eta_v\mathbb{1}_{2n_vT} \\ \mu_0\mathbb{1}_{2n} \end{bmatrix}$. The robust counterpart in (15) can then be obtained using robust optimization techniques outlined in [22]. In addition, we post-multiplied the equality constraint of the robust counterpart with $\begin{bmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & \mu_0\mathbb{1} \end{bmatrix}$ and replace $(\tilde{\Pi}_3\mu_0)$ with another positive variable Π_3 to obtain (15b). ■

Note that the small change described at the end of the proof does not make any difference since we assumed that μ_0 is given. However, it has been found to be very beneficial if μ_0 is a decision variable, as described in [18, Proposition 1] since the bilineary will be moved to the sparser side of (15b). Further, the problem is bilinear (hence non-convex) but also sparse; thus off-the-shelf solvers can still easily solve it.

B. Missing Data Case with Data Loss Automaton

So far, in Section III-A, we kept our focus on a single path with no data loss at any instance in the path. To design a framework that takes the missing data model in the form of a data loss automaton $\mathcal{A}$, we will discuss the necessary modifications to the framework in Theorem 1 and by extension, Proposition 3.

Since the whole framework surrounds the design of the injection term $\boldsymbol{u}_e$ in (7) with $\bar{M}$ obtained from (14), then if there are multiple direct paths in $\mathcal{D}$ involved in the automaton $\mathcal{A}$, along with the ability to keep track of the location of the data loss signal via the nodes of $\mathcal{A}$, we just need to design the gain $\bar{M}$ for each path $d_a^b \in \mathcal{D}$ in $\mathcal{A}$, i.e., for each $d^\alpha = d_a^b \in \mathcal{D}$, there is a gain matrix associated with it as $\bar{M}^\alpha$, where:

$$\bar{M}^\alpha = \Phi_{uy}^\alpha - \Phi_{ux}^\alpha(\Phi_{xx}^\alpha)^{-1}\Phi_{xy}^\alpha. \tag{16}$$

Note that for notation efficiency, we will now use $\alpha \in \mathbb{N}_1^{|\mathcal{D}|}$ as the identifier for a direct path $d^\alpha = d_a^b \in \mathcal{D}$ for the remainder of this paper.

Apart from the multiple paths, another difference from the Section III-A is consideration of missing data. In that regard, let us first redefine the injection term in (7) to reflect that it is now dependent on path-specific gains:

$$\begin{aligned} \boldsymbol{u}_e = \bar{M}^\alpha\tilde{\boldsymbol{y}} = &\ \Phi_{uy,1}^\alpha\mathbb{1}_{pT} + \Phi_{uy,2}^\alpha\tilde{\boldsymbol{y}} \\ &- \mathbf{\Phi}_{ux}^\alpha(\mathbf{\Phi}_{xx}^\alpha)^{-1} \begin{bmatrix} 0 \\ \Phi_{xy,1}^\alpha\mathbb{1}_{pT} + \Phi_{xy,2}^\alpha\tilde{\boldsymbol{y}} \end{bmatrix}. \end{aligned} \tag{17}$$

Because any data loss in y_k affects the corresponding entry in $\tilde{\boldsymbol{y}}$, we will need to introduce some sort of constraints on $\Phi_{uy,2}^\alpha$ and $\Phi_{xy,2}^\alpha$ to cater to the missing data.

When defining the automaton in Definition 1, we mentioned that the event set $\mathcal{E}$ contains the event of *available* ('1') or *missing* ('0') corresponding to each transition in E. Then, when the data $\tilde{y}_k$ is missing at a particular time instance k on the node sequence of path d^α and the event associated event $e \in \mathcal{E}$ is '0', we can impose that constraint that the k-th block columns of $\Phi_{uy,2}^\alpha$ and $\Phi_{xy,2}^\alpha$ are set to zero, since these block columns multiplies the data $\tilde{y}_k$ and if they are set to zero, then $\mathbf{u}_e$ in (17) will be independent of the missing $\tilde{y}_k$.

Thus, for a direct path $d^\alpha \in \mathcal{D}$, since we have the transition information between each of the states in the sequence, we can also extract out the sequence of events, represented as $\mathcal{E}^\alpha$ that results in transition sequences of the direct path d^α. Then, $\mathcal{E}^\alpha$ contains the data loss signal associated with the $d^\alpha \in \mathcal{D}$ and can be used to set the corresponding block columns of $\Phi_{uy,2}^\alpha$ and $\Phi_{xy,2}^\alpha$ to zero when the data is missing.

The problem of designing an estimator that solves Problem 1 then becomes the problem of designing an estimator for each direct path d^α with its own corresponding data loss signal $\mathcal{E}^\alpha$ and thus, constraints on $\Phi^\alpha_{uy,2}$ and $\Phi^\alpha_{xy,2}$ that lead to constraints on its gain $\bar{M}^\alpha$. These constraints on $\Phi^\alpha_{uy,2}$ and $\Phi^\alpha_{xy,2}$ are imposed for simultaneously designing the estimator gains $\bar{M}^\alpha$ for all $d^\alpha \in \mathcal{D}$ to satisfy recurrent recovery (cf. Definition 5) via the theorem below. Then, once the gains $\bar{M}^\alpha = [\bar{\nu}^\alpha \; M^\alpha]$ (and hence, M^α and $\nu^\alpha = \bar{\nu}^\alpha \mathbb{1}_{pT}$) are designed offline (once only), the estimator in (17), i.e.,

$$u_{e,k} = \nu^\alpha_k + \sum_{i=0}^{k} M^\alpha_{(k,i)} \tilde{y}_k,$$

will be applied at real time by monitoring which direct path d^α the current missing data signal is traversing. This monitoring is possible by combining the knowledge of q_k for all $k \in \mathbb{N}$ and $\mathcal{A}$, as a consequence of Proposition 1.

Theorem 2 (Recurrent Recovery with Missing Data)**.** *Given a data loss automaton $\mathcal{A}$ with N nodes and its corresponding direct path set $\mathcal{D}$, an affine output feedback estimator that achieve recurrent recovery (cf. Definition 5) can be synthesized using the following optimization problem:*

$$\min_{\mathbf{\Phi}^\alpha_{xx}, \mathbf{\Phi}^\alpha_{xy}, \mathbf{\Phi}^\alpha_{ux}, \mathbf{\Phi}^\alpha_{uy}, \{\mu_s\}_{s=1}^N} J(\{\{\mu_s\}_{s=1}^N)$$

subject to

$$\forall d^\alpha = d^{b(\alpha)}_{a(\alpha)} = \{s^\alpha(0) = s_{a(\alpha)}, \ldots, s^\alpha(T^{b(\alpha)}_{a(\alpha)}) = s_{b(\alpha)}\} \in \mathcal{D}:$$

$$\forall (\|\mathbf{w}^\alpha\| \le \eta_w, \|\mathbf{v}^\alpha\| \le \eta_v, \|\tilde{x}^\alpha_0\| \le \mu_{s^\alpha(0)}):$$

$$\|\tilde{x}^\alpha_k\| \le \mu_{s^\alpha(k)}, \; \forall k \in \{0, \ldots, T^{b(\alpha)}_{a(\alpha)}\}, \tag{18a}$$

$$\tilde{\boldsymbol{x}}^\alpha = [0 \;\; I]\, \bar{\mathbf{x}}^\alpha, \tag{18b}$$

Proposition 2, Equations (8), (11), *and* (12) *hold for each*

$$\mathbf{\Phi}^\alpha_{xx}, \mathbf{\Phi}^\alpha_{xy}, \mathbf{\Phi}^\alpha_{ux}, \mathbf{\Phi}^\alpha_{uy}, \tag{18c}$$

that satisfy the structure in (10)*, where the k-th block columns of $\Phi^\alpha_{uy,2}$ and $\Phi^\alpha_{xy,2}$, i.e., $\Phi^\alpha_{uy,2}(:, (k-1)p+1 : kp) = 0$ and $\Phi^\alpha_{xy,2}(:, (k-1)p+1 : kp) = 0$ (using MATLAB notation for the elements of the matrices), are set to zero if $e^\alpha(k) =$'0' from the event sequence $\{e^\alpha(0), \ldots, e^\alpha(T^{b(\alpha)}_{a(\alpha)})\}$ corresponding to the direct path d^α.*

If the optimization problem is feasible, the output feedback estimator gains can be extracted via the following:

$$\begin{aligned} \bar{M}^\alpha &= [\bar{\nu}^\alpha \;\; \mathbf{M}^\alpha] = \mathbf{\Phi}^\alpha_{uy} - \mathbf{\Phi}^\alpha_{ux} (\mathbf{\Phi}^\alpha_{xx})^{-1} \mathbf{\Phi}^\alpha_{xy}, \\ \boldsymbol{\nu}^\alpha &= \bar{\nu}^\alpha \mathbb{1}_{pT}. \end{aligned} \tag{19}$$

Proof. First, we assign a recovery level representing the maximum estimation error bound to each node in the automaton $\mathcal{A}$. Then, by Proposition 1, all traces/paths of the automaton will lie on some direct path at each time instant. Hence, by enforcing that for each direct path, the estimation errors must satisfy the recovery levels for the nodes that are being traversed using the constraints in (13), we can ensure that the estimator satisfies recurrent recovery as defined in Definition 5. ∎

As with Theorem 1, the above contains *for all* constraints; thus, we derive the robustified equivalent of Theorem 2:

Proposition 4. *The robust estimator design that achieves recurrent recovery via Theorem 2 with data loss automaton $\mathcal{A}$ with N nodes and its corresponding direct path set $\mathcal{D}$ is equivalent to the following optimization problem:*

$$\min_{\mathbf{\Phi}^\alpha_{xx}, \mathbf{\Phi}^\alpha_{xy}, \mathbf{\Phi}^\alpha_{ux}, \mathbf{\Phi}^\alpha_{uy}, \{\mu_s\}_{s=1}^N, \Pi^\alpha_1, \Pi^\alpha_2, \Pi^\alpha_3} J(\{\{\mu_s\}_{s=1}^N)$$

subject to

$$\forall d^\alpha = d^{b(\alpha)}_{a(\alpha)} = \{s^\alpha(0) = s_{a(\alpha)}, \ldots, s^\alpha(T^{b(\alpha)}_{a(\alpha)}) = s_{b(\alpha)}\} \in \mathcal{D}:$$

$$\Pi^\alpha_1 \ge 0, \Pi^\alpha_2 \ge 0, \Pi^\alpha_3 \ge 0, \mu \ge 0, \tag{20a}$$

$$[\Pi^\alpha_1 \;\; \Pi^\alpha_2 \;\; \Pi^\alpha_3]\,(I_3 \otimes \begin{bmatrix} I \\ -I \end{bmatrix}) = \begin{bmatrix} I \\ -I \end{bmatrix} \mathbf{G}^\alpha \begin{bmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & \mu_{s^\alpha(0)} \mathbb{1}_n \end{bmatrix}, \tag{20b}$$

$$[\Pi^\alpha_1 \;\; \Pi^\alpha_2 \;\; \Pi^\alpha_3] \begin{bmatrix} \eta_w \mathbb{1}_{2n_w T} \\ \eta_v \mathbb{1}_{2n_v T} \\ \mathbb{1}_{2n} \end{bmatrix} \le \begin{bmatrix} \mu^\alpha \\ \mu^\alpha \end{bmatrix} - \begin{bmatrix} I \\ -I \end{bmatrix} \mathbf{h}^\alpha, \tag{20c}$$

Proposition 2, Equations (11) *and* (12) *hold for each*

$$\Phi^\alpha_{xx}, \Phi^\alpha_{xy}, \Phi^\alpha_{ux}, \Phi^\alpha_{uy}, \tag{20d}$$

that satisfy the structure in (10)*, where the k-th block columns of $\Phi^\alpha_{uy,2}$ and $\Phi^\alpha_{xy,2}$, i.e., $\Phi^\alpha_{uy,2}(:, (k-1)p+1 : kp) = 0$ and $\Phi^\alpha_{xy,2}(:, (k-1)p+1 : kp) = 0$ (using MATLAB notation for the elements of the matrices), are set to zero if $e^\alpha(k) =$'0' from the event sequence $\{e^\alpha(0), \ldots, e^\alpha(T^{b(\alpha)}_{a(\alpha)})\}$ corresponding to the direct path d^α, $\mathbf{G}^\alpha \triangleq [0 \;\; I] \left[\mathbf{\Phi}^\alpha_{xx} \begin{bmatrix} 0 \\ 0 \\ I_T \otimes W \end{bmatrix} \; \mathbf{\Phi}^\alpha_{xy} \bar{V} \;\; \mathbf{\Phi}^\alpha_{xx} \begin{bmatrix} 0 \\ I_n \\ 0 \end{bmatrix} \right]$, $\mathbf{h}^\alpha \triangleq [0 \;\; I]\, \mathbf{\Phi}^\alpha_{xx} \left[\mathbb{1}_{pT}{}^\top \;\; 0 \;\; 0 \right]^\top$ and $\mu^\alpha \triangleq [\mu_{s^\alpha(0)} \mathbb{1}^\top_n, \cdots, \mu_{s^\alpha(T^{b(\alpha)}_{a(\alpha)})} \mathbb{1}^\top_n]^\top$.*

Proof. This robustified equivalent is obtained using duality in robust optimization [22] in the same way as Proposition 3 for each direct path of the data loss automaton. ∎

Note that unlike the problem in Proposition 3 with given μ_0, in the above optimization problem, $\mu_{s^\alpha(0)}$ is a decision variable that leads to bilinearity since G^α is also dependent on decision variables $\mathbf{\Phi}^\alpha_{xx}$ and $\mathbf{\Phi}^\alpha_{xy}$. This is the main reason why we multiplied on both sides of the original equality constraint to obtain (20b), as was described in the proof of Proposition 3, since this was found in [18, Proposition 1] to move the bilineary to the sparser side of (20b), resulting in the optimization problem being more easily solved using off-the-shelf solvers.

IV. Illustrative Example

In this section, we demonstrate the effectiveness of our proposed system-level recurrent state estimator design on a simulation example. All simulations are performed in MATLAB®2020b using YALMIP [23] and IPOPT [24] as a solver, on a Windows 10 laptop with hexa-core Intel Core i9 processor and 32GB RAM.

A. Batch Reactor Process

We consider the linearized model of batch reactor process from [25], that was time-discretized with a sampling time of T_s = 0.05 seconds using MATLAB `c2d` command using the zero-order hold option. The resulting state space model is:

$$A = \begin{bmatrix} 1.0795 & -0.0045 & 0.2896 & -0.2367 \\ -0.0272 & 0.8101 & -0.0032 & 0.0323 \\ 0.0447 & 0.1886 & 0.7317 & 0.2354 \\ 0.0010 & 0.1888 & 0.0545 & 0.9115 \end{bmatrix}, B = \begin{bmatrix} 0.0006 & -0.0239 \\ 0.2567 & 0.0002 \\ 0.0837 & -0.1346 \\ 0.0837 & -0.0046 \end{bmatrix},$$

$$C = \begin{bmatrix} 1 & 0 & 1 & -1 \\ 0 & 1 & 0 & 0 \end{bmatrix}, V = I_p, W = \emptyset.$$

The measurement noise is bounded by the value of η_v = 0.05, while the missing-data specification considered for this example is $(1, 3)$-firmness, meaning that in any 3-step window, there can only be at most one missing data.

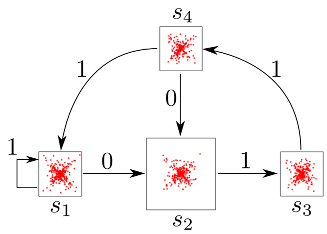

Fig. 2: Estimation error spread at each node of the (1,3)-firm automaton in Figure 2, where each box represents the area $(2\mu_{s(i)})^2$ corresponding to each node s_i, with the center of the box implying $\tilde{x}_k = 0$ at that node and the deviation from the center equal to the estimation error $\tilde{x}_k$ for some k.

B. Data Loss Automaton and Simulation Results

It can be shown that the missing-data specification of (1,3)-firmness can be modeled by the automaton in Figure 1. Associating estimation error bounds (i.e., recovery levels) $\{\mu_{s(1)}, \mu_{s(2)}, \mu_{s(3)}, \mu_{s(4)}\}$ to the nodes $\{s_1, s_2, s_3, s_4\}$ respectively, along with the cost function $J(\cdot) = \mu_{s(1)} + \mu_{s(2)} + \mu_{s(3)} + \mu_{s(4)}$, we use the robust design in Proposition 4 to design the affine estimator gains (3) for the estimator given by (2). The error bounds corresponding to each node after solving the design problem are $\{\mu_{s(1)}, \mu_{s(2)}, \mu_{s(3)}, \mu_{s(4)}\} = \{0.3556, 0.5726, 0.3541, 0.3519\}$.

We then performed 50 simulation runs with different randomly generated noise vectors $\boldsymbol{v}$ over a duration of 20 steps in each run, considering the node s_1 as the starting node with the initial estimation error $\tilde{x}_0$ randomly generated from within $\|\tilde{x}_0\| \leq \mu_{s(1)}$. The data-loss signal considered in all of the 50 runs is:

$$\{q_k\}_{k=0}^{T-1} = \{1, 0, 1, 1, 1, 0, 1, 1, 0, 1, 1, 0, 1, 1, 1, 1, 1, 1, 0, 1\}.$$

The resulting estimation error spread at each node of the automaton is shown in Figure 2. As expected, the estimation errors $\tilde{x}_k$ for all k in all runs are within the boxes, whose sizes represent the maximum estimation errors $\mu_{s(i)}$ at each node s_i and the scatter points represent actual estimation errors $\tilde{x}$ at those nodes with the centers of the boxes as the origins (i.e., $\tilde{x} = 0$). Hence, the estimation errors with our proposed estimator are guaranteed to be recurrent despite missing data and always remain below $\max_s \mu_s = 0.5726$.

Moreover, we implemented two slight variations of the proposed estimator design where we replaced the system-level synthesis based estimator gains with the estimator gains based on output feedback parameterization [10] and output error feedback parameterization [11] (details omitted for brevity). Our simulation results show that we obtain the exact same estimation errors bounds/recovery levels at each node as in Figure 2, which shows, at least for this example, that the proposed system-level synthesis approach for satisfying recurrent recovery is competitive with output feedback and output error feedback parameterizations.

V. Conclusions

This paper introduced a system-level robust state estimator for uncertain/bounded-error affine systems subject to data losses modeled by automata. Our robust estimator design guarantees that the state estimation errors robustly remain bounded in a recurrent manner despite worst-case realizations of noise and sensor uncertainties by enforcing a recurrent recovery property, where the estimation errors are required to be recurrent to some minimum recovery levels at each node of the automata. Our design was built upon system-level parameterization for which we also proposed a novel extension of system-level synthesis methods to the case with affine output feedback. Our future work will include the extension of this approach to obtain robustness to delayed data that is also common in networked control systems as well as the consideration of nonlinear system dynamics.

References

[1] B. Sinopoli, L. Schenato, M. Franceschetti, K. Poolla, M. I. Jordan, and S. S. Sastry, "Kalman filtering with intermittent observations," *IEEE TAC*, vol. 49, no. 9, pp. 1453–1464, 2004.
[2] S. C. Smith and P. Seiler, "Estimation with lossy measurements: jump estimators for jump systems," *IEEE Transactions on Automatic Control*, vol. 48, no. 12, pp. 2163–2171, 2003.
[3] G. Battistelli, L. Benavoli, and L. Chisci, "State estimation with remote sensors and intermittent transmissions," *Systems & Control Letter*, vol. 61, no. 1, pp. 155–164.
[4] I. Matei and J. S. Baras, "Optimal state estimation for discrete-time Markovian jump linear systems, in the presence of delayed output observations," *IEEE TAC*, vol. 56, no. 9, pp. 2235–2240, 2011.
[5] X. Wang, W. Qin, Y. Bai, and N. Cui, "Cooperative target localization using multiple uavs with out-of-sequence measurements," *Aircraft Engineering and Aerospace Techn.*, vol. 89, no. 1, pp. 112–119, 2017.
[6] S. Amin, A. A. Cárdenas, and S. S. Sastry, "Safe and secure networked control systems under denial-of-service attacks," in *International Workshop on Hybrid Systems: Computation and Control*. Springer, 2009, pp. 31–45.
[7] J. Shamma and K. Tu, "Set-valued observers and optimal disturbance rejection," *IEEE TAC*, vol. 44, no. 2, pp. 253–264, 1999.
[8] M. Milanese and A. Vicino, "Optimal estimation theory for dynamic systems with set membership uncertainty: An overview," *Automatica*, vol. 27, no. 6, pp. 997–1009, 1991.
[9] J. Chen and C. Lagoa, "Observer design for a class of switched systems," in *IEEE CDC/ECC*, 2005, pp. 2945–2950.
[10] K. Rutledge, S. Z. Yong, and N. Ozay, "Finite horizon constrained control and bounded-error estimation in the presence of missing data," *Nonlinear Analysis: Hybrid Systems*, vol. 36, p. 100854, 2020.
[11] S. M. Hassaan, Q. Shen, and S. Z. Yong, "Path-dependent controller and estimator synthesis with robustness to delayed and missing data," in *International Conference on Hybrid Systems: Computation and Control*, 2021, pp. 1–11.
[12] F. Felicioni, N. Jia, Y.-Q. Song, and F. Simonot-Lion, "Impact of a (m,k)-firm data dropouts policy on the quality of control," in *IEEE Intl. Workshop on Factory Communication Systems*, 2006, pp. 353–359.
[13] R. M. Jungers, A. Kundu, and W. Heemels, "Observability and controllability analysis of linear systems subject to data losses," *IEEE Trans. on Automatic Control*, vol. 63, no. 10, pp. 3361–3376, 2017.
[14] J. Skaf and S. P. Boyd, "Design of affine controllers via convex optimization," *IEEE Transactions on Automatic Control*, vol. 55, no. 11, pp. 2476–2487, 2010.
[15] P. J. Goulart and E. C. Kerrigan, "Output feedback receding horizon control of constrained systems," *International Journal of Control*, vol. 80, no. 1, pp. 8–20, 2007.
[16] J. C. Doyle, N. Matni, Y.-S. Wang, J. Anderson, and S. Low, "System level synthesis: A tutorial," in *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*. IEEE, 2017, pp. 2856–2867.
[17] J. Sieber, S. Bennani, and M. N. Zeilinger, "A system level approach to tube-based model predictive control," *IEEE Control Systems Letters*, vol. 6, pp. 776–781, 2021.
[18] S. M. Hassaan, Q. Shen, and S. Z. Yong, "Equalized recovery state estimators for linear systems with delayed and missing observations," *IEEE Control Systems Letters*, vol. 6, pp. 85–90, 2021.
[19] F. Blanchini and M. Sznaier, "A convex optimization approach to synthesizing bounded complexity ℓ^∞ filters," *IEEE Transactions on Automatic Control*, vol. 57, no. 1, pp. 216–221, 2012.
[20] K. Rutledge, S. Z. Yong, and N. Ozay, "Optimization-based design of bounded-error estimators robust to missing data," *IFAC-PapersOnLine*, vol. 51, no. 16, pp. 157–162, 2018.
[21] S. M. Hassaan, Q. Shen, and S. Z. Yong, "Bounded-error estimator design with missing data patterns via state augmentation," in *American Control Conference (ACC)*, 2019, pp. 447–452.
[22] A. Ben-Tal, L. El Ghaoui, and A. Nemirovski, *Robust optimization*. Princeton University Press, 2009.
[23] J. Löfberg, "Yalmip : A toolbox for modeling and optimization in matlab," in *Proceedings of the CACSD Conference*, Taiwan, 2004.
[24] A. Wächter and L. Biegler, "On the implementation of a primal-dual interior point filter line search algorithm for large-scale nonlinear programming," *Math. Programming*, vol. 106, no. 1, pp. 25–57, 2006.
[25] Y. Shoukry, J. Araujo, P. Tabuada, M. Srivastava, and K. H. Johansson, "Minimax control for cyber-physical systems under network packet scheduling attacks," in *ACM HiCoNS*, 2013, pp. 93–100.