UNKNOWN INPUT OBSERVERS FOR CYBER-PHYSICAL SYSTEMS SUBJECTED TO MALICIOUS ATTACKS

by

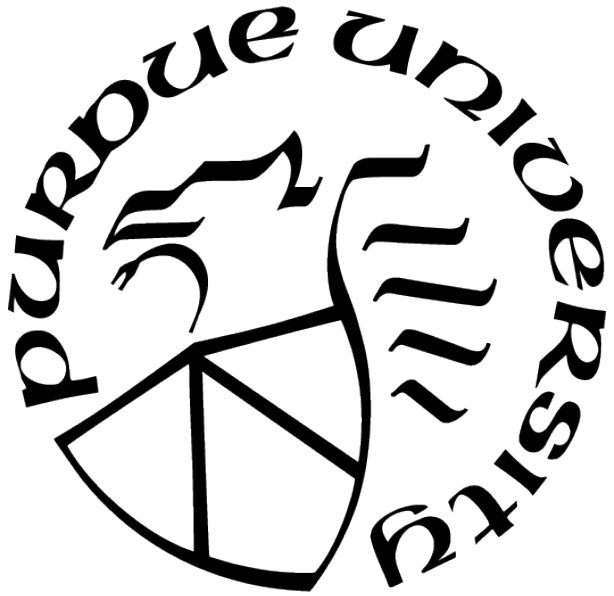
Mukai Zhang

A Dissertation

Submitted to the Faculty of Purdue University

In Partial Fulfillment of the Requirements for the degree of

Doctor of Philosophy



School of Electrical and Computer Engineering

West Lafayette, Indiana

December 2021

THE PURDUE UNIVERSITY GRADUATE SCHOOL
STATEMENT OF COMMITTEE APPROVAL

Dr. Stanislaw H. Żak, Chair

School of Electrical and Computer Engineering

Dr. Jianghai Hu

School of Electrical and Computer Engineering

Dr. Shreyas Sundaram

School of Electrical and Computer Engineering

Dr. Mark R. Bell

School of Electrical and Computer Engineering

Approved by:

Dimitrios Peroulis

For Father, who guides me the right path,

For Mother, who is an example of me,

For Grandma and Grandpa, who brought me happiness,

For Xinyu, who loves me so much.

ACKNOWLEDGMENTS

''I am a slow walker, but I never walk backwards." Abraham Lincoln.

Many people have offered me valuable help on this journey so far, including my advisor, my classmates, my parents, and my girlfriend.

First, I would like to give my sincere gratitude to Prof. Stanislaw. H. Żak, my advisor who, with extraordinary patience and consistent encouragement, gave me great help by providing me with necessary materials, advice of great value and inspiration of new ideas. It is his suggestions that draw my attention to a number of deficiencies and make many things clearer. Without his strong support, this dissertation could not been the present form. My heartfelt thanks also go to Prof. Kennell, my teacher of thesis writing course, for his help in the making of this thesis as well as his enlightening lectures from which I have benefited a great deal.

I am pleased to acknowledge Professor Jianghai Hu for an amazing linear/hybrid systems class and for always being patient to answer questions. It is he who guided me to make my first steps into the Automatic Control area. Also, a hearty thank you to Professor Mark R. Bell for giving knowledge of random variables. A mega-super-thanks also goes to Prof. Shreyas Sundaram, who approved my qualifying exam remediation proposal.

I am especially grateful to Professor Stefen Hui for his discussions and insightful comments that contributed significantly to this research.

Also, I please to acknowledge Badriah Alenezi for her invaluable assistance throughout the preparation of the original manuscript. She graciously made numerous comments and sound suggestions to the outline of this document.

I am extremely grateful for my roommate, Zizhuang Wu, for his cooperation, comments, and wise advice, although his advice is not always taken, where faults and infelicities remain.

Lastly, my thanks would go to my beloved family, my father, Yunchun Zhang, my mother, Juan Chen, and love of my life, Xinyu Hu, for their loving considerations and great confidence in me throughout these years. I also owe my sincere gratitude to my friends and my fellow classmates who gave me their help and time in listening to me and helping me work out my problems during the difficult courses.

To everyone else, I apologize if I have not mentioned you by name. If you do read this document and find yourself disappointed, please free to call me so that we can have a nice conversation in Harry's "Chocolate" Shop.

— *Mukai Zhang*

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABSTRACT

Cyber-Physical Systems (CPSs) consist of physical and computational components usually interconnected through the internet. This type of systems have found applications in robotic surgery, smart medical services, driverless cars, smart power grids as well as in modern homes and offices. For a CPS to function properly, a reliable and secure communications between the system physical and cyber elements is of utmost importance. Malicious attacks during control signals and output measurements transmission between the physical plant and the control center must be addressed, which is the main research problem studied in this thesis.

A novel robust observer was proposed to synthesize a combined controller-observer compensator for a class of CPSs with sparse malicious attacks and arbitrary disturbances. The compensator consists of a controller, a norm approximator, and an unknown input observer (UIO). The proposed observer was compared with a norm-based observer given in the literature to show its advantage. To further enhance the proposed observer's performance against arbitrary disturbances, design methods were given that use fictitious output measurements and error correcting code (ECC) approach. The design of the UIO was extended to a bank of UIOs in order to improve the observer's performance against sparse malicious attacks.

The proposed observer can be used in the design of UIO-based fault detection and isolation (FDI) algorithms as well as in the distributed fault-tolerant control of large-scale interconnected systems. The results of this thesis can be applied to the design of controller-observer compensators for CPSs with modeling uncertainties.

1. INTRODUCTION

1.1 Motivation and Literature Overview

The term "cyber-physical system" (CPS) was first proposed in 2006 by Helen Gill of the US National Science Foundation [1]. A CPS consists of two or more processing and physical subsystems linked by communication networks. The simplest CPS consists of a processing/controlling subsystem and a physical subsystem of sensors, actuators and a physical plant. Sensors at the physical plant collect measurements that are sent to the processor/controller and control signals from the controller are sent to the actuators. Typical examples of CPSs include Internet of Things (IoT) [2], industrial internet [3], smart grids [4], and self-driving vehicles [5].

Since the CPSs are interconnected by communication networks that are not necessarily secure, the issue of reliable data communication must be addressed in the design of CPSs. In particular, sparse malicious attacks in the communication networks need to be addressed. Overcoming sparse malicious attacks issue in CPSs has been researched by many groups around the world. For example, the authors of [6] present constraint optimization methods for evaluating the impacts of sparse undetectable sensor attacks against CPSs. In [7], secure Luenberger-like observers are proposed for CPSs under sparse actuator and sensor attacks. In [8], methods were reported for solving security issues in remote state estimation of CPS and sensor measurements being corrupted by external sparse malicious packet drop attacks in communication networks. Other methods of overcoming sparse malicious packet drops were reported in [9]–[12].

A useful tool to solve the problem of sparse malicious attacks in CPSs is the sparse vector recovery method [13]–[17]. This is because the analysis of vulnerabilities due to the unknown disturbance of the communication network such as noise, delay and packet drops can be formulated as a sparse vector recovery problem. The sparse vector recovery problem can be formulated as estimating an unknown sparse vector e in the linear system, $b = Ax + \mathrm{e}$, when the vector b and the matrix A are known. Here by sparse, we mean there are more zero entries than non-zero entries in the vector e. It is reported in [18] that this problem can be transformed into a 0-norm minimization problem with equality constraints and solved

using norm approximation, that is, using 1-norm to approximate the 0-norm solution of the minimization problem. We know that certain algorithms for solving minimum norm problems can be applied to solve sparse vector recovery problems. Two such algorthims are given in [19], [20]. These algorithms are based on the projected steepest descent method for constrained optimization problems for obtaining minimum 1-norm solutions. In [20], an algorithm using a penalty function approach and the gradient method to solve minimum norm problems is given. In [21], a linear programming is used to solve such problems. One disadvantage of using the norm approximation method for sparse malicious attacks recovery in CPS is that it creates one sampling period time delay [22], [23]. Therefore, a robust control strategy needs to take this issue into account. A notion of robustness for cyber systems inspired by existing notions of input-output stability is introduced in [24]. Different robust control strategies have been proposed in the literature. For example, authors in [25] propose a fault-tolerant controller to compensate for actuator faults in CPS.

Since the state or a good estimate of it must be available for the efficient control the CPS subjected to disturbances, a state estimator is an essential component of a CPS. A common way to estimate the state of a system subjected to disturbances is to use an unknown input observer (UIO); see for example [26]–[28]. The problem of designing observers for a linear system with both known and unknown inputs has been studied since at least 1969 [29]. See [30] for an overview of early UIO developments and [31] for a comparative study of some UIO architectures. Our motivation incorporating the UIO approach to the control and state estimation of CPSs is that in practice sparse communication errors and arbitrary disturbances can be modeled as unknown inputs.

In computing as well as in telecommunication, error correcting code (ECC) approach is used to control errors in data transmitted through unreliable or noisy communication channels. The main idea is to add redundant information to the message sent. The redundancy enables the receiver to detect errors that may occur in the transmitted message [32, p. 355]. To illustrate this approach, suppose we send the same signal three or more times. Let $\mathcal{P}$ be

the probability that the sparse error is not zero. Then the probability having two or more errors, using the binomial formula, when three copies are sent is

$$\text{Probability } (\#\text{ error} \geq 2) = 3(1-\mathcal{P})\mathcal{P}^2 + \mathcal{P}^3.$$

When $\mathcal{P} = 0.05$, this error is 0.0073. When there are fewer than two errors, then at least two copies have the correct value. So the probability of correct decoding is 99.27%. In general, if n copies are sent, then we can be sure of correct decoding if we have no more than $n-2$ errors because we would have two identical copies. We assume a continuous probability density and so the probability of having two errors leading to the same value is zero. The probability of having $n-1$ or n errors is

$$\text{Probability } (\#\text{ error} \geq n-1) = n(1-\mathcal{P})\mathcal{P}^{n-1} + \mathcal{P}^n.$$

When $\mathcal{P} = 0.05$ and $n = 4$, this probability is 4.8125×10^{-4} and the probability of correct decoding is 99.95%. When $n = 5$, the probability of correct decoding is 99.997%. Another advantage of sending multiple copies is that even when the error is not sparse, the copies can be averaged to reduce the effective noise variance by a factor of n when n copies are sent. However, the above approach may not work when the communication errors are injected maliciously. For example, suppose we have a system under malicious attacks. Let $\mathrm{e}_m[k]$ be the maliciously injected communication error to the transmitted signal $y[k]$ at sample k. Then no matter how many copies of the output signal $y[k]$ have been transmitted, all these copies will be corrupted by the same injected error $\mathrm{e}_m[k]$ during the signal transmission at sample k. There is then no difference between the signals transmitted and therefore correct signal cannot be recovered.

1.2 Organization of the Thesis

A novel robust observer is proposed to synthesize a combined controller-observer compensator for a class of CPSs with sparse malicious attacks and arbitrary disturbances. The compensator consists of a controller, a norm approximator, and an unknown input observer

(UIO). The proposed observer is compared with a norm-based observer given in the literature to show its advantage. To further enhance the presented observer's performance against arbitrary disturbances, novel design methods are proposed that use fictitious output measurements and error correcting code (ECC) approach. The design of the UIO is extended to a bank of UIOs in order to improve the observer's performance against sparse malicious attacks. The thesis is organized as follows.

In Chapter 2, a method for recovering an unknown sparse error e in the overdetermined system $b = Ax + \mathrm{e}$ is proposed. The Q-R decomposition [33] is first used to find a left annihilator $Q_2^\top$ of the matrix A. The original overdetermined system is then transformed into an underdetermined system of the form $Q_2^\top b = Q_2^\top \mathrm{e}$. Next, a Gaussian random matrix G is premultiplied to both sides of $Q_2^\top b = Q_2^\top \mathrm{e}$ to obtain $z = F\mathrm{e}$, where $z = GQ_2^\top b$ and $F = GQ_2^\top$. The sparse error vector e is then recovered by solving a convex optimization problem $\min \|\mathrm{e}\|_1$, subject to $z = F\mathrm{e}$. Two methods for solving such optimization problems are presented and compared. The proposed 1-norm regularization method is then applied to the secure state estimation of a CPS. A simple case with the output measurements corrupted by sparse malicious packet drops and a secure control signal transmission of the CPS, is considered in this chapter.

In Chapter 3, a novel discrete-time (DT) observer architecture is proposed for DT CPSs corrupted by unknown sparse errors between the controller and the sensors and between the actuators and the controller. The sparse error between the controller and the sensors is recovered using the method described in Chapter 2. The linear programming optimization is used to solve the 1-norm optimization problem. The sparse error between the actuators and the controller can be estimated using two proposed unknown input estimators. A combined approximator and the UIO architecture forms the observer to estimate the state of the plant corrupted by unknown sparse errors simultaneously at the plant's inputs and outputs. The observer design is formulated in terms of linear matrix inequalities (LMIs).

In Chapter 4, the proposed DT observer is used to synthesize a combined controller-observer compensator for continuous-time (CT) network systems. Necessary and sufficient conditions for the existence of the proposed UIO are given. Linearized and discretized plant parameters are used in the compensator design. The advantage of the compensator synthesis

in the DT domain over the CT domain is that in many cases the condition for the existence of an UIO fails for a CT plant while it holds for a discretized plant. A class of systems for which the existence condition for the UIO fails in the CT domain while holds in the DT domain is given.

In Chapter 5, two types of estimators for CPSs with unsecured communication channels and subject to disturbances are presented and their performance compared. The communication errors are assumed to be sparse while the disturbances are arbitrary. The first proposed estimator uses the 1-norm approximation of the 0-norm minimization problem which is the basis of this estimator. The second estimator combines the norm-based estimator with the UIO architecture. It is demonstrated through analytic considerations and simulations that the combined norm-UIO based estimator is superior to the norm-based estimator. To further enhance the presented estimators' performance, a novel design method is proposed that uses fictitious output measurements.

In Chapter 6, a controller-observer compensator is proposed for CPSs with sensor and actuator faults and unsecured communication networks. The observer combines a norm approximator for sparse malicious attacks recovery with a bank of UIOs to estimate the CPS state. Convergence analysis of the state estimation error of the proposed UIO architecture is given. To enhance the proposed observer's performance, a sensor and actuator fault filters are proposed that use error correcting code (ECC) approach. A model reference controller with a performance level that can be calculated is given. The controller-observer compensator is applied to a self-driving ground vehicle to show its effectiveness.

In Chapter 7, the results obtained so far are summarized and open problems for further research are presented.

1.3 Publications

- M. Zhang, S. Hui, M.R. Bell, S.H. Żak, *Vector Recovery for a Linear System Corrupted by Unknown Sparse Error Vectors with Applications to Secure State Estimation*, IEEE Control Systems Letters (L-CSS), vol. 3, pp. 895–900, October 2019.

- M. Zhang, B. Alenezi, S. Hui, S.H. Żak, *State Estimation of Networked Control Systems Corrupted by Unknown Input and Output Sparse Errors*, in 2020 American Control Conference (ACC), Denver, CO, USA, 2020, pp. 4393–4398.

- M. Zhang, B. Alenezi, S. Hui, S.H. Żak, *Unknown Input Observers for Discretized Systems with Application to Networked Systems Corrupted by Sparse Malicious Packet Drops*, IEEE Control Systems Letters (L-CSS), vol. 5, no. 4, pp. 1261–1266.

- M. Zhang, B. Alenezi, S. Hui, S.H. Żak, *Estimators for Cyber-Physical Systems With Unsecured Communication Networks and Subjected to Disturbances.* Regular paper submitted to IEEE Transactions on Cybernetics.

- M. Zhang, B. Alenezi, S. Hui, S.H. Żak, *Controller-Observer Compensator Synthesis for Unsecured Cyber-Physical Systems.* Regular paper submitted to IEEE Transactions on Automatic Control.

- B. Alenezi, M. Zhang, S. Hui, S.H. Żak, *State Observers and Unknown Input Estimators for Discrete-Time Nonlinear Systems Characterized by Incremental Multiplier Matrices*, in 59^{th} Conference on Decision and Control (CDC), Jeju Island, Republic of Korea, 2020, pp. 5409–5414.

- B. Alenezi, M. Zhang, S.H. Żak, *Observer-Based Controller Synthesis for Decentralized Networked Systems*, in 2020 IEEE Conference on Control Technology and Applications (CCTA), Montreal, QC, Canada, 2020, pp. 732–737.

- B. Alenezi, M. Zhang, S. Hui, S.H. Żak, *Simultaneous Estimation of the State, Unknown Input, and Output Disturbance in Discrete-Time Linear Systems*, IEEE Transactions on Automatic Control, in press.

- B. Alenezi, M. Zhang, S. Hui, S.H. Żak, *Delayed Estimation of Unknown Input and Output Disturbances in Discrete-Time Linear Systems.* Submitted to IEEE Control Systems Letters (L-CSS).

2. VECTOR RECOVERY FOR A LINEAR SYSTEM CORRUPTED BY UNKNOWN SPARSE ERRORS WITH APPLICATIONS TO SECURE STATE ESTIMATION

2.1 Problem Statement

Let $m > n$ and let $A \in \mathbb{R}^{m\times n}$ have full column rank. We consider the case where the measurement vector Ax is corrupted by an unknown error e. This is modeled as $b = Ax + \mathrm{e}$. The problem we consider is whether it is possible to recover x exactly from the given data A and the measurement b. In our discussion, we use the following definition.

Definition 1 (0-norm)**.** [34] The 0-norm of a finite dimensional vector x, denoted $\|x\|_0$, is the number of nonzero entries in x.

Candes and Tao [21] observe that in order to be able to recover a solution vector x to $b = Ax + \mathrm{e}$, it is necessary to assume that only a small number of entries of b has been corrupted. They further observe that since we have full knowledge of the vector b and the full rank matrix A, then to reconstruct x, it is sufficient to reconstruct e. In [21], they propose an approach that allows one to reconstruct the error e. Their idea is to first find a matrix $F \in \mathbb{R}^{(m-n)\times m}$ such that $FA = 0$, then premultiply both sides of $b = Ax + \mathrm{e}$ by F to obtain, $Fb = FAx + F\mathrm{e}$. Let $z = Fb$. Then since $FAx = 0$, we obtain $F\mathrm{e} = z$. Thus the original problem has been reduced to reconstructing the sparse error vector e. In our analysis, we use standard vector p-norms defined by $\|x\|_p = (|x_1|^p + \cdots + |x_n|^p)^{1/p}$, $1 \le p < \infty$.

The problem of finding a sparse approximation to the solution of $F\mathrm{e} = z$ is formulated in [21] as

$$\left.\begin{array}{ll} \min \|\mathrm{e}\|_0, & \mathrm{e} \in \mathbb{R}^m \\ \text{subject to} & F\mathrm{e} = z. \end{array}\right\} \tag{2.1}$$

To find a sparse approximation of such an underdetermined system, one needs to perform exhaustive searches over all subsets of columns of F, which is technically NP-hard [35]. We instead look for an alternative optimization method for minimizing $\|\mathrm{e}\|_0$ without performing exhaustive searches. One possibility is to replace the minimization of $\|\mathrm{e}\|_0$ with the min-

imization of $\|\mathrm{e}\|_p$ for some $p \geq 1$. For example, if $p = 2$, then we would be solving the following constrained optimization problem,

$$\min \|\mathrm{e}\|_2 \quad \text{subject to} \quad F\mathrm{e} = z. \tag{2.2}$$

However, the solutions for this case are almost never sparse. We illustrate this with a well-known simple example from [36].

Example 1. Suppose $f \in \mathbb{R}^{1\times 2}$ and $\mathrm{e} \in \mathbb{R}^2$. Consider the optimization problem

$$\min \|\mathrm{e}\|_p \quad \text{subject to} \quad f\mathrm{e} = z, \quad p = 1, 2.$$

The optimization problem in 2-D plane is solved graphically in Figure 2.1.

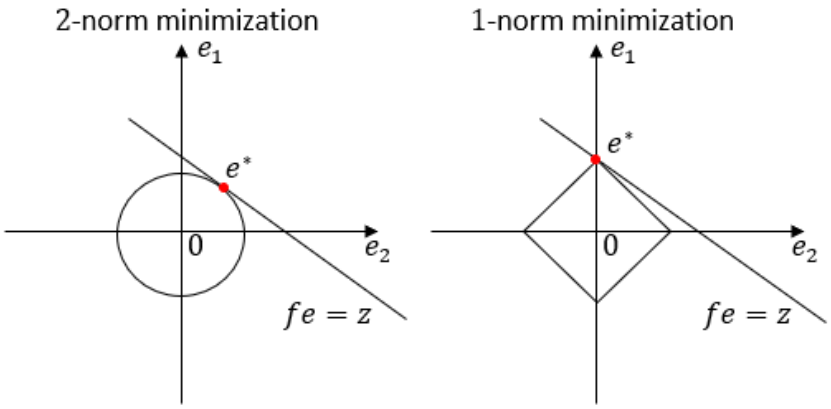


Figure 2.1. Comparison between 1-norm minimization and 2-norm minimization

In the left subplot, the line represents the equality constraint $f\mathrm{e} = z$. The solution is e^*. We can see that both entries of e^* are nonzero. This example illustrates the fact that the solution to the optimization problem of minimizing the 2-norm subject to equality constraints is almost never sparse [37].

We choose to use an optimization scheme where we minimize the 1-norm of a solution subject to $f\mathrm{e} = z$. As illustrated in Figure 2.1, the solution for the 1-norm minimization is sparse.

Therefore, instead of minimizing $\|e\|_0$, we consider an optimization problem where we minimize the 1-norm of a solution subject to the constraint, $Fe = z$. This optimization problem is known as Basis Pursuit [38]. It is usually applied in cases where there is an underdetermined system of linear equations that must be exactly satisfied. Since $\|e\|_1 = \sum_{i=1}^{m} |e_i|$ is a convex function [21], [38], we have a convex optimization problem,

$$\left.\begin{array}{ll} \min \|e\|_1, & e \in \mathbb{R}^m \\ \text{subject to} & Fe = z. \end{array}\right\} \tag{2.3}$$

Our objective is to find the unique solution e to (2.3). Once we find e, we can then recover x.

2.2 Conditions for Unique Sparse Vector Recovery

2.2.1 Existence of the unique sparse solution

We use the following definitions in our analysis.

Definition 2 (i-sparse vector)**.** [21] A vector e is i-sparse if it has at most i non-zero components, that is, $\|e\|_0 \leq i$.

Definition 3 (Spark of a matrix)**.** [39] The spark of the matrix F is the smallest number of linearly dependent columns in F.

Let $\Sigma_i = \{e : \|e\|_0 \leq i\}$ be the set of all i-sparse vectors and let $\mathcal{N}(F)$ denote the null space of the matrix F. We have the following lemma.

Lemma 1. *If $\Sigma_{2i} \cap \mathcal{N}(F) = \{0\}$, then any* i*-sparse solution to the underdetermined system $Fe = z$ is unique.*

Proof. Suppose $e^{(1)}$ and $e^{(2)}$ are i-sparse solutions to the under-determined system $Fe = z$. Then $F(e^{(1)} - e^{(2)}) = 0$ and thus $e^{(1)} - e^{(2)} \in \mathcal{N}(F)$. Since $e^{(1)}$ and $e^{(2)}$ are in Σ_i, we also have $e^{(1)} - e^{(2)} \in \Sigma_{2i}$ and therefore $e^{(1)} - e^{(2)} \in \Sigma_{2i} \cap \mathcal{N}(F) = \{0\}$. It follows that $e^{(1)} = e^{(2)}$ and thus any i-sparse solution to the underdetermined system $Fe = z$ is unique. □

Remark 1. *Note that spark*$(F) > 2\mathrm{i}$ *is equivalent to* $\Sigma_{2\mathrm{i}} \cap \mathcal{N}(F) = \{0\}$. *Therefore, by Lemma 1, spark*$(F) > 2\mathrm{i}$ *implies that the* i*-sparse solution to* $F\mathrm{e} = z$ *is unique.*

The above observation appears in [40] as Corollary 1.

2.2.2 Conditions for sparse vector recovery

We present the following theorem from Zhang [41] that we use in our subsequent discussion.

Theorem 1. *Let* $A \in \mathbb{R}^{m\times n}$ *be independent and identically distributed (i.i.d.) normal or be a full (column) rank matrix such that* $FA = 0$ *for an i.i.d. normal* $F \in \mathbb{R}^{(m-n)\times m}$. *Then there exist absolute positive constants* c_0 *and* c_1 *(independent of* m *and* n*) such that if* $\|\mathrm{e}\|_0 < \frac{c_1^2}{4}\frac{n}{1+\log(m/n)}$, *then the* 1*-norm minimization recovers the sparse vector* e *with probability at least* $1 - exp(-c_0(m-n))$.

Remark 2. *It is worth noting that Zhang's Theorem says that if we perform an experiment by generating an* $m \times n$ *matrix* A *with random independent normal entries, or if we generate an* $(m-n)\times m$ *matrix* F *with random independent normal entries and* A *satisfies* $FA = 0$, *and take* e *to satisfy the norm inequality* $\|\mathrm{e}\|_0 < \frac{c_1^2}{4}\frac{n}{1+\log(m/n)}$, *then the* 1*-norm minimization recovers the sparse vector* e *with probability at least* $1 - exp(-c_0(m-n))$. *In the frequentist interpretation, this says that if we repeat the experiment a large number of times, then the rate of successful* 1*-norm minimization recovery should be at least* $1 - exp(-c_0(m-n))$. *The theorem says nothing whatsoever if we had started with a fixed matrix* A, *or if the entries of* A *or* F *are not generated by i.i.d. normal random variables.*

Remark 3. *Zhang [41, p. 91] discusses the issue of calculating the constants* c_0 *and* c_1. *An in-depth treatment of this problem is provided by Donoho and Tanner [42].*

Next, we present a method of recovering a sparse vector. We first take the QR decomposition of A to obtain:

$$A = QR = \begin{bmatrix} Q_1 & Q_2 \end{bmatrix} \begin{bmatrix} R_1 \\ 0 \end{bmatrix}, \tag{2.4}$$

where $Q \in \mathbb{R}^{m\times m}$ is orthogonal, $Q_1 \in \mathbb{R}^{m\times n}$, $\quad Q_2 \in \mathbb{R}^{m\times(m-n)}$, and $R_1 \in \mathbb{R}^{n\times n}$ is a full rank upper triangular matrix.

Lemma 2. *Let Q_2 be defined as above. Then $Q_2^\top$ is a left annihilator of A, that is, $Q_2^\top A = 0$.*

Proof. Since $Q = \begin{bmatrix} Q_1 & Q_2 \end{bmatrix}$ is orthogonal, we have

$$Q^\top Q = \begin{bmatrix} Q_1^\top Q_1 & Q_1^\top Q_2 \\ Q_2^\top Q_1 & Q_2^\top Q_2 \end{bmatrix} = \begin{bmatrix} I_n & 0 \\ 0 & I_{m-n} \end{bmatrix}.$$

Then

$$Q_2^\top A = Q_2^\top QR = \begin{bmatrix} Q_2^\top Q_1 & Q_2^\top Q_2 \end{bmatrix} \begin{bmatrix} R_1 \\ 0 \end{bmatrix},$$

which implies

$$Q_2^\top A = \begin{bmatrix} 0 & I_{m-n} \end{bmatrix} \begin{bmatrix} R_1 \\ 0 \end{bmatrix} = 0_{(m-n)\times n},$$

which completes the proof. □

Remark 4. *Another possible method to obtain a left annihilator of the matrix A is to use the singular value decomposition (SVD). Let*

$$A = U\Sigma V^\top = \begin{bmatrix} U_1 & U_2 \end{bmatrix} \begin{bmatrix} \Sigma_+ & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} V_1 & V_2 \end{bmatrix}^\top,$$

where $U \in \mathbb{R}^{m\times m}$ is an orthogonal matrix, $U_1 \in \mathbb{R}^{m\times n}$, $U_2 \in \mathbb{R}^{m\times(m-n)}$. It is easy to see that $U_2^\top$ is a left annihilator of A.

Premultiply $b = Ax + \mathrm{e}$ by $Q_2^\top$ to obtain

$$Q_2^\top b = Q_2^\top \mathrm{e}. \tag{2.5}$$

To proceed, we need the following lemmas.

Lemma 3. *[43, p. 690]. A real random square matrix with i.i.d normal entries is nonsingular with probability 1.*

Lemma 4. *[44, p. 257]. The random variables $x_1, \ldots, x_n$ are jointly normal if and only if for all $a_1, \ldots, a_n$, not all zero, the sum $a_1x_1 + \cdots + a_nx_n$ is a normal random variable.*

Premultiplying both sides of (2.5) by a randomly generated normal square matrix G whose elements are i.i.d. gives $GQ_2^\top \mathrm{e} = GQ_2^\top b$, where $Q_2^\top \in \mathbb{R}^{(m-n)\times m}$. Let $F = GQ_2^\top$ and $z = GQ_2^\top b$, then we obtain $F\mathrm{e} = z$. Applying Sylvester's inequalities (see, for example, [45, p. 655]), Lemma 3, and Lemma 4, we conclude that $F \in \mathbb{R}^{(m-n)\times m}$ is a full row rank matrix with probability 1 whose entries are i.i.d. normal. By Theorem 1, we conclude that with the probability at least $1 - \exp\{-c_0(m-n)\}$, the sparse vector e, where $\|\mathrm{e}\|_0 < (c_1^2/4)n/\left(1 + \log(m/n)\right)$, can be recovered by solving problem (2.3) with the matrix $F = GQ_2^\top$. For a further discussion of this method, see [46].

It follows from the above discussion that when $m - n$ increases, then we have higher probability of recovering the vector e. However, in order to recover the unique k-sparse vector, it follows from Lemma 1 that we first need to ensure that $\mathrm{spark}(F) > 2k$. This means we need to have a sufficiently large n for a fixed m. We next introduce the methods for solving (2.3).

2.3 Methods For Sparse Vector Recovery

2.3.1 Steepest descent method for sparse vector recovery

Consider the optimization problem (2.3), where $F \in \mathbb{R}^{(m-n)\times m}$, $\mathrm{spark}(F) > 2\mathrm{i}$, and e is a i-sparse vector. The first method for solving (2.3) is based on the steepest descent algorithm for constrained optimization [19]. Before presenting the method, we recall the following definition:

Definition 4 (Orthogonal Projection)**.** [47] Let $\mathcal{V}$ be a subspace of $\mathbb{R}^n$ and $v \in \mathcal{V}$. Then the orthogonal complement of $\mathcal{V}$, denoted $\mathcal{V}^\perp$, consists of all vectors that are orthogonal to every vector in $\mathcal{V}$, that is, $\mathcal{V}^\perp = \{x : v^\top x = 0 \quad \text{for all} \quad v \in \mathcal{V}\}$. As $\mathcal{V}$ and $\mathcal{V}^\perp$ span $\mathbb{R}^n$, every vector $x \in \mathbb{R}^n$ can be uniquely represented as $x = x_1 + x_2$, where $x_1 \in \mathcal{V}$ and $x_2 \in \mathcal{V}^\perp$. The orthogonal projection of x on $\mathcal{V}$ and $\mathcal{V}^\perp$ are x_1 and x_2, respectively.

The algorithm for solving (2.3) can be described as follows. Let $\mathrm{e}^{(1)}$ be an initial point satisfying the constraint in (2.3). Then to find the next iteration $\mathrm{e}^{(2)}$, we first calculate the sub-gradient $g^{(1)}$, where

$$g^{(1)} = \nabla \|\mathrm{e}^{(1)}\|_1 = \begin{bmatrix} -\mathrm{sign}(\mathrm{e}^{(1)})_1 & \cdots & -\mathrm{sign}(\mathrm{e}^{(1)})_m \end{bmatrix}^\top. \tag{2.6}$$

We project $g^{(1)}$ onto the null-space $\mathcal{N}(F)$ of the matrix F and then minimize e in the direction of the projected gradient. More explicitly, we decompose $g^{(1)}$ as

$$g^{(1)} = {g_N}^{(1)} + {g_{N^\perp}}^{(1)}, \tag{2.7}$$

where ${g_N}^{(1)} \in \mathcal{N}(F)$ and ${g_{N^\perp}}^{(1)} \in \mathcal{N}^\perp(F)$. It is shown in Kolev [19] that for a full row rank matrix F, ${g_{N^\perp}}^{(1)} = F^\top (FF^\top)^{-1} F g^{(1)}$. From (2.7), we see that the orthogonal projection of $g^{(1)}$ on $\mathcal{N}(F)$ is $p^{(1)} = [I - F^\top (FF^\top)^{-1} F] g^{(1)}$. The initial point $\mathrm{e}^{(1)}$ can be taken as $\mathrm{e}^{(1)} = F^\top (FF^\top)^{-1} z$. For the $(k+1)^{th}$ iteration ($k = 1, 2, \ldots$), $\mathrm{e}^{(k+1)} = \mathrm{e}^{(k)} + \alpha_k p^{(k)}$, and

$$p^{(k)} = [I - F^\top (FF^\top)^{-1} F] g^{(k)}. \tag{2.8}$$

To determine α_k, we minimize

$$f(\mathrm{e}^{(k+1)}) = \|\mathrm{e}^{(k)} + \alpha_k p^{(k)}\|_1 = \sum_{\mathrm{i}=1}^{m} |{\mathrm{e}_\mathrm{i}}^{(k)} + \alpha_k {p_\mathrm{i}}^{(k)}|.$$

Let $\phi = \{\mathrm{i} : \mathrm{sign}({\mathrm{e}_\mathrm{i}}^{(k)}) \neq \mathrm{sign}({p_\mathrm{i}}^{(k)}), \quad \mathrm{i} = 1, 2, \cdots, m\}$. Using the method in [19], we calculate α_k as ${\alpha_k}^\mathrm{i} = -\frac{{\mathrm{e}_\mathrm{i}}^{(k)}}{{p_\mathrm{i}}^{(k)}} > 0$. Let $f_k = \min_{\mathrm{i} \in \phi} {f_k}^\mathrm{i}$. Then, α_k is chosen to be equal to ${\alpha_k}^\mathrm{i}$ for which ${f_k}^\mathrm{i} = f_k$. The algorithm terminates when $f_{k+1} \geq f_k$.

2.3.2 The 1-norm minimization for sparse vector recovery

The second method for solving (2.3) presented in [20] uses a class of penalty functions to transform constrained optimization minimum norm problems into unconstrained optimization problems.

Let $r = -F^\top (FF^\top)^{-1} z \in \mathbb{R}^m$. Then $Fr = -z$. Let $x = \mathrm{e} + r$. Then the constrained problem (2.3) takes the form

$$\min \|x - r\|_1, \quad x \in \mathbb{R}^m$$

$$\text{subject to} \quad Fx = 0.$$

It is shown in [20] that the above constrained optimization problem is equivalent to an unconstrained problem of the form, $\min(\|x - r\|_1 + c\|Fx\|_s)$, where $s \geq 1$ is an integer and the value of c is given in [20] as $c = \frac{K}{\sqrt{\lambda_{\min}(FF^\top)}}$, where

$$K = \begin{cases} n^{(1/2)} & \text{for} \quad s < 2 \\ n^{(1/2)} m^{(1/2)-(1/s)} & \text{for} \quad s \geq 2. \end{cases}$$

Here $\lambda_{\min}(FF^\top)$ denotes the minimal eigenvalue of the positive definite matrix $FF^\top$.

We can use the MATLAB optimization toolbox function `fminunc` to solve the above unconstrained optimization problem.

2.3.3 Comparison of the methods

We use an example from Kolev [19] to test our methods for sparse vector recovery, which is equivalent to solving optimization problem (2.3), where

$$F = \begin{bmatrix} 2 & -1 & 4 & 0 & 3 & 1 \\ 5 & 1 & -3 & 1 & 2 & 0 \\ 1 & -2 & 1 & -5 & -1 & 4 \end{bmatrix}$$

and $z = \begin{bmatrix} 2 & 1 & -4 \end{bmatrix}^\top$.

We first use the projected steepest descent algorithm to solve the above optimization problem. Performing manipulations described in the algorithm we obtain $\mathrm{e}^{(1)}$ of the form

$$\begin{bmatrix} 0.08825 & 0.1083 & 0.2733 & 0.5047 & 0.3828 & -0.3097 \end{bmatrix}^\top.$$

Then we obtain $\mathrm{e}^{(2)}$,

$$\begin{bmatrix} 0.0502 & 0.0000 & 0.2207 & 0.5564 & 0.4274 & -0.2654 \end{bmatrix}^\top.$$

At this point, we have $\alpha_k < 10^{-7}$ for $k \geq 2$. The convergence of the iterative process, $\mathrm{e}^{(k+1)} = \mathrm{e}^{(k)} + \alpha^{(k)} p^{(k)}$, becomes very slow. In order to accelerate the algorithm convergence, we proceed as follows. If for each i, we have $|\mathrm{e}_\mathrm{i}^{(k+1)} - \mathrm{e}_\mathrm{i}^{(k)}| > 10^{-5}$, then we resume the

iterative process. We let $\mathrm{e}_2^{(2)} = 0$ and delete from the matrix F the column corresponding to $\mathrm{e}_2^{(2)} = 0$. We obtain

$$F_1 = \begin{bmatrix} 2 & 4 & 0 & 3 & 1 \\ 5 & -3 & 1 & 2 & 0 \\ 1 & 1 & -5 & -1 & 4 \end{bmatrix}$$

and the initial point

$$\mathrm{e}^{(2)_0} = \begin{bmatrix} 0.0502 & 0.2207 & 0.5564 & 0.4274 & -0.2654 \end{bmatrix}^\top.$$

After one iteration, we face the same situation as before, where

$$\mathrm{e}^{(2)_1} = \begin{bmatrix} 0.0000 & 0.1851 & 0.5914 & 0.4820 & -0.1866 \end{bmatrix}^\top.$$

Using the same strategy, the optimal solution is found to be $\begin{bmatrix} 0.1923 & 0.7564 & 0.4103 & 0 \end{bmatrix}^\top$. The optimal solution to the original problem is

$$\mathrm{e}^* = \begin{bmatrix} 0 & 0 & 0.1923 & 0.7564 & 0.4103 & 0 \end{bmatrix}^\top. \tag{2.9}$$

We next use the 1-norm minimization method to solve the same problem. We first construct the unconstrained optimization problem, that is, $\min(\|x - r\|_1 + c\|Fx\|_s)$. We compute c and r for $s = 2$, where $c = 0.4907$ with $\lambda_{\min}(FF^\top) = 24.945$ and $K = \sqrt{6}$. The vector r is

$$\begin{bmatrix} 0.08825 & 0.1083 & 0.2733 & 0.5047 & 0.3828 & -0.3097 \end{bmatrix}^\top.$$

We use the MATLAB toolbox function `fminunc` to solve the above unconstrained problem. After 28 iterations, we obtain the same optimal solution given by (2.9) to four decimal places.

The original projected steepest descent method had to be modified to accelerate its convergence. The 1-norm minimization method, on the other hand, does not require line search in each iteration. In addition, this algorithm can easily be implemented using MATLAB `fminunc` function. After comparing the two methods, we decided to use the 1-norm minimization method in our further simulations.

In summary, to reconstruct the unknown sparse error vector e in a corrupted linear system $b = Ax + \mathrm{e}$, we perform the following steps:

- Use Lemma 2 to find the left annihilator, $Q_2^\top$, of the matrix A
- Construct optimization problem (2.3).
- Solve optimization problem (2.3) for e using the method from Subsection 2.3.2.

2.3.4 An example illustrating finding the unique sparse solution

We begin by generating a random matrix $F \in \mathbb{R}^{l\times m}$, where $l = m - n$. The entries of F are i.i.d. normal. In our simulations, we used the following parameters: $m = 18$, $k = 1$, and $\|\mathrm{e}\|_1 = 1$. For different values of $l = 2, 3, \ldots, 12$, we perform 500 iterations, where in each iteration, the vector e is randomly generated. The objective of this exercise is to find a matrix F with l rows and $m = 18$ columns that satisfies conditions of Lemma 1 and Theorem 1.

In Figure 2.2 and 2.3, we show simulation results for $l = 2$ and 12, respectively. In each figure, the first subfigure shows the true sparse vector, the middle subfigure shows the estimated sparse vector, a blue square indicates that an error occurring at a given sample. For each figure, the bottom subfigure shows the estimation error. Note that for $l = 12$, in Figure 2.3, there is no estimation error. The estimation errors decrease when l is increased form 2 to 12.

From this study, we conclude that for the normal random matrix $F \in \mathbb{R}^{l\times m}$, when $m = 18$ and $l = 12$, conditions of Lemma 1 and Theorem 1 hold. We will use this result in the following section.

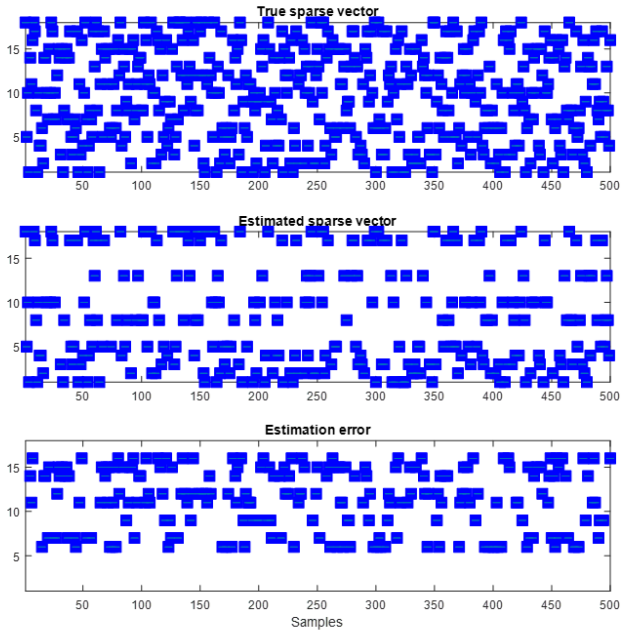


Figure 2.2. Case $l = 2$. Top subfigure shows true sparse error, the middle subfigure shows estimated sparse error, the bottom subfigure shows estimation error. A blue square indicates an error occurring at a given sample.

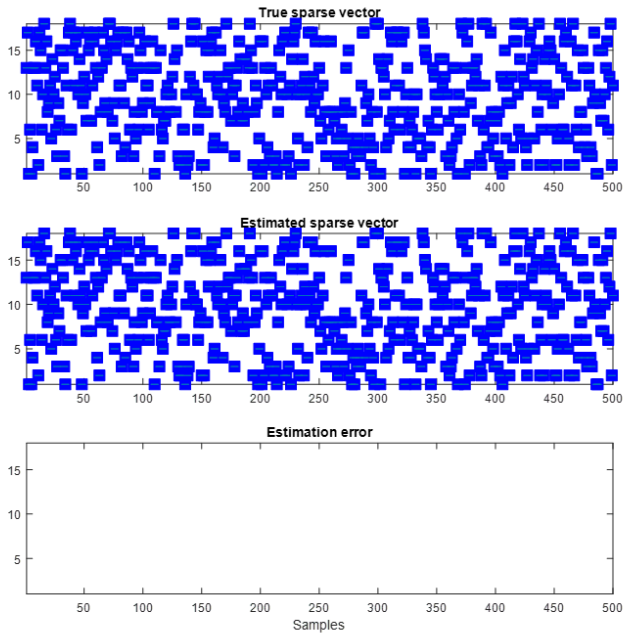


Figure 2.3. Case $l = 12$. Top subfigure shows true sparse error, the middle subfigure shows estimated sparse error, the bottom subfigure shows estimation error. A blue square indicates an error occurring at a given sample.

2.4 An Application to Secure State Estimation

We apply the method for the vector recovery for a linear system corrupted by unknown sparse error to obtain state estimation of a CPS with sparse malicious packet drops.

The system under consideration is a linear discrete-time dynamical control system modeled as:

$$\left.\begin{aligned} x[k+1] &= A_d x[k] + B_d u^a[k] \\ y^s[k] &= C_d x[k] \end{aligned}\right\} \tag{2.10}$$

where $A_d \in \mathbb{R}^{n\times n}$, $B_d \in \mathbb{R}^{n\times m}$, $C_d \in \mathbb{R}^{p\times n}$, $y^s[k] \in \mathbb{R}^p$ is the output measured by sensors, $u^a[k] \in \mathbb{R}^m$ is the input received by the actuators, $y^c[k] \in \mathbb{R}^p$ is the controller output, and $u^c[k] \in \mathbb{R}^m$ is the controller signal. A block diagram of the control system under consideration is shown in Figure 2.4, which is adapted from [10].

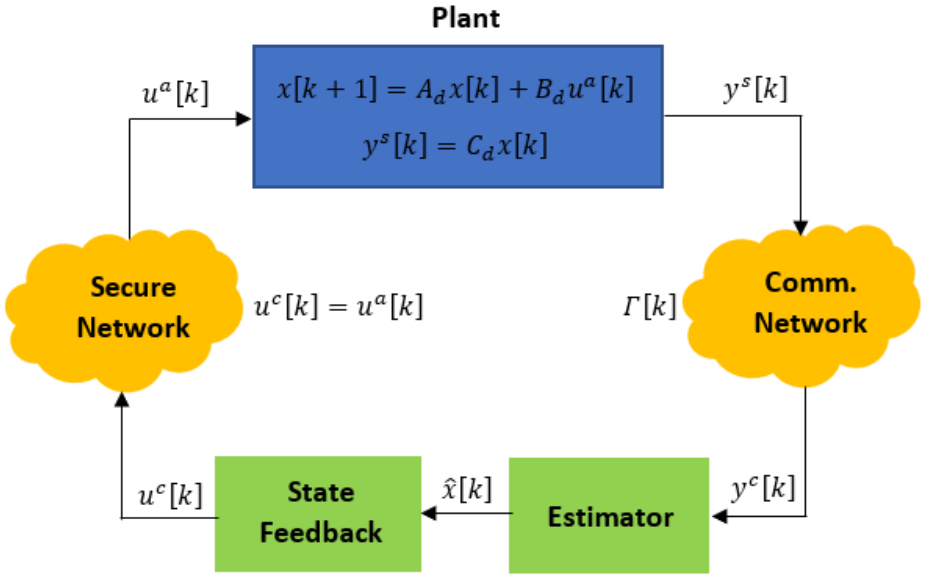


Figure 2.4. Block diagram of a wireless control system.

We assume that the pair (A_d, C_d) is observable and that the connection between the actuator and the controller is secure, which implies that $u^a[k] = u^c[k]$. Following [10], we model the malicious packet drops in the communication flow from the sensor to the controller by means of the matrix $\Gamma[k] = \text{diag}\{\gamma_1[k], \gamma_2[k], \cdots, \gamma_p[k]\} \in \mathbb{R}^{p\times p}$, where $\gamma_i[k], i = 1, \ldots, p$, are Boolean variables, where $\gamma_i[k] = 1$ if the packet is correctly received by the controller,

and $\gamma_{\mathrm{i}}[k] = 0$ if the packet is dropped. Therefore, the system under consideration can be modeled as

$$\left.\begin{array}{rcl} x[k+1] &=& A_d x[k] + B_d u^c[k] \\ y^c[k] &=& \Gamma[k] C_d x[k]. \end{array}\right\} \tag{2.11}$$

Let $\overline{\Gamma}[k] = \Gamma[k] - I_p$. Because the control output $y^c[k]$ is known at all time instants k, we have τ observations for the above system, which can be represented as

$$\begin{aligned} y^c|_{[k-\tau+1,k]} &= \begin{bmatrix} C_d \\ C_d A_d \\ \vdots \\ C_d A_d^{\tau-1} \end{bmatrix} x[k-\tau+1] \\ &+ \begin{bmatrix} \overline{\Gamma}[k-\tau+1] C_d x[k-\tau+1] \\ \overline{\Gamma}[k-\tau+2] C_d x[k-\tau+2] \\ \vdots \\ \overline{\Gamma}[k] C_d x[k] \end{bmatrix} + \begin{bmatrix} 0 \\ C_d B_d u^c[k-\tau+1] \\ \vdots \\ \Sigma_{\mathrm{i}=1}^{\tau-1} C_d A_d^{\tau-1-\mathrm{i}} B_d u^c[k-\tau+\mathrm{i}] \end{bmatrix}. \end{aligned}$$

Let $v = [0 \ \cdots \ \Sigma_{\mathrm{i}=1}^{\tau-1} C_d A_d^{\tau-1-\mathrm{i}} B_d u^c[k-\tau+\mathrm{i}]]^\top$ denote the last term in the above expression. Note that v is known for all k and τ and can thus be removed from the observation vector y^c. Let $\hat{y}^c|_{[k-\tau+1,k]} = y^c|_{[k-\tau+1,k]} - v$ and let

$$\begin{aligned} Y_k &\triangleq \begin{bmatrix} \hat{y}^c[k] \\ \hat{y}^c[k-1] \\ \vdots \\ \hat{y}^c[k-\tau+1] \end{bmatrix} \\ &= \begin{bmatrix} C_d A_d^{\tau-1} \\ C_d A_d^{\tau-2} \\ \vdots \\ C_d \end{bmatrix} x[k-\tau+1] + I_{\tau p} \begin{bmatrix} \overline{\Gamma}[k] C_d x[k] \\ \overline{\Gamma}[k-1] C_d x[k-1] \\ \vdots \\ \overline{\Gamma}[k]-\tau+1] C_d x[k-\tau+1] \end{bmatrix} \\ &\triangleq \mathcal{O}^{\tau-1} x[k-\tau+1] + I_{\tau p} E_s[k], \end{aligned} \tag{2.12}$$

where $Y_k \in \mathbb{R}^{\tau p}$, $E_s[k] \in \mathbb{R}^{\tau p}$ and $\mathcal{O}^{\tau-1} \in \mathbb{R}^{\tau p \times n}$ is the τ-step observation matrix with full column rank.

We assume that $\tau p > n$. We now apply Lemma 2. We first perform the QR decomposition of $\mathcal{O}^{\tau-1}$ to obtain

$$\mathcal{O}^{\tau-1} = QR = \begin{bmatrix} Q_1 & Q_2 \end{bmatrix} \begin{bmatrix} R_1 \\ 0 \end{bmatrix}, \tag{2.13}$$

where $Q \in \mathbb{R}^{\tau p \times \tau p}$ is orthogonal, $Q_1 \in \mathbb{R}^{\tau p \times n}$, $Q_2 \in \mathbb{R}^{\tau p \times (\tau p - n)}$, and $R_1 \in \mathbb{R}^{n \times n}$ is an upper triangular matrix with full rank. Pre-multiplying (2.12) by $Q_2^\top$, we obtain

$$Q_2^\top Y_k = Q_2^\top E_s[k]. \tag{2.14}$$

Next, we construct the matrix F as $F = GQ_2^\top$, where $Q_2^\top \in \mathbb{R}^{(\tau p - n) \times \tau p}$, $G \in \mathbb{R}^{(\tau p - n) \times (\tau p - n)}$ whose entries are i.i.d normal. By Lemma 3 and Lemma 4, the matrix $F \in \mathbb{R}^{(\tau p - n) \times \tau p}$ is a full row rank matrix with probability 1 whose entries are i.i.d. normal. Pre-multiplying both sides of (2.14) by G, we obtain

$$z[k] = \tilde{Y}_k = GQ_2^\top Y_k = FE_s[k]. \tag{2.15}$$

We assume that the number of packet drops over the time interval $[t - \tau + 1, t]$ is bounded by i_s with $\mathrm{spark}(F) > 2\mathrm{i}_s$. Applying Lemma 1, we conclude that the sparse vector $E_s[k]$ satisfying (2.15) is unique. We then search for the unique sparse solution to (2.15) by solving the optimization problem,

$$\left.\begin{aligned} &\hat{E}_s[k] = \min \|E_s[k]\|_1, \quad E_s[k] \in \mathbb{R}^{\tau p} \\ &\text{subject to} \quad FE_s[k] = z[k]. \end{aligned}\right\} \tag{2.16}$$

Then, the state estimate of $x[k - \tau + 1]$ is obtained as

$$\hat{x}[k - \tau + 1] = \mathcal{O}^\dagger (Y_k - I_{\tau p} \hat{E}_s[k]), \tag{2.17}$$

where $\mathcal{O}^\dagger = (\mathcal{O}^\top \mathcal{O})^{-1}\mathcal{O}^\top$ is the pseudo-inverse of $\mathcal{O}^{\tau-1}$. Pre-multiplying both sides of (2.12) by $\mathcal{O}^\dagger$, we obtain $x[k-\tau+1] = \mathcal{O}^\dagger(Y_k - I_{\tau p}E_s[k])$. Let $\tilde{x}[k]$ be the estimation error of the state, then we have $\tilde{x}[k-\tau+1] = \mathcal{O}^\dagger(E_s[k] - \hat{E}_s[k])$. If $E_s[k] = \hat{E}_s[k]$, then $\tilde{x}[k-\tau+1] = 0$. From the previous discussion, we know that if the hypothesis of Lemma 1 and Theorem 1 are satisfied, then the unique sparse vector $E_s[k]$ in (2.15) can be successfully recovered with probability at least $1-\exp(-c_0(m-n))$, which means $E_s[k] = \hat{E}_s[k]$ is true with probability at least $1-\exp(-c_0(m-n))$. Thus, we can correctly estimate the state $x[k-\tau+1]$ from the measurements Y_k with probability at least $1-\exp(-c_0(m-n))$. We then estimate the state $x[k]$ of the system (2.11) as follows:

$$\hat{x}[k] = A^{\tau-1}\hat{x}[k-\tau+1] + [A^{\tau-2}B \ \cdots \ B]\begin{bmatrix} u[k-\tau+1] \\ \vdots \\ u[k-1] \end{bmatrix}. \tag{2.18}$$

To illustrate the performance of our proposed state estimation method, we use a remotely controlled UAV as a CPS given in [48] and illustrated in Figure 2.4. The state vector is $x = [\theta_x, \dot{\theta}_x, \theta_y, \dot{\theta}_y, \theta_z, \dot{\theta}_z]$, where θ_x, θ_y and θ_z are the pitch, roll, and yaw angles, respectively, and $\dot{\theta}_x, \dot{\theta}_y$ and $\dot{\theta}_z$ are their corresponding angular velocities. The output $y^c = [\overline{\theta}_x, \overline{\theta}_y, \overline{\theta}_z]$ represents the corresponding angular measurements corrupted by unknown sparse malicious packet drops. We perform a simulation over 50 samples with the step size $T_s = 0.01$ second. We assume that there are i_s malicious packet drops in each sampling interval $[k, k+\tau-1]$, where $k = 0, 1, \cdots, 44$. The initial state $x[0]$ is randomly selected. Without loss of generality, we assume $u^a[k] = 0$.

We first choose $\tau = 6$ and $\mathrm{i}_s = 1$ in order to satisfy the condition of the sparse error recovery as discussed in Section 2.3.4.

Figure 2.5 shows the estimation errors. The red, blue and black crosses represent the estimation errors for θ_x, θ_y and θ_z respectively. In our simulation, we assume $y[-5] = y[-4] = \cdots = y[-1] = 0$. This explains the initial observation errors.

We then choose $\tau = 6$ and $\mathrm{i}_s = 6$ in order to show that when the malicious packet drops are more severe, we will have difficulties with correct state estimation. In Figure 2.6, incorrectly estimated states are observed when $k \geq 5$.

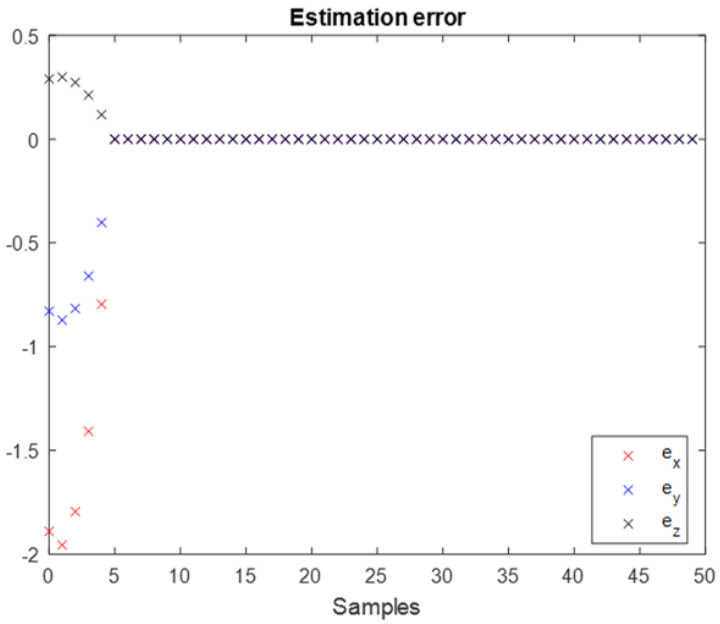


Figure 2.5. Estimation errors for θ_x, θ_y and θ_z. One packet drop in each sampling interval $[k, k+5]$.

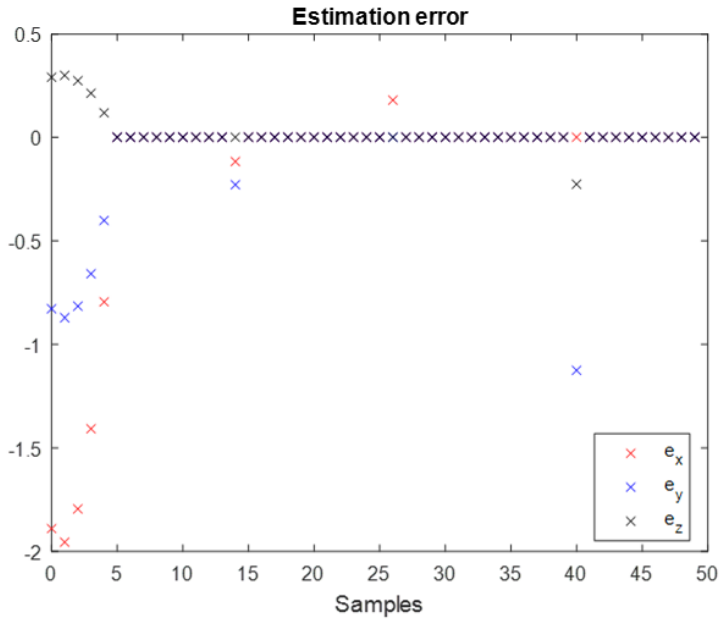


Figure 2.6. Estimation errors for θ_x, θ_y and θ_z. Six packet drops in each sampling interval $[k, k+5]$.

Let $\eta_{\mathrm{e}} = \|E_s[k]\|_0/(\tau p)$ be the percentage of malicious packet drops. We conclude that when $\tau = 6$, malicious packet drops are recovered with $100\%, 98.7\%$, and 86.9% accuracy rates when $\eta_{\mathrm{e}} = 5.6\%, 11.1\%$, and 33.3%, respectively.

2.5 Conclusions

We propose a method for the recovery of the unknown sparse vector in an overdetermined linear system by converting the original linear system into an equivalent convex optimization problem with equality constraints. The proposed method is applied to the secure state estimation of a CPS in the presence of malicious packet drops for the output measurements. In the next Chapter, we extend the method for the case when there are sparse malicious packet drops between the controller and the actuator.

3. NORM-UIO BASED OBSERVERS FOR CYBER-PHYSICAL SYSTEMS CORRUPTED BY UNKNOWN INPUT AND OUTPUT SPARSE ERRORS

3.1 Introduction

In this chapter, we consider the problem of designing observers for plants for which the sensor measurements and actuator inputs are corrupted by sparse errors. This could be a case of a CPS experiencing malicious packet drops between the controller and the plant actuators and malicious packet drops between the plant sensors and the controller, see for example [9], [10], [12], [49], [50]. State estimation for systems with packet drops during output transmission is considered in [51], where the state estimator is designed using a projection method and minimum mean square error estimation with the estimator gain designed using the Riccati equation. State estimators for nonlinear systems are proposed in [52] with output packet drops whose design is based on extended Kalman filter and moving horizon estimation. Different CT observers for plants with unknown inputs have been proposed in [31], [53]–[59]. Discrete-time unknown input observers were proposed in [26], [28], [60].

Our objective is first to use the sparse vector recovery method to recover the unknown sparse error corrupting the plant output measurements and the input to the plant actuators. Then, an unknown input observer (UIO) is designed to estimate the state of the plant. We propose a novel state observer architecture that combines an output sensor error approximator with an unknown input observer (UIO) structure. The attractive feature of the proposed novel state observer architecture is that it can be used in CPSs whose inputs and outputs are simultaneously corrupted by sparse malicious packet drops. The approximator recovers the unknown sparse error vector e_s corrupting the output measurements y_c, see Figure 3.1.

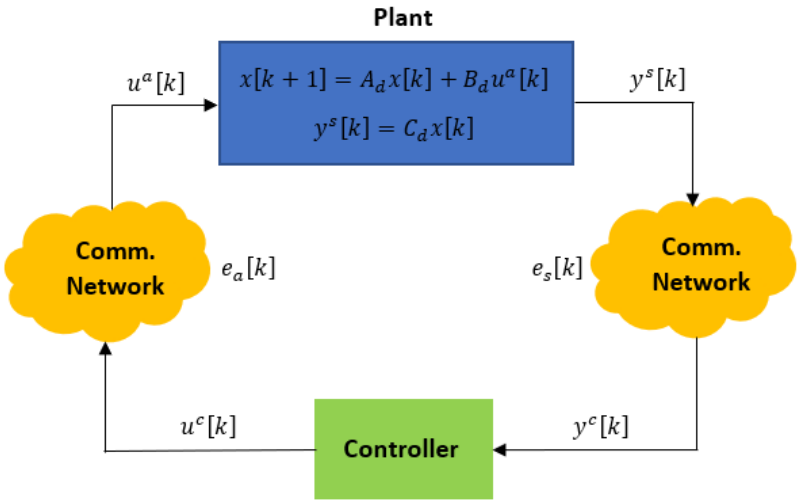


Figure 3.1. A block diagram of a CPS with unknown input and output sparse errors.

3.2 Problem Statement

We consider a linear DT plant model,

$$\left.\begin{aligned} x[k+1] &= A_d x[k] + B_d u^a[k] \\ y^s[k] &= C_d x[k] \end{aligned}\right\} \tag{3.1}$$

where $A_d \in \mathbb{R}^{n\times n}$, $B_d \in \mathbb{R}^{n\times m}$ has full column rank, $C_d \in \mathbb{R}^{p\times n}$. $u^a[k] \in \mathbb{R}^m$ is the input received by actuators, $y^s[k] \in \mathbb{R}^p$ is the output measured by sensors. We assume the plant model (3.1) is controllable and observable–see, for example [61, Subsection 1.1.2 and Chapter 2] or [62] for a discussion on modeling of DT systems.

We consider a scenario in which sensor measurements, $y^s[k]$, are being sent to the controller through a communication network. We assume the presence of malicious attacker that causes packet drops in the communication network. As in Chapter 2, we model this malicious packet drops by means of the matrix $\Gamma(k) = \text{diag}\{\gamma_1(k), \gamma_2(k), \cdots, \gamma_p(k)\}$, where $\gamma_i(k), i = 1, \ldots, p$ are Boolean variables, where $\gamma_i(k) = 1$ if the packet is correctly received; $\gamma_i(k) = 0$ if the packet is dropped by the controller. Therefore, the signal received by the controller is $y^c[k] = \Gamma(k)y^s[k]$. Similarly, the control signal is being sent to the plant through a communication network, where we also assume malicious packet drops modeled by the ma-

trix $\Lambda(k) = \text{diag}\{\lambda_1(k), \lambda_2(k), \cdots, \lambda_m(k)\}$, where $\lambda_{\text{i}}(k), \text{i} = 1, \ldots, m$ are Boolean variables, where $\lambda_{\text{i}}(k) = 1$ if the packet is correctly received; $\lambda_{\text{i}}(k) = 0$ if the packet is dropped by the actuator. Therefore, the signal received by the actuator is $u^a[k] = \Lambda(k)u^c[k]$. A block diagram of the CPS under consideration is shown in Figure 3.1.

The network communication errors in the communication flow from the sensor to the controller and from the controller to the actuator are presented as $\text{e}_s[k]$ and $\text{e}_a[k]$, respectively. More specifically, we have $\text{e}_s[k] = y^c[k] - y^s[k] \in \mathbb{R}^p$ and $\text{e}_a[k] = u^a[k] - u^c[k] \in \mathbb{R}^m$. We model $\text{e}_s[k]$ and $\text{e}_a[k]$ as the malicious packet drops by means of the matrices $\Gamma(k)$ and $\Lambda(k)$. Let $\overline{\Gamma}(k) = \Gamma(k) - I_p \in \mathbb{R}^{p\times p}$ and $\overline{\Lambda}(k) = \Lambda(k) - I_m \in \mathbb{R}^{m\times m}$. Then we have,

$$\text{e}_s[k] = \overline{\Gamma}(k)y^s[k], \quad \text{e}_a[k] = \overline{\Lambda}(k)u^c[k]. \tag{3.2}$$

We assume the malicious packet drop is sparse. Here by sparse, we mean most of the entries of $\text{e}_s[k]$ and $\text{e}_a[k]$ are zeros.

The system under consideration now can be modeled as

$$\left.\begin{array}{rcl} x[k+1] & = & A_d x[k] + B_d(u^c[k] + \text{e}_a[k]) \\ y^c[k] & = & C_d x[k] + \text{e}_s[k]. \end{array}\right\} \tag{3.3}$$

Our objective is to correctly estimate the state $x[k]$ of the CPS (3.3) in the presence of the malicious packet drops modeled by $\text{e}_s[k]$ and $\text{e}_a[k]$.

3.3 Vector Recovery Strategy

In this section, we describe a method for recovering error vector $\text{e}_s[k]$ in the network system (3.3) under existence conditions.

3.3.1 Vector recovery method

Substituting $u^a[k] = \Lambda(k)u^c[k]$ and $y^c[k] = \Gamma(k)y^s[k]$ into (3.1), we obtain

$$\left.\begin{array}{rcl} x[k+1] & = & A_d x[k] + B_d \Lambda(k) u^c[k] \\ y^c[k] & = & \Gamma(k) C_d x[k]. \end{array}\right\} \tag{3.4}$$

Since the control output $y^c[k]$ and the control input $u^c[k]$ are known at all time instant k, following the discussion form Section 2.4, we collect τ observations for the above system, which can be represented as

$$\begin{aligned} y^c|_{[k-\tau+1,k]} = & \begin{bmatrix} C_d \\ C_d A_d \\ \vdots \\ C_d A_d^{\tau-1} \end{bmatrix} x[k-\tau+1] \\ & + \begin{bmatrix} \overline{\Gamma}(k-\tau+1) C_d x[k-\tau+1] \\ \overline{\Gamma}(k-\tau+2) C_d x[k-\tau+2] \\ \vdots \\ \overline{\Gamma}(k) C_d x[k] \end{bmatrix} + \begin{bmatrix} 0 \\ C_d B_d \Lambda(k-\tau+1) u^c[k-\tau+1] \\ \vdots \\ \Sigma_{\mathrm{i}=1}^{\tau-1} C_d A_d^{\tau-1-\mathrm{i}} B_d \Lambda(k-\tau+\mathrm{i}) u^c[k-\tau+\mathrm{i}] \end{bmatrix}. \end{aligned}$$

Let now $U^c[k] \in \mathbb{R}^{m\times m}$ represents a diagonal matrix whose components consist of $u^c[k]$, and $vec(\Lambda(k)) \in \mathbb{R}^m$ represents the vectorization of diagonal components of $\Lambda(k)$. Then we have

$\Lambda(k)u^c[k] = U^c[k]vec(\Lambda(k))$. Let $v[k] = [0 \ \cdots \ \Sigma_{\mathrm{i}=1}^{\tau-1}(C_d A_d^{\tau-1-\mathrm{i}} B_d u^c[k-\tau+\mathrm{i}])^\top]^\top$. Note that $v[k]$ is known for all k and τ. Let $\hat{y}^c|_{[k-\tau+1,k]} = y^c|_{[k-\tau+1,k]} - v[k]$ and define $Y[k]$ as

$$
\begin{aligned}
Y[k] \triangleq \begin{bmatrix} \hat{y}^c[k] \\ \hat{y}^c[k-1] \\ \vdots \\ \hat{y}^c[k-\tau+1] \end{bmatrix} &= \begin{bmatrix} C_d A_d^{\tau-1} \\ C_d A_d^{\tau-2} \\ \vdots \\ C_d \end{bmatrix} x[k-\tau+1] \\
&+ I_{\tau p} \begin{bmatrix} \overline{\Gamma}(k) C_d x[k] \\ \overline{\Gamma}(k-1) C_d x[k-1] \\ \vdots \\ \overline{\Gamma}(k-\tau+1) C_d x[k-\tau+1] \end{bmatrix} + F[k] \begin{bmatrix} vec(\overline{\Lambda}(k-1)) \\ vec(\overline{\Lambda}(k-2)) \\ \vdots \\ vec(\overline{\Lambda}(k-\tau+1)) \end{bmatrix} \\
&\triangleq O^{\tau-1} x[k-\tau+1] + I_{\tau p} E_s[k] + F[k]\mathcal{V}[k], \qquad (3.5)
\end{aligned}
$$

where $O^{\tau-1} \in \mathbb{R}^{\tau p \times n}$, $Y[k] \in \mathbb{R}^{\tau p}$, $F[k] \in \mathbb{R}^{\tau p \times (\tau-1)m}$ and

$$
F[k] = \begin{bmatrix} C_d B_d U^c[k-1] & \cdots & C_d A_d^{\tau-2} B_d U^c[k-\tau+1] \\ \vdots & \ddots & \vdots \\ 0_{p\times m} & \cdots & C_d B_d U^c[k-\tau+1] \\ 0_{p\times m} & \cdots & 0_{p\times m} \end{bmatrix}.
$$

Let $\Omega[k] = [I_{\tau p} \quad F[k]]$ and $E[k] = [E_s^\top[k] \quad \mathcal{V}^\top[k]]^\top$, then

$$
Y[k] = O^{\tau-1} x[k-\tau+1] + \Omega[k] E[k], \qquad (3.6)
$$

where $\Omega \in \mathbb{R}^{\tau p \times [\tau p + (\tau-1)m]}$ and $E \in \mathbb{R}^{\tau p + (\tau-1)m}$.

To proceed, we take the QR decomposition of $O^{\tau-1}$ to obtain

$$
O^{\tau-1} = QR = \begin{bmatrix} Q_1 & Q_2 \end{bmatrix} \begin{bmatrix} R_1 \\ 0 \end{bmatrix}, \qquad (3.7)
$$

where $Q \in \mathbb{R}^{\tau p \times \tau p}$ is orthogonal, $Q_1 \in \mathbb{R}^{\tau p \times n}$, $Q_2 \in \mathbb{R}^{\tau p \times (\tau p - n)}$, and $R_1 \in \mathbb{R}^{n \times n}$ is a full rank upper triangular matrix.

Premultiplying (3.6) by $Q_2^\top$, we obtain $Q_2^\top Y[k] = Q_2^\top \Omega[k] E[k]$. Let $Z[k] = Q_2^\top Y[k]$ and $W[k] = Q_2^\top \Omega[k]$, then we have

$$Z[k] = W[k]E[k], \tag{3.8}$$

where $Z[k] \in \mathbb{R}^{\tau p - n}$ and $W[k] \in \mathbb{R}^{(\tau p - n) \times [\tau p + (\tau - 1)m]}$. Note that $W[k]$ is full row rank, that is $\text{rank}(W[k]) = \tau p - n$. This is because $\text{rank}(Q_2^\top) = \tau p - n$, $\text{rank}(\Omega[k]) = \tau p$, then $\text{rank}(W[k]) = \text{rank}(Q_2^\top \Omega[k]) = \text{rank}(Q_2^\top)$.

It is shown in [21] that if $E[k]$ is an i-sparse vector, the solution to (3.8) can be obtained by solving the following optimization problem,

$$\min \|E[k]\|_0 \quad \text{subject to} \quad Z[k] = W[k]E[k]. \tag{3.9}$$

We assume that over the time interval $[k - \tau + 1, k]$, there are at most i_s malicious packet drops from the sensor to the controller and at most i_a malicious packet drops from the controller to the actuator. We assume that $E[k]$ is i-sparse. Hence,

$$\mathrm{i} = \|E[k]\|_0 = \|E_s[k]\|_0 + \|E_a[k]\|_0 = \mathrm{i}_s + \mathrm{i}_{E_a[k]} \leq \mathrm{i}_s + \mathrm{i}_a. \tag{3.10}$$

We present the following lemma.

Lemma 5. *[10], [49] If the solution $E[k]$ to (3.8) is i-sparse and $(\tau p - n) \geq 2(\mathrm{i}_s + \mathrm{i}_a)$ and all subsets of $2(\mathrm{i}_s + \mathrm{i}_a)$ columns of $W[k]$ are full rank, then $E[k]$ is unique.*

Proof. See [10]. □

Since finding the solution to (3.9) is technically NP-hard [35], following [21], we approximate (3.9) by solving the following 1-norm minimization problem,

$$\tilde{E}[k] = \min \|E[k]\|_1 \quad \text{subject to} \quad Z[k] = W[k]E[k]. \tag{3.11}$$

Let $\tilde{\mathrm{e}}_s[k]$ be an estimate of $\mathrm{e}_s[k]$, that is,

$$\tilde{\mathrm{e}}_s[k] = \begin{bmatrix} \tilde{E}_1[k] & \cdots & \tilde{E}_p[k] \end{bmatrix}^\top. \tag{3.12}$$

Remark 5. *Let* $\tilde{\mathrm{e}}_a[k]$ *be an approximation of* $\mathrm{e}_a[k]$*. We have,*

$$\tilde{\mathrm{e}}_a[k-1] = diag\{\tilde{E}_{\tau p+1}[k] \quad \cdots \quad \tilde{E}_{\tau p+m}[k]\}u^c[k-1]. \tag{3.13}$$

This means that the unknown input $\mathrm{e}_a[k]$ *can only be estimated with one sampling period time delay using the* 1*-norm minimization method given by* (3.11)*. Therefore, this estimate cannot be applied to* (3.3) *to cancel* $\mathrm{e}_a[k]$ *as we will do with* $\mathrm{e}_s[k]$ *later in the paper. This is the reason we use an unknown input observer (UIO) to estimate the state* $x[k]$ *in the presence of malicious packet drops modeled by* $\mathrm{e}_a[k]$*. We discuss the UIO design in the next section.*

3.3.2 Solving 1-norm minimization problem

The 1-norm minimization problem (3.11) can be represented as an equivalent linear programming problem using a well-known scheme (see, for example [47]).

Let $E_{\mathrm{i}}^+, E_{\mathrm{i}}^-$ be such that $|E_{\mathrm{i}}| = E_{\mathrm{i}}^+ + E_{\mathrm{i}}^-$, $E_{\mathrm{i}} = E_{\mathrm{i}}^+ - E_{\mathrm{i}}^-$ and $E_{\mathrm{i}}^+ E_{\mathrm{i}}^- = 0$. Then we obtain

$$\begin{aligned} \min \quad & (E_1^+ + E_1^-) + (E_2^+ + E_2^-) + \cdots + (E_q^+ + E_q^-) \\ \text{subject to} \quad & W(E^+ - E^-) = Z \\ & E^+, E^- \geq 0, \end{aligned}$$

where $E^+ = [E_1^+ \quad \cdots \quad E_q^+]^\top$, $E^- = [E_1^- \quad \cdots \quad E_q^-]^\top$, and $q = \tau p + (\tau - 1)m$. Rewriting, we get:

$$\left.\begin{aligned} \min \quad & c^\top x_{lp} \\ \text{subject to} \quad & A_{lp} x_{lp} = Z \\ & x_{lp} \geq 0, \end{aligned}\right\} \tag{3.14}$$

where $c = [1 \quad \cdots \quad 1]^\top \in \mathbb{R}^{2q}$, $A_{lp} = [W \quad -W]$, and $x_{lp} = [{E^+}^\top \quad {E^-}^\top]^\top$. Note that (3.14) is a linear programming problem that can be solved using standard methods.

3.3.3 Algorithm for sparse vector approximation

In summary, we use the following algorithm to recover the unknown sparse error vector $\mathrm{e}_s[k]$ corrupting the networked control system (3.3).

Algorithm for the output sensor error recovery

1. Choose τ such that $(\tau p - n) \geq 2(\mathrm{i}_s + \mathrm{i}_a)$
2. Use (3.6) to construct vector $Y[k]$ and matrices $O^{\tau-1}$ and $\Omega[k]$
3. Use Lemma 2 to find the left annihilator, $Q_2^\top$, of the matrix $O^{\tau-1}$
4. Construct optimization problem (3.11), where $Z[k] = Q_2^\top Y[k]$, $W[k] = Q_2^\top \Omega[k]$
5. Solve optimization problem (3.11) for $E[k]$ using the method from Subsection 3.3.2
6. Compute $\tilde{\mathrm{e}}_s[k]$ that approximates $\mathrm{e}_s[k]$ using (3.12).

3.4 Combined Approximator and UIO Design

In this section, we present a novel state observer architecture that combines an output sparse error approximator with an unknown input observer (UIO) structure. We give existence conditions for the proposed UIO.

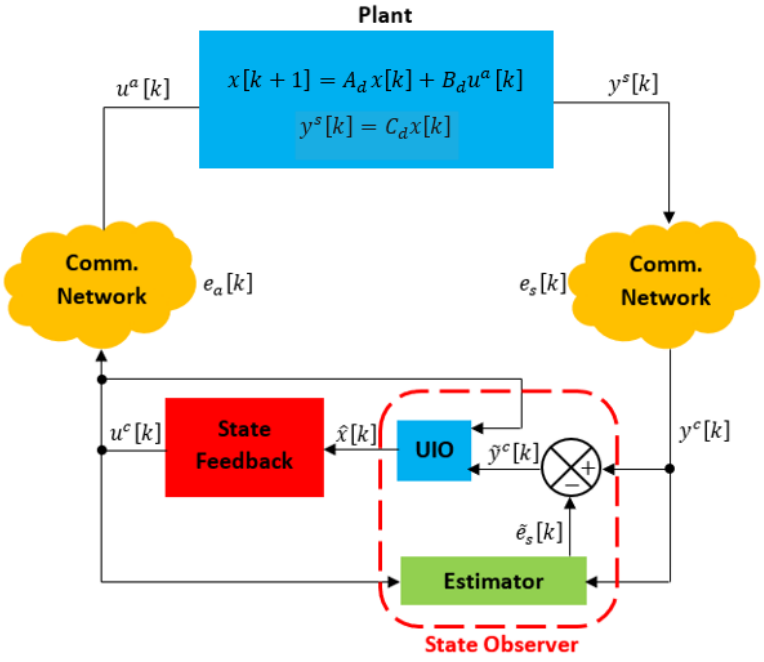


Figure 3.2. Closed-loop system with a state observer consisting of an UIO and the output sparse error approximator.

The proposed state observer architecture is depicted in Figure 3.2. We first use the algorithm from Subsection 3.3.3 to design an approximator of $\mathrm{e}_s[k]$ denoted as $\tilde{\mathrm{e}}_s[k]$. We then subtract $\tilde{\mathrm{e}}_s[k]$ from $y^c[k]$ to obtain

$$\begin{aligned}\tilde{y}^c[k] &= y^c[k] - \tilde{\mathrm{e}}_s[k] \\ &= y^s[k] + \mathrm{e}_s[k] - \tilde{\mathrm{e}}_s[k].\end{aligned} \tag{3.15}$$

It is shown in Section 2.3.4 that for sufficiently large τ, it is possible to have 100% recovery rate of vector $\mathrm{e}_s[k]$ using the proposed estimator. We will illustrate this with simulations in the next section.

To proceed, we assume $\tilde{y}^c[k] = y^s[k]$. Substituting $\tilde{y}^c[k]$ into (3.3) and taking into account the conclusion of the discussion above, we obtain

$$\left.\begin{aligned} x[k+1] &= A_d x[k] + B_d(u^c[k] + \mathrm{e}_a[k]) \\ \tilde{y}^c[k] &= C_d x[k]. \end{aligned}\right\} \tag{3.16}$$

We next introduce the UIO design for the network control system (3.16).

3.4.1 UIO design

Recall that $\mathrm{e}_a[k]$ models unknown inputs originating from sparse malicious packet drops in (3.16). Similarly as in [31], we use the UIO to estimate the states of (3.16).

We first decompose the state $x[k]$ as

$$\begin{aligned} x[k] &= x[k] - MC_d x[k] + MC_d x[k] \\ &= (I - MC_d)x[k] + M\tilde{y}^c[k], \end{aligned} \tag{3.17}$$

where $M \in \mathbb{R}^{n\times p}$. Let $z[k] = (I - MC)x[k]$, we then have

$$\begin{aligned} z[k+1] &= (I - MC_d)x[k+1] \\ &= (I - MC_d)(A_d x[k] + B_d u^c[k] + B_d \mathrm{e}_a[k]) \\ &= (I - MC_d)(A_d x[k] + B_d u^c[k]) + (I - MC_d)B_d \mathrm{e}_a[k]. \end{aligned} \tag{3.18}$$

Substituting $x[k] = z[k] + M\tilde{y}^c[k]$ into (3.18), we obtain

$$\begin{aligned} z[k+1] \;=\; & (I - MC_d)(A_d z[k] + A_d M\tilde{y}^c[k] + B_d u^c[k]) \\ & +(I - MC_d)B_d \mathrm{e}_a[k]. \end{aligned} \tag{3.19}$$

Following [31], to improve the convergence rate, we add an extra term to the right-hand side of (3.19) to obtain

$$\begin{aligned} z[k+1] \;=\; & (I - MC_d)\Big(A_d z[k] + A_d M\tilde{y}^c[k] + B_d u^c[k] \\ & +L(\tilde{y}^c[k] - C_d z[k] - C_d M\tilde{y}^c[k])\Big) \\ & +(I - MC_d)B_d \mathrm{e}_a[k], \end{aligned} \tag{3.20}$$

where $L \in \mathbb{R}^{n\times p}$. The state estimate is

$$\hat{x}[k] = z[k] + M\tilde{y}^c[k]. \tag{3.21}$$

Let

$$\mathrm{e}[k] = x[k] - \hat{x}[k] \tag{3.22}$$

be the state estimation error. Using (3.16) , (3.20), and (3.21), we obtain

$$\begin{aligned} \mathrm{e}[k+1] \;=\; & x[k+1] - \hat{x}[k+1] \\ =\; & (I - MC_d)(A_d - LC_d)\mathrm{e}[k] \\ & +(I - MC_d)B_d \mathrm{e}_a[k]. \end{aligned} \tag{3.23}$$

If the condition $(I - MC_d)B_d = 0$ is satisfied, then

$$\mathrm{e}[k+1] = (I - MC_d)(A_d - LC_d)\mathrm{e}[k]. \tag{3.24}$$

Because $\mathrm{rank}(MC_dB_d) \leq \mathrm{rank}(C_dB_d) \leq \mathrm{rank}(B_d)$, if we want $(I - MC_d)B_d = 0$ to be satisfied, it is necessary to have

$$\mathrm{rank}(C_dB_d) = \mathrm{rank}\, B_d. \tag{3.25}$$

Combining (3.20), (3.21) with $(I - MC_d)B_d = 0$, we obtain

$$\left.\begin{array}{rcl} z[k+1] &=& (I - MC_d)\left(A_dz[k] + A_dM\tilde{y}^c[k] + B_du^c[k]\right) \\ && +L_1(\tilde{y}^c[k] - C_dz[k] - C_dM\tilde{y}^c[k]) \\ \hat{x}[k] &=& z[k] + M\tilde{y}^c[k], \end{array}\right\} \tag{3.26}$$

where $z[k] \in \mathbb{R}^n$, $\hat{x}[k] \in \mathbb{R}^n$ are the state of (3.26), and the state estimation of (3.16), respectively, $M \in \mathbb{R}^{n\times p}$, and $L_1 = (I - MC_d)L \in \mathbb{R}^{n\times p}$. It is shown in [26], [28] that (3.26) is defined as an UIO for the system (3.16) if the matrix $(I - MC_d)(A_d - LC_d)$ is Schur stable.

We now present and prove the following theorem.

Theorem 2. *Let $A_1 = (I - MC_d)A_d$ and $T = PL_1$. Suppose*

1. *$(I - MC_d)B_d = 0$,*
2. *there exists $P = P^\top \succ 0$ such that*

$$\begin{bmatrix} -P & A_1^\top P - C_d^\top T^\top \\ PA_1 - TC_d & -P \end{bmatrix} \prec 0. \tag{3.27}$$

Then the UIO given by (3.26) exists.

Proof. Substituting $(I-MC_d)B_d = 0$ into (3.23), we obtain $\mathrm{e}[k+1] = (A_1 - L_1C_d)\mathrm{e}[k]$, where $L_1 = (I - MC_d)L$. We need the matrix $A_1 - L_1C_d$ to be Schur stable, which is equivalent to the existence of $P = P^\top \succ 0$ such that

$$(A_1 - L_1C_d)^\top P(A_1 - L_1C_d) - P \prec 0.$$

Substituting $P = PP^{-1}P$ into the Lyapunov matrix inequality above, we obtain

$$(A_1 - L_1C_d)^\top PP^{-1}P(A_1 - L_1C_d) - P \prec 0,$$

which is equivalent to (3.27) by taking the Schur complement. □

Remark 6. *The first condition in Theorem 2 is a linear matrix equality in M, while the second condition is a linear matrix inequality in P and T. We solve the above matrix equality and matrix inequality using the* `cvx` *toolbox.*

3.4.2 Algorithm for combined approximator and UIO design

We summarize our discussion in the form of the algorithm for state observer design, depicted in Figure 3.2, in the presence of malicious packet drops between the plant output and controller input and between the controller output and the actuator input.

Observer design algorithm

1. Design an approximator $\tilde{\mathrm{e}}_s[k]$ of $\mathrm{e}_s[k]$ and use (3.15) to obtain (3.16)
2. Design the UIO given by (3.26) performing the following steps:
   - Check if $\mathrm{rank}(C_dB_d) = \mathrm{rank}\, B_d$ is satisfied. If not, STOP
   - Solve $(I - MC_d)B_d = 0$ to obtain

     $$M = B_d\left((C_dB_d)^\dagger + M_0(I_p - (C_dB_d)(C_dB_d)^\dagger)\right)$$

     , where the superscript † denotes the Moore-Penrose pseudo-inverse and M_0 is a design parameter matrix (see, for example [31])

- Solve (3.27) for matrices P and T
- If $P = P^\top \succ 0$, the UIO exists
- Calculate $L_1 = P^{-1}T$.

3.5 An Alternative Approach to Reconstruct Malicious Packet Drops During the Control Signal Transmission

In this section, we propose an alternative estimator of $\mathrm{e}_a[k]$. To proceed, we need the following lemma.

Lemma 6. *If* $\mathrm{rank}(C_dB_d) = \mathrm{rank}\, B_d = m$, *then there exists a matrix* $(C_dB_d)^\dagger \in \mathbb{R}^{m\times p}$ *such that* $(C_dB_d)^\dagger C_dB_d = I_m$.

Proof. Since C_dB_d is $p \times m$ and $\mathrm{rank}(C_dB_d) = \mathrm{rank}\, B_d = m$, C_dB_d has full column rank. It follows that $(C_dB_d)^\top C_dB_d$ is nonsingular. Then it is immediate that

$$(C_dB_d)^\dagger = \left((C_dB_d)^\top C_dB_d\right)^{-1}(C_dB_d)^\top.$$

□

We assume that condition (3.25) holds. We let $T_r = (C_dB_d)^\dagger$ for notational convenience. Premultiplying both sides of

$$x[k+1] = A_dx[k] + B_d(u^c[k] + \mathrm{e}_a[k])$$

by the matrix T_rC_d, we obtain

$$T_rC_dx[k+1] = T_rC_dA_dx[k] + T_rC_dB_du^c[k] + T_rC_dB_d\mathrm{e}_a[k]. \tag{3.28}$$

By Lemma 6, $T_rC_dB_d = I_m$. We can thus rewrite (3.28) as

$$\mathrm{e}_a[k] = T_ry[k+1] - T_rC_dA_dx[k] - u^c[k]. \tag{3.29}$$

Using the above equation, we propose the following estimator of $\mathrm{e}_a[k]$,

$$\tilde{\mathrm{e}}_a[k] = T_r y[k+1] - T_r C_d A_d \hat{x}[k] - u^c[k]. \tag{3.30}$$

Since the above estimate of $\mathrm{e}_a[k]$ depends on $y[k+1]$, we can estimate the unknown input with one sampling period time-delay as,

$$\tilde{\mathrm{e}}_a[k-1] = T_r y[k] - T_r C_d A_d \hat{x}[k-1] - u^c[k-1]. \tag{3.31}$$

3.6 The Robustness of the Alternative Unknown Input Estimator

The alternative unknown input estimator design can also be used to observe different types of unknown inputs $\mathrm{e}_a[k]$ with one sampling period time delay. We illustrate this on a linear DT plant model,

$$\left.\begin{aligned} x[k+1] &= A_d x[k] + B_{1d} u_1[k] + B_{2d} u_2[k] \\ y[k] &= C_d x[k] \end{aligned}\right\} \tag{3.32}$$

where $A_d \in \mathbb{R}^{n\times n}$, $B_{1d} \in \mathbb{R}^{n\times m_1}$, $B_{2d} \in \mathbb{R}^{n\times m_2}$ has full column rank, and $C_d \in \mathbb{R}^{p\times n}$. $u_1[k]$ is the control signal, $u_2[k]$ is the unknown input. We assume the plant model (3.32) is controllable and observable. We also assume $\operatorname{rank}(C_d B_{2d}) = \operatorname{rank} B_{1d}$.

Using the algorithm in Section 3.4.2, we design the UIO for (3.32) as

$$\begin{aligned} z[k+1] =& (I - MC_d)(A_d z[k] + A_d M y[k] + B_{1d} u_1[k]) \\ &+ L_1(y[k] - C_d z[k] - C_d M y[k]) \qquad (3.33a) \\ \hat{x}[k] =& z[k] + My[k]. \qquad (3.33b) \end{aligned}$$

We then apply the unknown input estimator design in Section 3.5 to obtain

$$\hat{u}_2[k-1] = T_r y[k] - T_r C_d A_d \hat{x}[k-1] - T_r C_d B_{1d} u_1[k-1]. \tag{3.34}$$

The robustness of the proposed unknown input estimator is tested with the following numerical example.

Let

$$A_d = 10^{-2} \times \begin{bmatrix} 99.0423 & 1.9495 & 0.9564 & 0.0064 \\ -94.9625 & 94.6692 & 94.7146 & 0.9564 \\ -0.3862 & -0.0025 & 99.6125 & 1.9974 \\ -38.5357 & -0.3805 & -38.7837 & 99.6125 \end{bmatrix},$$

$$B_{1d} = 10^{-2} \times \begin{bmatrix} 0.4253 \\ 42.1505 \\ -0.0003 \\ -0.0551 \end{bmatrix}, \; B_{2d} = 10^{-2} \times \begin{bmatrix} 66.2418 \\ -31.8204 \\ -0.0856 \\ -12.8333 \end{bmatrix}, \; C_d = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}.$$

Note that C_d is full row rank and the rank condition of our DT plant model is satisfied, that is, $\operatorname{rank}(C_d B_{2d}) = \operatorname{rank} B_{2d} = 1$. By solving the DT algerbraic Riccati equation, we obtain the feedback controller as

$$u_1[k] = -\begin{bmatrix} 2.2263 & 0.4538 & 2.6579 & -0.5152 \end{bmatrix} \hat{x}[k].$$

To proceed with our observer design (3.33), first, by solving for M so that $(I - MC_d)B_{2d} = 0$, we obtain,

$$M = \begin{bmatrix} 0.8125 & -0.3903 \\ -0.3903 & 0.1875 \\ -0.0011 & 0.0005 \\ -0.1574 & 0.0756 \end{bmatrix}.$$

We then solve the matrices P and L_1 by using the algorithm in Section 3.4.2, we obtain

$$P = \begin{bmatrix} 94.33051 & 18.0754 & -36.0293 & 0.1537 \\ 18.0754 & 96.8613 & -61.7687 & 0.2902 \\ -36.0293 & -61.7687 & 54.7671 & -0.3795 \\ 0.1537 & 0.2902 & -0.3795 & 0.0050 \end{bmatrix}, L_1 = \begin{bmatrix} 0.4317 & 1.3569 \\ 1.0290 & 2.7607 \\ 2.2369 & 3.5281 \\ 63.3664 & 85.5940 \end{bmatrix}.$$

In our simulation, we let $x[0] = [3\ 2\ 3\ -2]^\top$, zero initial condition for the state observer, $\hat{x}[-1] = [0\ 0\ 0\ 0]^\top$, and $u_1[-1] = 0$. A plot of the unknown input u_2 is shown on the top part of Figure 3.3. The unknown input estimate is shown on the bottom part of Figure 3.3.

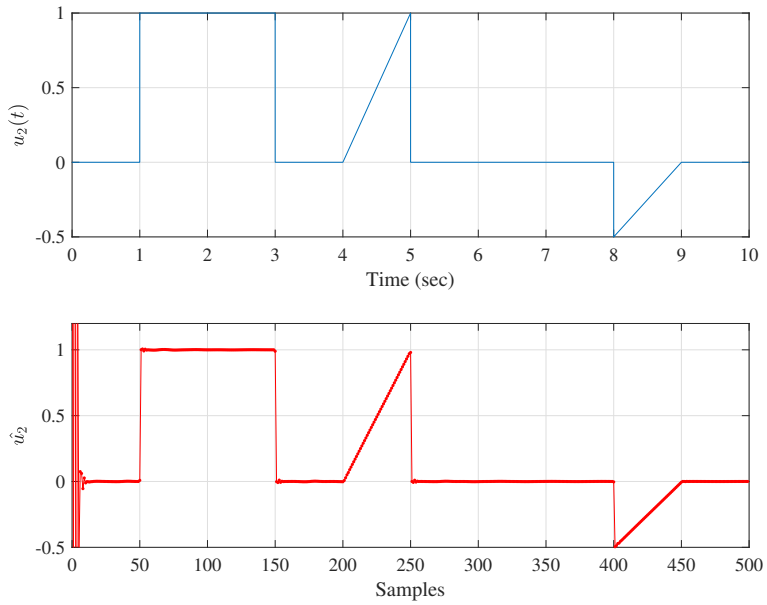


Figure 3.3. A plot of the unknown input signal and its estimate.

3.7 Numerical Example

In this section, we validate the proposed state observer design method on a numerical example. We consider a discrete-time state-space model of a coupled mass-spring-damper system, where

$$A = \begin{bmatrix} 0.9907 & 0.0047 & 0.0903 & 0.0002 \\ 0.0047 & 0.9907 & 0.0002 & 0.0903 \\ -0.1805 & 0.0900 & 0.8100 & 0.0044 \\ 0.0900 & -0.1805 & 0.0044 & 0.8100 \end{bmatrix}, B = \begin{bmatrix} 0 \\ 0.0047 \\ 0.0002 \\ 0.0903 \end{bmatrix}, C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}.$$

See [63, p. 148] for modeling equations of such a system. We assume that the plant model is remotely controlled as a networked control system as shown in Figure 3.1. Using the algorithm in Subsection 3.4.2, we first compute matrices M and L to obtain

$$M = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0.0033 \\ 0.0017 & 199.66 \end{bmatrix}, L_1 = \begin{bmatrix} 0.9936 & 0.0039 \\ -0.0008 & -0.0000 \\ -0.1551 & 0.0581 \\ -0.0159 & -19.3226 \end{bmatrix}.$$

We next compute the matrix P and find that $P = P^\top \succ 0$, which means that the necessary and sufficient conditions for the existence of the UIO are satisfied.

In our simulation, we let $\tau = 10$ so that the condition of Lemma 5 is satisfied, that is, $(\tau p - n) \geq 2(\mathrm{i}_s + \mathrm{i}_a)$. The control $u^c[k] = -K_d \hat{x}[k]$, where

$$K_d = \begin{bmatrix} 0.1381 & 0.2677 & 0.0651 & 0.3401 \end{bmatrix}$$

is the feedback gain calculated using the discrete-time LQR. We assume zero initial conditions on the plant input and its output, that is, $u^c[-1] = \cdots = u^c[1-\tau] = 0$ and $y^c[-1] = \cdots = y^c[1-\tau] = 0$. This explains the initial observation errors.

We simulated the case with 15% output transmission drops and 5% input transmission drops. From Figure 3.4, we see that the observer correctly estimates the plant's states. Fig-

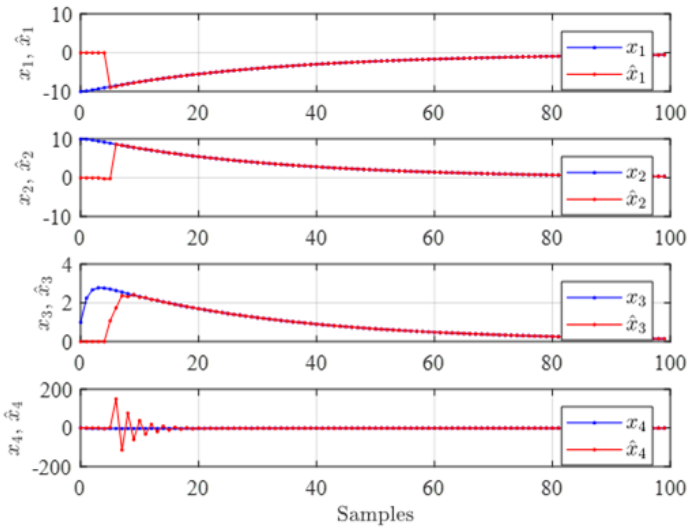


Figure 3.4. State estimates with 15% output transmission packet drops and 5% input transmission packet drops.

ure 3.5 shows the effectiveness of the proposed output transmission packet drops estimation. In Figure 3.6, the top plot shows the plots of the control signal and this signal corrupted by the malicious packet drops during the control signal transmission. The middle plot shows the plots of the malicious packet drops and their estimates using (3.13). The bottom plot shows the plots of the malicious packet drops and their estimates using the alternative $\mathrm{e}_a[k]$ estimator given by (3.31). Note that both estimates of $\mathrm{e}_a[k]$ are delayed by one sampling period.

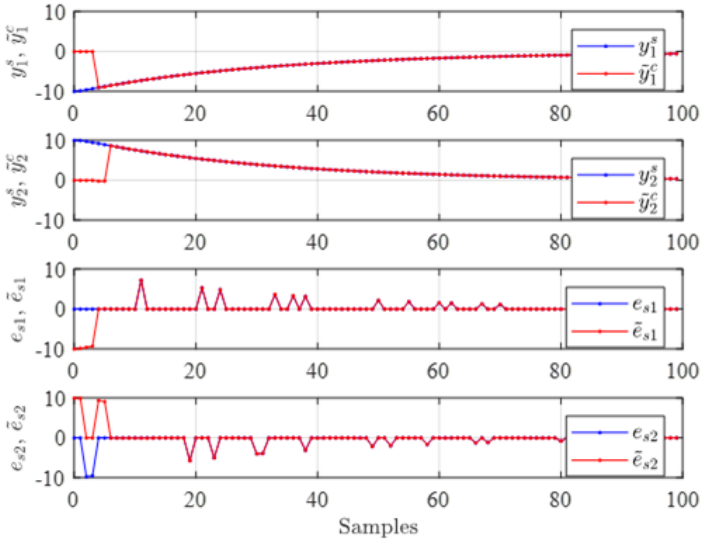


Figure 3.5. Plots of the output recovery and the output recovery errors with 15% output transmission packet drops and 5% input transmission packet drops.

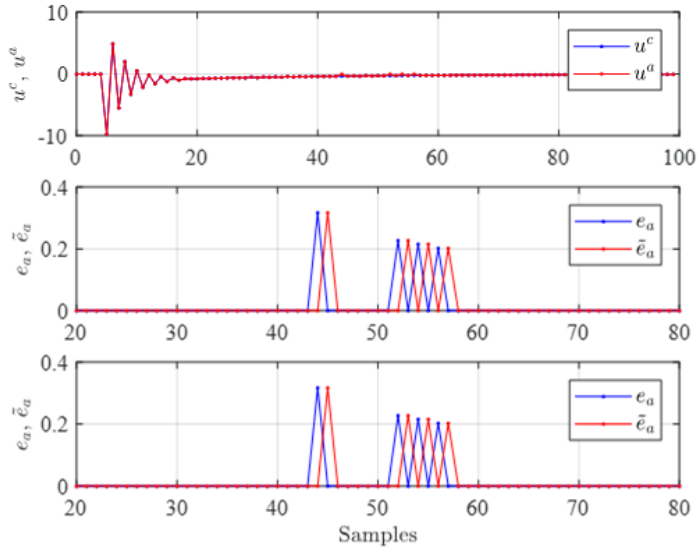


Figure 3.6. Top subfigure shows plots of the control signal generated by the controller and the control signal received by the plant. Middle plot shows the malicious packet drops $\mathrm{e}_a[k]$ and their estimates using the unknown input estimator given by (3.13). Bottom plot shows the malicious packet drops and their estimates using the alternative $\mathrm{e}_a[k]$ estimator given by (3.31).

3.8 Conclusions

A novel observer architecture for discrete-time systems subjected to sparse errors between the sensors and the controller and between the controller and the actuators is proposed. This novel observer consists of an approximator that recovers the unknown sparse errors between the sensors and the controller. This approximation is used to cancel the sparse error resulting in the plant output to the controller approximately equal to the actual plant output. Then the plant state estimate is obtained using a novel unknown input observer (UIO). The proposed observer can be used to construct a combined controller-observer compensator for a given networked system.

4. UNKNOWN INPUT OBSERVERS FOR DISCRETIZED SYSTEMS WITH APPLICATION TO CYBER-PHYSICAL SYSTEMS CORRUPTED BY SPARSE MALICIOUS PACKET DROPS

4.1 Introduction

The problem of designing observers for linear systems with both known and unknown inputs can be formulated as an UIO design problem. This problem was already studied by Basile and Marro [29] in 1969. Since then, different UIO structures have been reported in the literature, see for example [26], [31]. For example, UIOs for switched discrete-time (DT) systems for fault detection are reported in [60]. UIO designs for continuous-time (CT) systems are presented, for example, in [56], [57], [64]. UIO architectures for DT systems are given in [26], [65].

As it is discussed in Chapter 3, one of the conditions for the existence of a CT UIO is the matrix rank condition, $\text{rank}(C_c B_{2c}) = \text{rank}(B_{2c})$, where B_{2c} is the input matrix corresponding to the unknown input and the matrix C_c is the output measurement matrix of the plant modeled as, $\dot{x} = A_c x + B_{1c} u_1 + B_{2c} u_2$, $y = C_c x$. Our proposed UIO architectures use discretized plant parameters to design the UIO in the DT domain. We use the exact discretization method. The matrix rank condition for the discretized system is $\text{rank}(C_d B_{2d}) = \text{rank}(B_{2d})$, where $C_d = C_c$, and $B_{2d} = \int_0^{T_s} \mathrm{e}^{A_c \eta} B_{2c} d\eta$. In many cases, while the original CT plant does not satisfy the matrix rank condition for the existence of an UIO, its discretized model satisfies the matrix rank condition for the existence of a DT UIO. In this chapter, we characterize a class of systems showing this continuous-discrete UIO existence dichotomy.

4.2 Problem Statement

We consider a CT linear time-invariant (LTI) plant,

$$\left.\begin{aligned} \dot{x}(t) &= A_c x(t) + B_c u(t) \\ y(t) &= C_c x(t), \end{aligned}\right\} \tag{4.1}$$

where $A_c \in \mathbb{R}^{n\times n}$, $B_c \in \mathbb{R}^{n\times m}$, $C_c \in \mathbb{R}^{p\times n}$.

We consider the case when the CT plant is remotely controlled by a digital compensator. The CT plant and the digital compensator are interconnected through unsecured communication networks. A block diagram of a remotely controlled CT plant is depicted in Figure 4.1. The output measurement $y(t)$ of the plant is sampled and the sampled signal

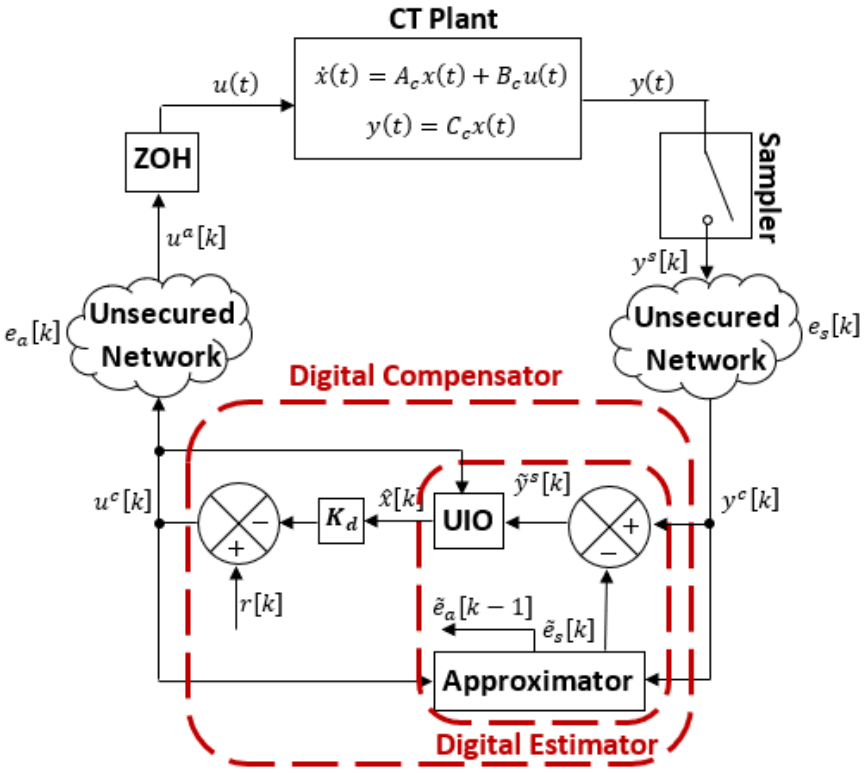


Figure 4.1. CPS considered in this chapter.

$y^s[k]$ is transmitted to the digital compensator. During the signal transmission, the unsecured network introduces unknown error $\mathrm{e}_s[k]$ to the sampled output measurement. The corrupted measurement $y^c[k]$ is received by the digital compensator. The compensator gen-

erates the control signal $u^c[k]$, which is then transmitted to the plant. During the control signal transmission, the unsecured network introduces unknown error $\mathrm{e}_a[k]$ to the control signal. The corrupted control signal $u^a[k]$ is passed through the Zero-Order-Hold (ZOH) element and then received by the actuators of the plant. Note that $r[k]$ in Figure 4.1 is the reference signal. When we consider the stabilization problem, we set $r[k] = 0$.

In this chapter, we employ the digital compensator that was developed in Section 3.4. The compensator contains two components: a digital estimator and a controller. The digital estimator, on the other hand, comprises the approximator of the sensor errors and the UIO that estimates the plant state. The recovered sensor error is used to cancel the sensor error introduced by the unsecured network during the transmission of the plant output signal to the compensator. In this paper, we focus our attention on the matrix rank condition for the existence of UIOs in the DT domain.

The design of the digital compensator is performed in the DT domain using the discretized plant model. A critical condition for the existence of an UIO is the matrix rank condition, $\mathrm{rank}(C_d B_d) = \mathrm{rank}(B_d)$. For many CT plants, the matrix rank condition fails in the CT domain while it is satisfied in the DT domain. Our objective is to study the matrix rank condition after exact discretization.

4.3 The Estimator Architecture Overview

In this section, we review the estimator architecture proposed in Section 3.4 and compare it with the one given in the literature [10], [15] to show the advantage of our proposed estimator.

The estimator design is performed using the discretized plant model. We use the exact discretization to obtain the following DT plant model,

$$\left.\begin{array}{rcl} x[k+1] &=& A_d x[k] + B_d u^a[k] \\ y^s[k] &=& C_d x[k], \end{array}\right\} \tag{4.2}$$

where T_s is a sampling period, $A_d = \mathrm{e}^{A_c T_s}$, $B_d = \int_0^{T_s} \mathrm{e}^{A_c \eta} B_c d\eta$, and $C_d = C_c$. See, for example [61, Subsection 1.1.2 and Chapter 2] or [62, Subsection 4.2.1] for discussions on modeling

and the properties of DT systems. The DT plant with corrupted output measurements and corrupted control signals is modeled as

$$\left.\begin{array}{rcl} x[k+1] &=& A_d x[k] + B_d\big(u^c[k] + \mathrm{e}_a[k]\big) \\ y^c[k] &=& C_d x[k] + \mathrm{e}_s[k], \end{array}\right\} \tag{4.3}$$

where $\mathrm{e}_a[k]$ and $\mathrm{e}_s[k]$ are sparse malicious packet drops of the form $\mathrm{e}_s[k] = y^c[k] - y^s[k] = (\Gamma[k]-I_p)y^s[k]$, $\mathrm{e}_a[k] = u^a[k]-u^c[k] = (\Lambda[k]-I_m)u^c[k]$, where $\Gamma[k] = \mathrm{diag}\{\gamma_1[k], \gamma_2[k], \ldots, \gamma_p[k]\}$, and $\Lambda[k] = \mathrm{diag}\{\lambda_1[k], \lambda_2[k], \ldots, \lambda_m[k]\}$, where $\gamma_{\mathrm{i}}[k], \mathrm{i} = 1, \ldots, p$, and $\lambda_{\mathrm{i}}[k], \mathrm{i} = 1, \ldots, m$, are Boolean variables, with 1 for packet received and 0 for packet dropped. For more detail on the above model, see [10], [15], [66]. To proceed, we collect τ output measurements and represent them in the form,

$$Y[k] = O^{\tau-1}x[k-\tau+1] + I_{\tau p}E_s[k] + F[k]\mathcal{V}[k], \tag{4.4}$$

where $Y[k] \in \mathbb{R}^{\tau p}$, $O^{\tau-1} \in \mathbb{R}^{\tau p \times n}$, $I_{\tau p} \in \mathbb{R}^{\tau p \times \tau p}$, and $F[k] \in \mathbb{R}^{\tau p \times (\tau-1)m}$ are known matrices; where $E_s[k] \in \mathbb{R}^{\tau p}$, and $\mathcal{V}[k] \in \mathbb{R}^{(\tau-1)m}$ are unknown vectors. Let $\Omega[k] = [I_{\tau p} \quad F[k]]$ and $E[k] = [E_s^\top[k] \quad \mathcal{V}^\top[k]]^\top$, then

$$Y[k] = O^{\tau-1}x[k-\tau+1] + \Omega[k]E[k]. \tag{4.5}$$

To proceed, we find the left annihilator $Q_2^\top$. We then premultiply both sides of (4.5) by $Q_2^\top$ to obtain $Z[k] = W[k]E[k]$, where $Z[k] = Q_2^\top Y[k]$, and $W[k] = Q_2^\top \Omega[k]$. We recover $E[k]$ as a solution of the 1-norm minimization problem,

$$\tilde{E}[k] = \min \|E[k]\|_1 \quad \text{subject to} \quad Z[k] = W[k]E[k], \tag{4.6}$$

where $\|\cdot\|_1$ denotes the 1-norm. Let $\tilde{\mathrm{e}}_s[k]$ and $\tilde{\mathrm{e}}_a[k]$ be the estimates of $\mathrm{e}_s[k]$ and $\mathrm{e}_a[k]$, respectively. We then have

$$\tilde{\mathrm{e}}_s[k] = \begin{bmatrix} \tilde{E}_1[k] & \cdots & \tilde{E}_p[k] \end{bmatrix}^\top \tag{4.7a}$$

$$\tilde{\mathrm{e}}_a[k-1] = \mathrm{diag}\{\tilde{E}_{\tau p+1}[k] \quad \cdots \quad \tilde{E}_{\tau p+m}[k]\} u^c[k-1]. \tag{4.7b}$$

Let $f_s[k] = \mathrm{e}_s[k] - \tilde{\mathrm{e}}_s[k]$. The recovered output measurement is $\tilde{y}^s[k] = y^c[k] - \tilde{\mathrm{e}}_s[k]$. Then the DT plant model takes the form

$$\left.\begin{aligned} x[k+1] &= A_d x[k] + B_d(u^c[k] + \mathrm{e}_a[k]) \\ \tilde{y}^s[k] &= C_d x[k] + f_s[k]. \end{aligned}\right\} \tag{4.8}$$

It is shown in [12], [22], [66] that sparse errors $\mathrm{e}_s[k]$ and $\mathrm{e}_a[k]$ can be almost correctly recovered. Based on this, we assume for our purposes that $f_s[k] = 0$. Then, the output of (4.8) takes the form $\tilde{y}^s[k] = C_d x[k]$, and the resulting system can be considered as a system with unknown input only, for which we construct the UIO given by (3.26).

4.4 Analysis of the Matrix Rank Condition

In this section, we analyze the case when the matrix rank condition is satisfied by a discretized model, that is, we study the conditions under which the following equality holds:

$$\mathrm{rank}\left(\int_0^{T_s} C_c \mathrm{e}^{A_c\eta} B_c d\eta\right) = \mathrm{rank}\left(\int_0^{T_s} \mathrm{e}^{A_c\eta} B_c d\eta\right).$$

This question arises naturally in the study of the rank condition for the existence of an UIO in the DT domain. Our main results show that discretization does not negatively impact the construction of an UIO (Section 4.4.1) and in fact can help in some cases (Section 4.4.2 and Section 4.4.3).

4.4.1 The Discretization Theorem

In this subsection, we show that if the continuous system satisfies the matrix rank condition $\text{rank}(C_c B_c) = \text{rank}(B_c) = m$, then the matrix rank condition is guaranteed to be satisfied by the discretized system given by (4.2) for almost all sampling periods $T_s > 0$ (see Theorem 3 for the exact statement). The proof is a little long and, to help the reader see the overall outline of the proof, it is presented as a series of lemmas.

In our analysis, we use standard Euclidean vector norm and the induced matrix operator norm.

Lemma 7. *Let $G \in \mathbb{R}^{n\times m}$ be a full column rank matrix. Then there exist $\delta > 0$ and $\|x_0\| = 1$ such that,*

$$\delta = \|Gx_0\| = \min_{\|x\|=1} \|Gx\|.$$

Proof. The results follows from the continuity of the vector norm and the theorem of Weierstrass. □

Remark 7. *Note that if $G \in \mathbb{R}^{n\times n}$ is a full column rank matrix, then all its singular values are positive and we can take $\delta = \sigma_m(G)$, where $\sigma_m(G)$ is the smallest singular value of G. Thus, for all $\|x\| = 1$, we have $\delta \leq \|Gx\|$.*

Lemma 8. *Let $G \in \mathbb{R}^{n\times m}$ be a full column rank matrix and let $\delta = \sigma_m(G) > 0$. If $H \in \mathbb{R}^{n\times m}$ is such that $\|H - G\| < \delta$, then H has full column rank.*

Proof. By Lemma 7, we have $\delta = \sigma_m(G) > 0$. For any $x \in \mathbb{R}^m$ such that $\|x\| = 1$, if $\|H - G\| < \delta$, then by the triangle inequality,

$$\begin{aligned}
\|Hx\| &= \|Gx + (H - G)x\| \\
&\geq \|Gx\| - \|(H - G)x\| \\
&> \delta - \delta = 0,
\end{aligned}$$

for all $\|x\| = 1$. Therefore the columns of H are linearly independent and so $\text{rank}(H) = m$. □

To proceed, we define for $T_s \geq 0$, $Q(T_s) = \int_0^{T_s} C_c \mathrm{e}^{A_c \eta} B_c \, d\eta$.

Lemma 9. *The entries in the matrix function $Q(T_s)$ are analytic functions of T_s on $\mathbb{R}$.*

Proof. We have $\mathrm{e}^{A_c \eta} = \sum_{k=0}^{\infty} \frac{\eta^k}{k!} A_c^k$ and therefore each entry of the matrix $\mathrm{e}^{A_c \eta}$ is a power series that converges for each $\eta \in \mathbb{R}$. It follows that each entry of the matrix $\mathrm{e}^{A_c \eta}$ is analytic on $\mathbb{R}$. Since the entries of $C_c \mathrm{e}^{A_c \eta} B_c$ are linear combinations of the entries of $\mathrm{e}^{A_c \eta}$, they are also analytic. The claim now follows from the fact that an integral of an analytic function is analytic. □

Lemma 10. *If $C_c B_c$ has full rank, then there exists $\varepsilon > 0$ such that $Q(T_s)$ has full rank for all $0 < T_s < \varepsilon$.*

Proof. We have

$$\begin{aligned} Q(T_s) &= \int_0^{T_s} C_c \mathrm{e}^{A_c \eta} B_c \, d\eta \\ &= C_c \left(\int_0^{T_s} \left(I + \eta A_c + \frac{\eta^2}{2!} A_c^2 + \cdots \right) d\eta \right) B_c \\ &= C_c \left(T_s I + T_s^2 R(T_s) \right) B_c, \end{aligned}$$

where $R(T_s) = \int_0^{T_s} \left(\eta A_c + \frac{\eta^2}{2!} A_c^2 + \cdots \right) d\eta$. By Lemma 9, $R(T_s) \in \mathbb{R}^{n \times n}$ is a matrix with analytic entries. It follows that $R(T_s)$ is bounded on bounded subsets of $\mathbb{R}$. So there exists $\mu > 0$ such that $\|R(T_s)\| < \mu$ for $0 \leq T_s \leq 1$. It follows that for $0 < T_s < 1$, we have $\left\| \frac{Q(T_s)}{T_s} - C_c B_c \right\| = \|T_s C_c R(T_s) B_c\| \leq T_s \|C_c\| \|R(T_s)\| \|B_c\| < T_s \mu \|C_c\| \|B_c\|$. By hypothesis, the product $C_c B_c \in \mathbb{R}^{p \times m}$ is full column rank, that is, $\operatorname{rank}(C_c B_c) = m$. Let $\delta = \sigma_m(C_c B_c) > 0$. By Lemma 8, if $Q(T_s)/T_s$ is such that $\|Q(T_s)/T_s - C_c B_c\| < \delta$, then $\operatorname{rank}(Q(T_s)/T_s) = m$. If we let $\varepsilon = \min\{1, \delta/(\mu \|C_c\| \|B_c\|)\}$, then $Q(T_s)/T_s$ has full column rank for $0 < T_s < \varepsilon$. To complete the proof, we only need to note that $Q(T_s)/T_s$ and $Q(T_s)$ have the same rank for $T_s > 0$. □

Lemma 11. *Suppose $Q(t_0)$ has full column rank for some fixed $t_0 > 0$. Then for all $T_s > 0$, $Q(T_s)$ has full column rank except for possibly countably many isolated T_s.*

Proof. First observe that $\det(Q(T_s)^\top Q(T_s))$ is an analytic function of T_s and therefore it is either identically zero or its zeros are isolated. Now suppose $Q(t_0)$ has full column rank.

Then $Q(t_0)u \neq 0$ for all nonzero $u \in \mathbb{R}^m$. It follows that $Q(t_0)^\top Q(t_0)$ is nonsingular and therefore $\det(Q(t_0)^\top Q(t_0)) \neq 0$. The zeros of $\det(Q(T_s)^\top Q(T_s))$ are isolated, which is clearly equivalent to the fact that $Q(T_s)$ has full column rank except for isolated points. □

Theorem 3. *If C_cB_c has full rank, then $Q(T_s)$ has full rank for all $T_s > 0$ except for countably many isolated T_s. That is, if rank(C_cB_c) = rank$(B_c) = m$, then the matrix rank condition is satisfied by the discretized system given by (4.2) for all $T_s > 0$ except for countably many isolated T_s.*

Proof. This is an immediate consequence of Lemma 10 and Lemma 11. □

Corollary 1. *If B_c is full column rank, then $\int_0^{T_s} \mathrm{e}^{A_c\eta}B_c\,d\eta$ is full column rank for all $T_s > 0$ except for countably many isolated T_s.*

4.4.2 Single-input systems

In this subsection, we show that if a nontrivial CT single-input system is controllable, then its exact discretization almost always satisfies the matrix rank condition. More precisely, we consider the case when B_c is a column vector b and we show that if the pair (A_c, b) is controllable, then $\int_0^{T_s} C_c\mathrm{e}^{A_c\eta}b\,d\eta$ is not identically zero for all nonzero C_c except for countably many isolated T_s.

Lemma 12. *Let the pair (A_c, B_c) be controllable. Suppose $0 \leq t_1 < t_2 < \infty$. Then*

$$\mathit{Span}\left\{\int_0^{T_s} \mathrm{e}^{A_c\eta}B_c\,d\eta \,:\, T_s \in (t_1, t_2)\right\} = \mathbb{R}^n.$$

Here "Span" denotes the span of the collection of columns.

Proof. Let

$$S = \mathrm{Span}\left\{\int_0^{T_s} \mathrm{e}^{A_c\eta}B_c\,d\eta : T_s \in (t_1, t_2)\right\}.$$

For each column b_i, $\mathrm{i} = 1, \ldots, m$, of the matrix B_c and for each $T_s \in (t_0, t_1)$, let $q_{b_\mathrm{i}}(T_s) = \int_0^{T_s} \mathrm{e}^{A_c\eta}b_\mathrm{i}\,d\eta$. Since finite dimensional subspaces are always closed and $\mathbb{R}^n$ is complete, therefore S is complete. Hence, the derivatives $q_{b_\mathrm{i}}(T_s), q_{b_\mathrm{i}}(T_s), \ldots$ are in S for each fixed

$T_s \in (t_1, t_2)$. Then for $k = 1, 2, \ldots$, we have $q_{b_\mathrm{i}}^{(k)}(T_s) = \mathrm{e}^{A_c T_s} A_c^{k-1} b_\mathrm{i}$. Since $\mathrm{e}^{A_c T_s}$ is nonsingular for each T_s, we conclude that

$$S \supset \operatorname{Span}\left\{q_{b_\mathrm{i}}^{(k)}(T_s) \,:\, k = 1, \ldots, n\right\} = \operatorname{Span}\left\{b_\mathrm{i}, \ldots, A_c^{n-1} b_\mathrm{i}\right\}$$

for each column b_i of B_c. It follows that

$$S \supset \operatorname{Span}\left\{B_c, A_c B_c, \ldots, A_c^{n-1} B_c\right\}.$$

Since (A_c, B_c) is controllable, we must have $S = \mathbb{R}^n$. □

Lemma 13. *Let (A_c, b) be controllable and let $0 \le t_1 < t_2 < \infty$. Then there exists $T_s \in (t_1, t_2)$ such that $\int_0^{T_s} C_c \mathrm{e}^{A_c \eta} b \, d\eta \neq 0$ for nonzero C_c. In particular, rank$(\int_0^{T_s} C_c \mathrm{e}^{A_c \eta} b \, d\eta)$ = rank$(\int_0^{T_s} \mathrm{e}^{A_c \eta} b \, d\eta)$.*

Proof. We have by Lemma 12 that $\operatorname{Span}\left\{\int_0^{T_s} \mathrm{e}^{A_c \eta} b \, d\eta \,:\, T_s \in (t_1, t_2)\right\} = \mathbb{R}^n$. Therefore, if $\int_0^{T_s} C_c \mathrm{e}^{A_c \eta} b \, d\eta = 0$ for all $T_s \in (t_1, t_2)$, then $C_c v = 0$ for all $v \in \mathbb{R}^n$ and thus $C_c = 0$. It follows that if C_c is not identically zero, then there must be $T_s \in (t_1, t_2)$ such that $\int_0^{T_s} C_c \mathrm{e}^{A_c \eta} b \, d\eta \neq 0$. □

We now present the main result of this subsection.

Theorem 4. *Let (A_c, b) be a controllable pair. Then $\int_0^{T_s} C_c \mathrm{e}^{A_c \eta} b \, d\eta \neq 0$ for all $T_s > 0$ except for countably many isolated T_s.*

Proof. Note that $\int_0^{T_s} C_c \mathrm{e}^{A_c \eta} b \, d\eta$ is an analytic function in T_s. By Lemma 13, there are points where it is not zero. The theorem now follows from the fact that if an analytic function is not identically zero, then its zeros are isolated, see [67, page 240]. □

Remark 8. *Isolated in Theorem 4 means that if $\int_0^{t_0} C_c \mathrm{e}^{A_c \eta} b d\eta = 0$ for $t_0 > 0$, then there exists $\varepsilon > 0$ such that $\int_0^{t} C_c \mathrm{e}^{A_c \eta} b d\eta \neq 0$ for all t such that $0 < |t - t_0| < \varepsilon$.*

We conclude from Theorem 4 that for the plant given by (4.1), if the pair (A_c, B_c) is controllable and B_c is a column vector, then the matrix rank condition is almost always satisfied by the discretized system given by (4.2).

4.4.3 Systems where the matrix rank condition not satisfied in the CT domain but it is satisfied in the DT domain

The following numerical example motivates our discussion in this subsection.

Example 2. We consider a CT state-space model from [63], where

$$x = \begin{bmatrix} q_1 \\ q_2 \\ \dot{q}_1 \\ \dot{q}_2 \end{bmatrix}, A_c = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ \frac{-2k}{m} & \frac{k}{m} & \frac{-c}{m} & 0 \\ \frac{k}{m} & \frac{-2k}{m} & 0 & \frac{-c}{m} \end{bmatrix}, B_c = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \frac{k}{m} \end{bmatrix}, C_c = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}.$$

Let $m = 5$ kg, $k = 5$ N/m, and $c = 10$ N·s/m. Then we have

$$A_c = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -2 & 1 & -1 & 0 \\ 1 & -2 & 0 & -1 \end{bmatrix}, \; B_c = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}.$$

By Theorem 4, a discretization using almost any sampling time would give the matrix rank condition. For example, if we discretize the above system with the sampling time $T_s = 0.1$ sec, then

$$A_d = \begin{bmatrix} 0.9907 & 0.0047 & 0.0903 & 0.0002 \\ 0.0047 & 0.9907 & 0.0002 & 0.0903 \\ -0.1805 & 0.0900 & 0.8100 & 0.0044 \\ 0.0900 & -0.1805 & 0.0044 & 0.8100 \end{bmatrix},$$

$B_d = \begin{bmatrix} 0 & 0.0047 & 0.0002 & 0.0903 \end{bmatrix}^\top$, and $C_d = C_c$, where $\text{rank}(C_c B_c) \neq \text{rank}(B_c)$ but $\text{rank}(C_d B_d) = \text{rank}(B_d)$.

We now characterize a class of systems for which the matrix rank condition does not hold in the CT domain but it is satisfied in the DT domain as illustrated by the above example. We consider a CT LTI plant modeled by (4.1). We make the following assumptions:

Assumption 1. *rank*$(A_c) = n$, *rank*$(B_c) = m$, *and rank*$(C_c) = p$, *where* $m \leq p \leq n$.

Assumption 2. *The matrices B_c and C_c have the form,*

$$B_c = \begin{bmatrix} O_{p\times m} \\ B^* \end{bmatrix}, \quad C_c = \begin{bmatrix} I_p & O_{p\times(n-p)} \end{bmatrix}.$$

Assumption 3. *Let $\mathcal{A} = A_c^{-1}(A_d - I_n)$, where $A_d = \mathrm{e}^{A_c T_s}$. Let $\mathcal{A}_1 \in \mathbb{R}^{p\times n}$ be the sub-matrix of $\mathcal{A}$ such that $\mathcal{A} = \begin{bmatrix} \mathcal{A}_1{}^\top & \mathcal{A}^{*\top} \end{bmatrix}^\top$ and rank$(\mathcal{A}_1 B_c) = m$.*

Note that $C_c B_c = O$ and hence $\mathrm{rank}(C_c B_c) \neq \mathrm{rank}(B_c)$, that is, the matrix rank condition for the existence of an UIO for the above CT plant. We will show that if the above assumptions are satisfied, then the matrix rank condition for the DT plant given by (4.2) is satisfied.

Theorem 5. *If Assumptions 1, 2, and 3 are satisfied, then the matrix rank condition for the DT plant given by (4.2) is satisfied.*

Proof. From Assumption 2, we have

$$C_c B_c = \begin{bmatrix} I_p & O_{p\times(n-p)} \end{bmatrix} \begin{bmatrix} O_{p\times m} \\ B^* \end{bmatrix} = O.$$

Thus, $\mathrm{rank}(C_c B_c) = 0$, but $\mathrm{rank}(B_c) = m$. Therefore, $\mathrm{rank}(C_c B_c) \neq \mathrm{rank}(B_c)$. We also have $A_d = \mathrm{e}^{A_c T_s}$, and $B_d = \int_0^{T_s} \mathrm{e}^{A_c \eta} B_c d\eta$. The condition, $\mathrm{rank}(A_c) = n$ implies that A_c is invertible. Therefore, $B_d = A_c^{-1}(A_d - I_n)B_c = \mathcal{A}B_c$. Since, by Assumption 3, $\mathrm{rank}(\mathcal{A}B_c) = m$, we have

$$C_d B_d = C_d \mathcal{A} B_c = \begin{bmatrix} I_p & O_{p\times(n-p)} \end{bmatrix} \begin{bmatrix} \mathcal{A}_1 B_c \\ \mathcal{A}^* B_c \end{bmatrix} = \mathcal{A}_1 B_c.$$

Hence, $\mathrm{rank}(C_d B_d) = \mathrm{rank}(B_d) = m$. □

Remark 9. *Note that rank$(C_d B_d)$ = rank$(B_d) = m$ implies that $p \geq m$ which is a part of Assumption 1.*

4.5 Example

We consider an NCS shown in Figure 4.1 with the nonlinear CT model of the inverted pendulum on a cart as a plant. We assume the presence of the unknown sparse input and output errors caused by malicious packet drops during output measurements and control signal transmissions. In our design, the nonlinear plant model is linearized about an equilibrium point of interest. Then, the linearized DT model is used to synthesize the proposed estimator employing a vector recovery method and an UIO. The state estimate from the proposed state estimator is then fed into the controller designed using the discrete LQR method. In the observer design, the linearized model of the inverted pendulum on a cart is used, however, in the simulations, the nonlinear model is employed. The nonlinear modelling equations are,

$$\left.\begin{aligned}(m_1+m_2)\ddot{x}+F_x\dot{x}+m_2l(\ddot{\theta}\cos\theta-\dot{\theta}^2\sin\theta)-u &= 0\\ J\ddot{\theta}+F_\theta\dot{\theta}-m_2lg\sin\theta+m_2l\ddot{x}\cos\theta &= 0\end{aligned}\right\} \tag{4.9}$$

where the model parameters are given in [55].

As in [58], the state vector is $x = [x \quad \theta \quad \dot{x} \quad \dot{\theta}]^\top$. We choose the output $y^s \in \mathbb{R}^2$ as $y^s = [x \quad \theta]^\top$. Linearizing the system model about the origin, we obtain the linear CT system model of the form (4.1), where

$$A_c = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & -1.9333 & -1.9872 & 0.0091 \\ 0 & 36.9771 & 6.2589 & -0.1738 \end{bmatrix},$$

$$B_c = \begin{bmatrix} 0 & 0 & 0.3205 & -1.0095 \end{bmatrix}^\top, C_c = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}.$$

The matrix rank condition for the above CT system does not hold. Therefore, a CT UIO for this system cannot be designed. To remedy this situation, the authors of [58] and [55] added one more sensor, which is equivalent to selecting $C_c = [I_3 \quad O_{3\times 1}]$. Our method does not need this additional sensor because by Theorem 4, a discretization using almost any sampling time would give the matrix rank condition. To demonstrate this, we first discretize

the above CT system with sampling time period $T_s = 0.2$ sec using the exact discretization method to obtain

$$A_d = \begin{bmatrix} 1 & -0.0381 & 0.1644 & -0.0023 \\ 0 & 1.8088 & 0.1235 & 0.2480 \\ 0 & -0.4037 & 0.6588 & -0.0362 \\ 0 & 8.9315 & 1.3069 & 1.7668 \end{bmatrix}, \; B_d = \begin{bmatrix} 0.0057 \\ -0.0199 \\ 0.0550 \\ -0.2108 \end{bmatrix}.$$

The matrix rank condition is satisfied by the discretized system model for $C_d = C_c = [I_2 \; O_{2\times 2}]$, that is, $\text{rank}(C_d B_d) = \text{rank}(B_d) = 1$. We next compute

$$M = \begin{bmatrix} 0.0768 & -0.2663 \\ -0.2663 & 0.9232 \\ 0.7360 & -2.5512 \\ -2.8192 & 9.7723 \end{bmatrix}, \; L = \begin{bmatrix} 1.0182 & 0.4739 \\ 0.2937 & 0.1367 \\ -0.2519 & 4.3787 \\ 2.9786 & -8.8061 \end{bmatrix},$$

$$K_d = \begin{bmatrix} -0.3027 & -62.3770 & -8.4185 & -10.3468 \end{bmatrix}.$$

We choose $\tau = 10$, 3% packet drops during control signal transmission, and 10% packet drops during measurements transmission. It can be seen in Figure 4.2 that the combined controller-observer compensator stabilizes the nonlinear plant about the equilibrium of interest and the observer estimates the states with sufficient accuracy. Figure 4.3 shows the plant outputs corrupted by the sensor measurement noise successfully recovered. In the top subfigure of Figure 4.4, the DT control signal sent by the controller is compared with the control signal received by the actuator. The bottom subfigure shows the control signal after passing through the ZOH.

4.6 Conclusions

In this chapter, we show that if a CT system satisfies the matrix rank condition, then the exact discretization of it will (with the possible exception of a countable set of sampling times) satisfy the matrix rank condition. We show that for a controllable single-input system, the exact discretization of it will (with the possible exception of a countable set of sampling

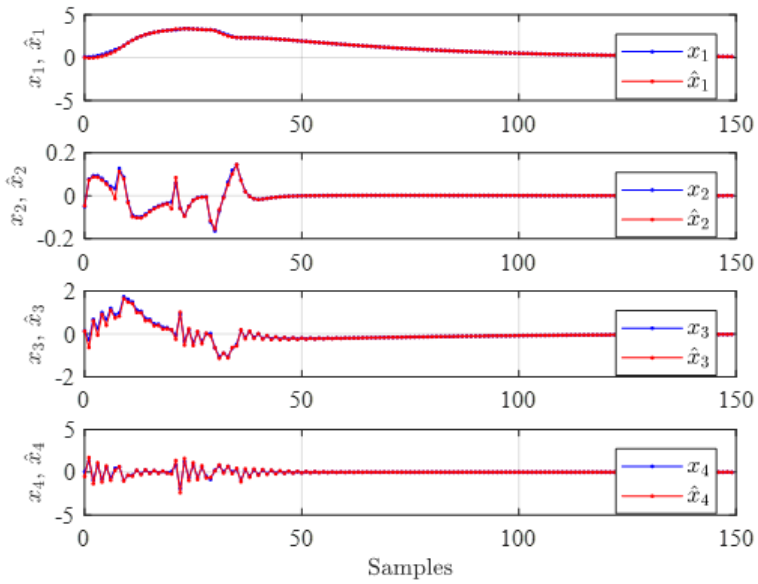


Figure 4.2. State estimates of the nonlinear plant.

times) satisfy the matrix rank condition, regardless of whether it holds for the original CT system. We characterize a class of systems for which the matrix rank condition fails in the CT domain but holds in the DT domain. We test our proposed estimators on a CT nonlinear model of the inverted pendulum on a cart corrupted by unknown input and output errors. The CT linearized inverted pendulum on a cart model does not satisfy the matrix rank condition for the existence of an UIO but its discretized model does satisfy the discrete matrix rank condition.

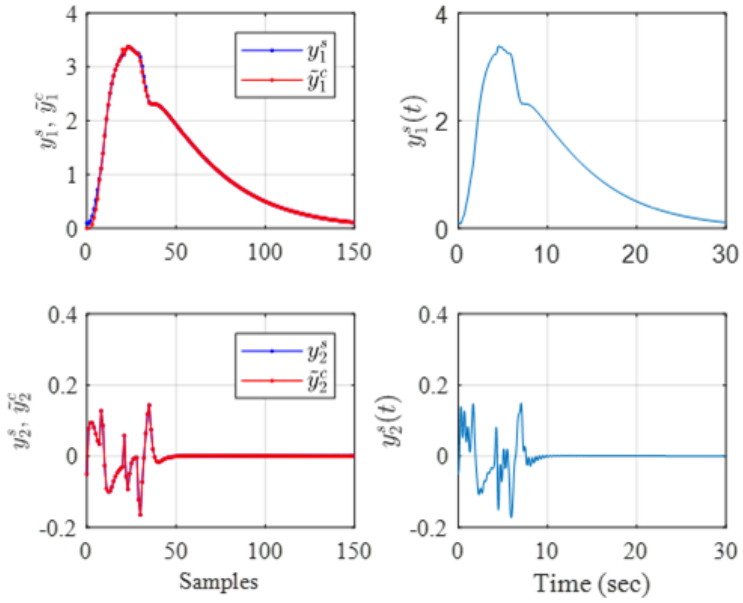


Figure 4.3. Left subfigures show the output estimates of the plant generated by the digital estimator. Right subfigures show the outputs of the nonlinear CT system.

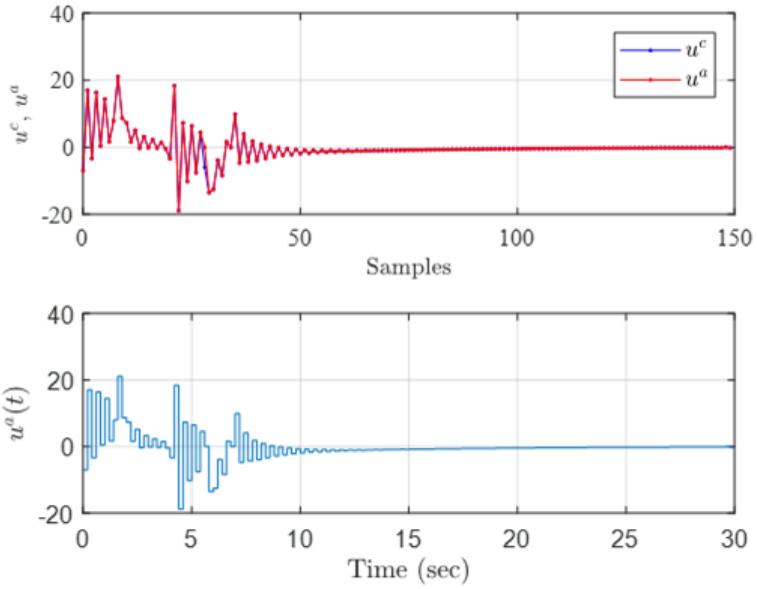


Figure 4.4. Top subfigure shows plots of the control signal generated by the controller and the control signal received by the plant. Bottom plot shows the control signal received by the nonlinear plant.

5. OBSERVERS FOR CYBER-PHYSICAL SYSTEMS WITH UNSECURED COMMUNICATION NETWORKS AND SUBJECTED TO DISTURBANCE

5.1 Introduction

In this chapter, the objective is to simultaneously estimate the state, communication errors, and unknown disturbance in a CPS with unsecured communication networks and subjected to disturbance. We assume that the communication errors are sparse and the unknown disturbance is arbitrary.

We perform a comparative study of the norm-based observer given in Chapter 2 and the combined norm-UIO based observer given in Chapter 3, when the cyber-physical system is subjected to sparse errors in the communication channels and arbitrary disturbance. We perform converges analysis of the estimation error for the closed-loop CPS driven by the combined controller-observer compensator. We propose a novel design method for the norm-based observer and the combined norm-UIO based observer using fictitious output measurements to improve their performance.

In our analysis, we use the following definitions. The set of natural numbers is denoted $\mathbb{N}$. The 0-*norm* of a finite dimensional vector v, denoted $\|v\|_0$, is the number of nonzero elements in v. A vector v is i_v-*sparse* if it has at most i_v nonzero components, that is, $\|v\|_0 \leq \mathrm{i}_v$. A vector $v \in \mathbb{R}^n$ is *sparse* if $\|v\|_0 < \frac{n}{2}$. The *sparsity* of the vector v is $s_v = 1 - \frac{\|v\|_0}{n}$, which is the ratio of number of zero elements in v by the number of elements in v. The *spark* of a matrix F, denoted spark(F), is the smallest number of linearly dependent columns of F. We will also use the standard vector p-norms defined by $\|x\|_p = (|x_1|^p + \cdots + |x_n|^p)^{1/p}$, $1 \leq p < \infty$. For any matrix G, its p-*norm* is: $\|G\|_p = \sup_{\|x\|_p=1} \|Gx\|_p$. For a sequence of vectors $f[0], f[1], \ldots$, its ℓ_∞-*norm* is $\|\{f[k]\}_{k=0}^{\infty}\|_\infty = \sup_{k\geq 0} \|f[k]\|_2$. By convention, when the norm of a vector or a matrix is used without any subscript p, it will mean the 2-norm. A sequence of scalars $\{a_q\} \in \ell_1$ if $\|\{a_q\}\|_1 = \sum_{q=0}^{\infty} |a_q| < \infty$.

5.2 Problem Statement

We consider a DT plant modeled by

$$\left.\begin{aligned} x[k+1] &= Ax[k] + B_1 u^a[k] + B_d d[k] \\ y^s[k] &= Cx[k], \end{aligned}\right\} \tag{5.1}$$

where $A \in \mathbb{R}^{n\times n}$, $B_1 \in \mathbb{R}^{n\times m}$, $B_d \in \mathbb{R}^{n\times n_d}$, and $C \in \mathbb{R}^{p\times n}$. The vectors $x[k] \in \mathbb{R}^n$, $u[k] \in \mathbb{R}^m$, and $d[k] \in \mathbb{R}^{n_d}$ are the state, the control input, and unknown disturbance actin on the system, respectively. See, for example, [61, Chapter 2] or [62] for the modeling and analysis of DT systems. We assume that the pair (A, B_1) is reachable and the pair (A, C) is observable. We consider the case when the plant is remotely controlled via unsecured communication networks. A block diagram of such a CPS architecture is depicted in Figure 5.1. The output

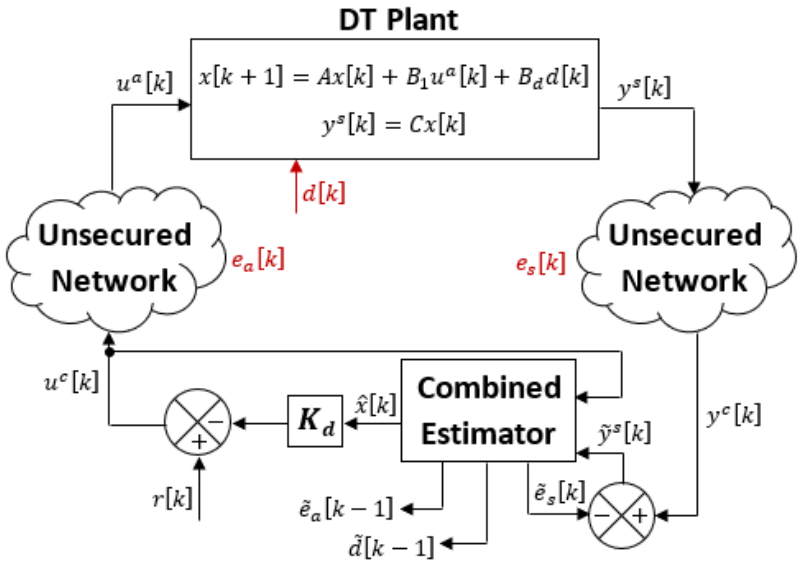


Figure 5.1. The CPS architecture considered in this paper.

signals of the plant are measured by the sensors and transmitted to the combined controller-estimator. In its passage through the unsecured network, the output signal is corrupted by the unknown error $\mathrm{e}_s[k]$. Thus the signal received by the estimator is $y^c[k] = y^s[k] + \mathrm{e}_s[k]$. In a similar way, the control signal $u^c[k]$ from the controller to the plant is corrupted by the

unknown error $\mathrm{e}_a[k]$ during its transmission causing the plant to receive the signal $u^a[k] = u^c[k] + \mathrm{e}_a[k]$ instead. The model of the CPS can thus be represented as

$$\left.\begin{aligned} x[k+1] &= Ax[k] + B_1(u^c[k] + \mathrm{e}_a[k]) + B_d d[k] \\ y^c[k] &= Cx[k] + \mathrm{e}_s[k]. \end{aligned}\right\} \tag{5.2}$$

We assume that the communication errors $\mathrm{e}_a[k]$ and $\mathrm{e}_s[k]$ are sparse, and the disturbance $d[k]$ is arbitrary.

In this chapter, our objective is to compare the performance of two types of observers for such CPSs. The first type of the observer is norm-based and the second type is combined norm-UIO based observer. We then propose a fictitious output design method to improve the observers' performance.

5.3 Application of the 1-Norm Approximation Based Observer

In this section, we apply the 1-norm approximation based observer given in Chapter 2 to a CPS with sparse communication errors and subjected to arbitrary disturbance. We first use the system model (5.2) to derive modeling equation, which expands the system into a form amenable to the 1-norm approximation method. The key to the derivation of this equation is the accumulation of output measurements. The equation is then used to estimate the sparse errors and the disturbance.

5.3.1 Accumulation of CPS measurements

Let $B_2 = \begin{bmatrix} B_d & B_1 \end{bmatrix}$, $u_1[k] = u^c[k]$, and $u_2[k] = \begin{bmatrix} d[k]^\top & \mathrm{e}_a[k]^\top \end{bmatrix}^\top$, where $B_2 \in \mathbb{R}^{n\times(n_d+m)}$, and $u_2[k] \in \mathbb{R}^{n_d+m}$. Then we represent the plant model given by (5.2) as

$$\left.\begin{aligned} x[k+1] &= Ax[k] + B_1 u_1[k] + B_2 u_2[k] \\ y^c[k] &= Cx[k] + \mathrm{e}_s[k]. \end{aligned}\right\} \tag{5.3}$$

We make the following assumption.

Assumption 4. *The matrix B_2 has full column rank and the output matrix C has full row rank.*

For each time k, let Y^k be the the vector composed of τ measurements of y^c on the sample interval $[k-\tau+1,\, k]$, where $\tau \geq 1$ is a design parameter. We have

$$\begin{aligned} Y^k \triangleq \begin{bmatrix} y^c[k] \\ y^c[k-1] \\ \vdots \\ y^c[k-\tau+1] \end{bmatrix} &= \begin{bmatrix} CA^{\tau-1} \\ CA^{\tau-2} \\ \vdots \\ C \end{bmatrix} x[k-\tau+1] \\ &+ \mathcal{B}_2 \begin{bmatrix} u_2[k-1] \\ u_2[k-2] \\ \vdots \\ u_2[k-\tau+1] \end{bmatrix} + I_{\tau p} \begin{bmatrix} \mathrm{e}_s[k] \\ \mathrm{e}_s[k-1] \\ \vdots \\ \mathrm{e}_s[k-\tau+1] \end{bmatrix} + \mathcal{B}_1 \begin{bmatrix} u_1[k-1] \\ u_1[k-2] \\ \vdots \\ u_1[k-\tau+1] \end{bmatrix} \\ &\triangleq \mathcal{O}^{\tau-1} x[k-\tau+1] + \mathcal{B}_2 U_2^k + I_{\tau p} E_s^k + \mathcal{B}_1 U_1^k, \end{aligned}$$

where $\mathcal{O}^{\tau-1} \in \mathbb{R}^{\tau p \times n}$, $\mathcal{B}_2 \in \mathbb{R}^{\tau p \times (\tau-1)(n_d+m)}$, $I_{\tau p} \in \mathbb{R}^{\tau p \times \tau p}$, $\mathcal{B}_1 \in \mathbb{R}^{\tau p \times (\tau-1)m}$. The matrices $\mathcal{B}_2$, $I_{\tau p}$, and $\mathcal{B}_1$ have the form

$$\mathcal{B}_1 = \begin{bmatrix} CB_1 & \cdots & CA^{\tau-2}B_1 \\ \vdots & \ddots & \vdots \\ O_{p\times m} & \cdots & CB_1 \\ O_{p\times m} & \cdots & O_{p\times m} \end{bmatrix}, \mathcal{B}_2 = \begin{bmatrix} CB_2 & \cdots & CA^{\tau-2}B_2 \\ \vdots & \ddots & \vdots \\ O_{p\times(d+m)} & \cdots & CB_2 \\ O_{p\times(d+m)} & \cdots & O_{p\times(d+m)} \end{bmatrix}, \; I_{\tau p} = \begin{bmatrix} 1 & & \\ & \ddots & \\ & & 1 \end{bmatrix},$$

where $O_{(\cdot)\times(\cdot)}$ denotes a zero matrix with appropriate dimensions. Note that $y^c[k]$ and $u^c[k]$ are known to the designer at all time instances. Let $\hat{Y}^k = Y^k - \mathcal{B}_1 U_1^k$. Then $\hat{Y}^k$ is also known at all times. We next let $\Omega = \begin{bmatrix} I_{\tau p} & \mathcal{B}_2 \end{bmatrix}$ and $\mathcal{E}^k = \begin{bmatrix} E_s^{k\top} & U_2^{k\top} \end{bmatrix}^\top$ to put $\hat{Y}^k$ into a form suitable for using norm minimization techniques:

$$\hat{Y}^k = \mathcal{O}^{\tau-1} x[k-\tau+1] + \Omega \mathcal{E}^k. \tag{5.4}$$

Note that $\hat{Y}^k \in \mathbb{R}^{\tau p}$, $\Omega \in \mathbb{R}^{\tau p \times (\tau p + (\tau-1)(n_d+m))}$, and $\mathcal{E}^k \in \mathbb{R}^{\tau p + (\tau-1)(n_d+m)}$. We will use (5.4) to estimate the vector $\mathcal{E}^k$, which will give estimates for e_s, e_a, and d that are needed in the construction of the state observers of the CPS.

5.3.2 Sparse errors and disturbance estimation

In this subsection, we present a norm approximation method for estimating the state $x[k]$, unknown input $u_2[k]$, and unknown output error $\mathrm{e}_s[k]$ in the CPS given by (5.3). More specifically, we use the 1-norm minimization problem to approximate the 0-norm minimization problem for estimating the sparse errors e_a and e_s, and the disturbance d.

To proceed, let $N_{\mathcal{O}^{\tau-1}} \in \mathbb{R}^{(\tau p - n) \times \tau p}$ be a left annihilator of $\mathcal{O}^{\tau-1}$, that is, $N_{\mathcal{O}^{\tau-1}}\mathcal{O}^{\tau-1} = O_{(\tau p-n)\times n}$. Premultiplying both sides of (5.4) by $N_{\mathcal{O}^{\tau-1}}$, we obtain $N_{\mathcal{O}^{\tau-1}}\hat{Y}^k = N_{\mathcal{O}^{\tau-1}}\Omega\mathcal{E}^k$. Since the pair (A, C) is assumed to be observable, we can choose $\tau > n - \operatorname{rank} C$ so that $\operatorname{rank}\mathcal{O}^{\tau-1} = n$. Then, $\operatorname{rank} N_{\mathcal{O}^{\tau-1}} = \tau p - n$. Let $Z^k = N_{\mathcal{O}^{\tau-1}}\hat{Y}^k$ and $W = N_{\mathcal{O}^{\tau-1}}\Omega$. Our objective is to solve for $\mathcal{E}^k$ subject to the constraint, $Z^k = W\mathcal{E}^k$. Note that $W \in \mathbb{R}^{(\tau p-n)\times(\tau p+(\tau-1)(n_d+m))}$ is full row rank since $\operatorname{rank} N_{\mathcal{O}^{\tau-1}} = \tau p - n$, $\operatorname{rank}\Omega = \tau p$, and $W = N_{\mathcal{O}^{\tau-1}}\Omega$.

To this end, let $\Sigma_{\mathrm{i}_\mathcal{E}} = \{\mathcal{E}^k : \|\mathcal{E}^k\|_0 \le \mathrm{i}_\mathcal{E}\}$ be the set of all $\mathrm{i}_\mathcal{E}$-sparse vectors and let $\mathcal{N}(W)$ denote the null space of the matrix W. We need the following lemma from [18]; see also [66].

Lemma 14. *If $\Sigma_{2\mathrm{i}_\mathcal{E}} \cap \mathcal{N}(W) = \{0\}$, then any $\mathrm{i}_\mathcal{E}$-sparse solution $\mathcal{E}^k$ to $Z^k = W\mathcal{E}^k$ is unique.*

We have the following corollary.

Corollary 2. *If spark$(W) > 2(\mathrm{i}_{E_s} + \mathrm{i}_{U_2})$, then any $\mathrm{i}_\mathcal{E}$-sparse solution $\mathcal{E}^k$ to $Z^k = W\mathcal{E}^k$ is unique.*

Remark 10. *Note that if spark$(W) > 2(\mathrm{i}_{E_s} + \mathrm{i}_{U_2})$, then $\tau p - n > 2(\mathrm{i}_{E_s} + \mathrm{i}_{U_2})$.*

It is shown in [21] that solving for the sparsest $\mathcal{E}^k$ subject to $Z^k = W\mathcal{E}^k$ is equivalent to solving the optimization problem,

$$\min \|\mathcal{E}^k\|_0 \quad \text{subject to} \quad Z^k = W\mathcal{E}^k.$$

It is observed in [18] that the above minimization problem can be approximated by the 1-norm minimization problem,

$$\tilde{\mathcal{E}}^k = \min \|\mathcal{E}^k\|_1 \quad \text{subject to} \quad Z^k = W\mathcal{E}^k. \tag{5.5}$$

To ensure the unique approximation, we choose τ so that the assumption of Corollary 2 is satisfied. We assume that over the time interval $[k-\tau+1,\, k]$, the vectors E_s^k and U_2^k are i_{E_s}-sparse and i_{U_2}-sparse, respectively. Premultiplying both sides of (5.4) by $(\mathcal{O}^{\tau-1})^\dagger$, we obtain

$$x[k-\tau+1] = (\mathcal{O}^{\tau-1})^\dagger \left(\hat{Y}^k - \Omega\mathcal{E}^k\right). \tag{5.6}$$

Let $\hat{x}[k]$ be the estimate of $x[k]$. Substituting $\tilde{\mathcal{E}}^k$ from (5.5) into the above equation gives

$$\hat{x}[k-\tau+1] = (\mathcal{O}^{\tau-1})^\dagger \left(\hat{Y}^k - \Omega\tilde{\mathcal{E}}^k\right). \tag{5.7}$$

Let $\tilde{U}_2^k$ be the estimate of U_2^k, then we have

$$\tilde{U}_2^k = \begin{bmatrix} \tilde{\mathcal{E}}^k_{\tau p+1} & \cdots & \tilde{\mathcal{E}}^k_{\tau p+(\tau-1)(n_d+m)} \end{bmatrix}^\top. \tag{5.8}$$

Using (5.7) and (5.8), we obtain the plant state estimate,

$$\begin{aligned} \hat{x}[k] =& A^{\tau-1}\hat{x}[k-\tau+1] + \begin{bmatrix} B_1 & \cdots & A^{\tau-2}B_1 \end{bmatrix} U_1^k \\ &+ \begin{bmatrix} B_2 & \cdots & A^{\tau-2}B_2 \end{bmatrix} \tilde{U}_2^k. \end{aligned} \tag{5.9}$$

Let $\tilde{\mathrm{e}}_s[k]$, $\tilde{\mathrm{e}}_a[k]$, and $\tilde{d}[k]$ be the estimates of $\mathrm{e}_a[k]$, $\mathrm{e}_a[k]$, and $d[k]$, respectively. We have

$$\tilde{\mathrm{e}}_s[k] = \begin{bmatrix} \tilde{\mathcal{E}}_1^k & \cdots & \tilde{\mathcal{E}}_p^k \end{bmatrix}^\top. \tag{5.10}$$

We also have,

$$\tilde{u}_2[k-1] = \begin{bmatrix} \tilde{U}^k_{2_1} & \cdots & \tilde{U}^k_{2_{n_d+m}} \end{bmatrix}^\top. \tag{5.11}$$

From (5.11), we obtain $\tilde{\mathrm{e}}_a[k-1]$ and $\tilde{d}[k-1]$, where

$$\tilde{d}[k-1] = \begin{bmatrix} \tilde{u}_{2_1}[k-1] & \cdots & \tilde{u}_{2_{n_d}}[k-1] \end{bmatrix}^\top, \tag{5.12}$$

$$\tilde{\mathrm{e}}_a[k-1] = \begin{bmatrix} \tilde{u}_{2_{n_d+1}}[k-1] & \cdots & \tilde{u}_{2_{n_d+m}}[k-1] \end{bmatrix}^\top. \tag{5.13}$$

5.3.3 The approximation accuracy

In [21], Candes and Tao use (5.5) to approximate the exact solution to $Z^k = W\mathcal{E}^k$, that is, using the 1-norm to approximate the 0-norm optimization problem. The reason for this is because the 0-norm optimization problem is NP-hard [35]. In addition, the 1-norm optimization problem is convex and the 1-norm solution is a good approximation to the 0-norm solution [40]. It is shown in [10], [15], [66] that the 1-norm approximation accuracy is related to the sparsity of the vector $\mathcal{E}^k$, that is, the bigger $s_{\mathcal{E}}$, the higher accuracy of using $\tilde{\mathcal{E}}^k$ to approximate $\mathcal{E}^k$. Following this fact, we will propose a novel fictitious output design method to increase the sparsity of the vector $\mathcal{E}^k$ in order to improve the estimation accuracy.

5.4 Application of the Combined Norm-UIO Based Observer

In this section, we apply the norm-UIO based observer architecture for estimating the state, the unknown input, and the unknown output error in the CPS modeled by (5.3).

5.4.1 UIO structure and unknown input estimation

To design the UIO, we need an estimate of the output $y^s[k]$. From (5.2), we have $y^s[k] = y^c[k] - \mathrm{e}_s[k]$. Since we already have an estimate $\tilde{\mathrm{e}}_s[k]$ of $\mathrm{e}_s[k]$ from (5.10), we can form the following estimate of the output $y^s[k]$,

$$\tilde{y}^s[k] = y^c[k] - \tilde{\mathrm{e}}_s[k]. \tag{5.14}$$

This plant output estimate from the norm-based estimator will be used in the combined UIO-based estimator; see Figure 5.1.

To proceed with our analysis, note that

$$\tilde{y}^s[k] = y^c[k] - \tilde{\mathrm{e}}_s[k] = Cx[k] + (\mathrm{e}_s[k] - \tilde{\mathrm{e}}_s[k]).$$

Let $f_s[k] = \mathrm{e}_s[k] - \tilde{\mathrm{e}}_s[k]$. Then the DT plant model can be represented as

$$\left.\begin{aligned} x[k+1] &= Ax[k] + B_1 u_1[k] + B_2 u_2[k] \\ \tilde{y}^s[k] &= Cx[k] + f_s[k]. \end{aligned}\right\} \tag{5.15}$$

We use the following UIO to estimate the state of the DT plant model (5.15):

$$\begin{aligned} z[k+1] =& (I - MC)\left(Az[k] + AM\tilde{y}^s[k] + B_1 u_1[k]\right) \\ &+ L(\tilde{y}^s[k] - Cz[k] - CM\tilde{y}^s[k]) \qquad (5.16a) \\ \hat{x}[k] =& z[k] + M\tilde{y}^s[k], \qquad (5.16b) \end{aligned}$$

where $M \in \mathbb{R}^{n\times p}$, and $L \in \mathbb{R}^{n\times p}$ are design parameter matrices, $z[k] \in \mathbb{R}^n$ is the state of the UIO, and $\hat{x}[k] \in \mathbb{R}^n$ is the state estimate of (5.15). The accuracy of the state estimate given by the UIO will be discussed in the next section.

We assume the following standard UIO existence conditions for the system, see for example, [68]:

- There exists a matrix L such that the matrix $(A_1 - LC)$ is Schur stable.
- There exists a matrix M such that $(I - MC)B_2 = O$.

In the following, we assume that such L and M have been chosen and are in use. It is easy to check that the second condition above is equivalent to the condition

$$\operatorname{rank} B_2 = \operatorname{rank}(CB_2), \tag{5.17}$$

since $\operatorname{rank}(MCB_2) \leq \operatorname{rank}(CB_2) \leq \operatorname{rank} B_2$. In fact, the rank condition above is a necessary condition for different types of UIO designs, see for example [30], [31].

Since B_2 by assumption has full column rank and (5.17) holds, CB_2 has a left inverse, for which we can use the Moore-Penrose pseudoinverse $(CB_2)^\dagger$. Premultiplying both sides of $x[k+1] = Ax[k] + B_1u_1[k] + B_2u_2[k]$ by the matrix $(CB_2)^\dagger C$, we obtain an exact formula for the unknown input:

$$u_2[k] = (CB_2)^\dagger \left(y^s[k+1] - CAx[k] - CB_1u_1[k]\right).$$

Since we do not have access to $y^s[k+1]$ or $x[k]$ directly, we use the approximations $\tilde{y}^s[k+1]$ and $\hat{x}[k]$ in their place and obtain an estimate of the unknown input $u_2[k]$:

$$\hat{u}_2[k] = (CB_2)^\dagger \left(\tilde{y}^s[k+1] - CA\hat{x}[k] - CB_1u_1[k]\right).$$

Note that there is a one step delay in the above estimate. We will use the above estimator in our convergence analysis. The unknown input estimator has the form

$$\hat{u}_2[k-1] = (CB_2)^\dagger \left(\tilde{y}^s[k] - CA\hat{x}[k-1] - CB_1u_1[k-1]\right). \tag{5.18}$$

The estimation error for the unknown input is

$$u_2[k] - \hat{u}_2[k] = (CB_2)^\dagger \left(CA\mathrm{e}[k] - f_s[k+1]\right), \tag{5.19}$$

which shows explicitly the dependency of the accuracy of the unknown input estimate on the state estimation error $\mathrm{e}[k]$ and the error $f_s[k+1]$ for the estimation of $\mathrm{e}_s[k+1]$. It is immediate that if $\mathrm{e}[k] \to 0$ and $f_s[k] \to 0$, which we do not know a priori, then $u_2[k] - \hat{u}_2[k] \to 0$. We will explore the issue of accuracy and convergence in the next subsection.

5.4.2 Approximation and convergence analysis

In this section, we discuss the accuracy of the state estimate and the unknown input estimate. Let $\mathrm{e}[k] = x[k] - \hat{x}[k]$ be the state estimation error. Then after some algebraic manipulations, we obtain

$$\begin{aligned}\mathrm{e}[k+1] =& ((I - MC)A - LC)\,\mathrm{e}[k] - Lf_s[k] \\ & - Mf_s[k+1] + (I - MC)B_2u_2[k] \\ =& ((I - MC)A - LC)\,\mathrm{e}[k] - Lf_s[k] - Mf_s[k+1]. \end{aligned} \tag{5.20}$$

Let $A_1 = (I - MC)A$. Then (5.20) can be represented as

$$\mathrm{e}[k+1] = (A_1 - LC)\,\mathrm{e}[k] - Lf_s[k] - Mf_s[k+1]. \tag{5.21}$$

Let $G = A_1 - LC$ and $N = -\begin{bmatrix} L & M \end{bmatrix}$, and $v[k] = \begin{bmatrix} f_s[k]^\top & f_s[k+1]^\top \end{bmatrix}^\top$. Let $h[k] = Nv[k]$, then (5.21) can be written as

$$\mathrm{e}[k+1] = G\mathrm{e}[k] + h[k]. \tag{5.22}$$

We have the following theorem.

Theorem 6. *Suppose $\{v[k]\} \in \ell_\infty$. If G is Schur stable and $\{h[k]\} \in \ell_\infty$, then there is $\zeta > 0$, which depends only on G, such that:*

1. *(term-wise absolute bound)*

$$\|\mathrm{e}[k]\| \le \|G^k\mathrm{e}[0]\| + \zeta\|\{h[k]\}\|_\infty \tag{5.23}$$

   for each $k \in \mathbb{N}$;

2. *(steady-state bound)*

$$\limsup_{k\to\infty} \|\mathrm{e}[k]\| \le \zeta \limsup_{k\to\infty} \|h[k]\|. \tag{5.24}$$

To prove Theorem 6, we need the following lemma.

Lemma 15. *Let $\{a_q\} \in \ell_1$ and $\{b_q\} \in \ell_\infty$ be nonnegative sequences. Then*

$$\limsup_{k\to\infty} \sum_{q=0}^{k} a_{k-q} b_q \le \|\{a_q\}\|_1 \limsup_{k\to\infty} b_k.$$

Proof. The claim is clearly true if $b_q = 0$ for all q. Suppose not all b_q's are zero. By scaling, we may assume that

$$\|\{a_q\}\|_1 = \sum_{q=0}^{\infty} a_q = 1.$$

We first treat the case when $\{b_q\}$ is a decreasing sequence, that is, $b_0 \ge b_1 \ge \cdots$. For each $k \ge 0$, let

$$p_k = \sum_{q=0}^{k} a_{k-q} b_q.$$

Let $m \ge 0$. Then for $k > m$, we have

$$\begin{aligned}
p_k &= \sum_{q=0}^{k} a_{k-q} b_q \\
&= \sum_{q=0}^{m-1} a_{k-q} b_q + \sum_{q=m}^{k} a_{k-q} b_q \\
&\le \|\{b_q\}\|_\infty \sum_{q=0}^{m-1} a_{k-q} + b_m \sum_{q=m}^{k} a_{k-q} \\
&= \|\{b_q\}\|_\infty \sum_{q=k-m+1}^{k} a_q + b_m \sum_{q=0}^{k-m} a_q \\
&\le \|\{b_q\}\|_\infty \sum_{q=k-m+1}^{k} a_q + b_m.
\end{aligned}$$

Let $\varepsilon > 0$. Note that there is $M \in \mathbb{N}$ such that if $m \ge M$, then $\sum_{q=m+1}^{\infty} a_{\mathrm{i}} < \varepsilon / \|\{b_q\}\|_\infty$. It follows that

$$p_k \le b_m + \varepsilon \text{ whenever } m \ge M \text{ and } k \ge 2m.$$

From this we can conclude that

$$\limsup_{k\to\infty} p_k \le b_m + \varepsilon \text{ for all } m \ge M$$

and thus

$$\limsup_{k\to\infty} p_k \le \limsup_{m\to\infty} b_m + \varepsilon.$$

Since $\varepsilon > 0$ is arbitrary, the case of decreasing $\{b_q\}$ follows.

For the general case, let

$$s_k = \sup\{b_q : q \ge k\}$$

for each $k \ge 0$. Then s_k is a decreasing sequence, $b_k \le s_k$ for each k, and $\limsup s_k = \limsup b_k$. Thus

$$\begin{aligned}\limsup_{k\to\infty} \sum_{q=0}^{k} a_{k-q} b_q &\le \limsup_{k\to\infty} \sum_{q=0}^{k} a_{k-q} s_q \\ &\le \limsup_{k\to\infty} s_k \\ &= \limsup_{k\to\infty} b_k\end{aligned}$$

and the proof of the lemma is complete. □

Using the above lemma, we will now prove Theorem 6.

Proof of Theorem 6. By simple recursion or induction, we have

$$\mathrm{e}[k] = G^k \mathrm{e}[0] + \sum_{\mathrm{j}=0}^{k-1} G^{\mathrm{j}} h[k-1-\mathrm{j}]. \tag{5.25}$$

Since G is Schur stable, we have $\sigma(G) < 1$, where $\sigma(G)$ is the spectral radius of G defined as $\sigma(G) = \max\{|\lambda_1(G)|, \ldots, |\lambda_n(G)|\}$. Recall that Gelfand's spectral radius formula [69, p. 195] states that

$$\sigma(G) = \lim_{k\to\infty} \|G^k\|^{1/k},$$

which implies $\lim_{k\to\infty} \|G^k\| = 0$. Thus there is $K \in \mathbb{N}$ such that $\|G^K\| < 1$. Let $\rho = \|G^K\|$ and $\xi = \max_{0\le k\le K-1}\{\|G^k\|\}$. Each $k \in \mathbb{N}$ can be written as $k = rK + s$, where r is a nonnegative integer and $s \in \{0, 1, \ldots, K-1\}$. Therefore

$$\begin{aligned} \|G^k\| &= \|G^{rK}G^s\| \\ &\le \|G^{rK}\|\|G^s\| \\ &\le \rho^r\|G^s\| \\ &\le \xi\rho^r. \end{aligned}$$

We can now conclude that

$$\sum_{k=0}^{\infty} \|G^k\| \le \frac{K\xi}{1-\rho},$$

which in particular implies that $\{\|G^k\|\} \in \ell_1$.

Lemma 15 and equation (5.25) gives both the absolute term-wise bound (5.23) and the steady-state bound (5.24). □

It is observed in [12], [22], [66] that the sparse error $\mathrm{e}_s[k]$ can be recovered with high accuracy when the sparsity $s_{\mathcal{E}}$ of $\mathcal{E}^k$ is close to 1. Then, for such cases, $f_s[k] \approx 0$ and it is perhaps possible to satisfy $\lim_{k\to\infty} f_s[k] = 0$. We therefore expect that when the sparsity $s_{\mathcal{E}}$ of $\mathcal{E}^k$ is close to 1, the state estimate and the unknown input estimate to be quite accurate after a transient period.

5.5 Comparison of the Proposed Observers

In this section, we compare the two types of observer designs given in Subsections 5.3.2 and 5.4.1.

5.5.1 Differences between the observers

The 1-norm approximation based observer uses the 1-norm to approximate the 0-norm optimization problem. The accuracy of such approximation depends on the sparsity of the vector $\mathcal{E}^k$. For example, if $d[k] = 0$ and $s_{\mathcal{E}} \approx 1$, then $\tilde{\mathcal{E}}^k \approx \mathcal{E}^k$. See [10], [22] for simulation results that illustrate the accuracy of the estimation with sparse vector $\mathcal{E}^k$. However, $s_{\mathcal{E}}$ cannot be guaranteed to be close to 1 since the plant of the CPS in a real environment usually contains arbitrary disturbance $d[k]$. For example, if $d[k]$ is normal distributed random noise, and $\mathrm{e}_s[k]$ and $\mathrm{e}_a[k]$ are sparse attacks on the output measurements and control signal transmission through communication networks, then the presence of $d[k]$ will decrease the sparsity of $\mathcal{E}^k$, which will reduce the accuracy of the 1-norm approximation.

To proceed with our discussion, recall that the vector $\mathcal{E}^k$ is the collection of $\mathrm{e}_s[k]$ to $\mathrm{e}_s[k-\tau+1]$, $u_2[k-1]$ to $u_2[k-\tau+1]$, and $u_2[k] = \begin{bmatrix} d^\top[k] & \mathrm{e}_a^\top[k] \end{bmatrix}^\top$. From (5.7) and (5.9), we see that the norm based state estimate is impacted by accuracy of the whole vector $\tilde{\mathcal{E}}^k$, including both the e_s and u_2 parts. On the other hand, we see from (5.20) that only the first p elements, or the e_s portion, of $\tilde{\mathcal{E}}^k$ impact the accuracy of the combined norm-UIO based state estimate. In particular, if the estimate for e_s is very accurate, then the UIO based state estimate should also be very accurate regardless of the accuracy of the u_2 estimate. An inaccurate estimate of u_2 would only impact the state estimate from the norm-based estimator. The norm-UIO based estimator is insensitive to such recovery errors. We thus can conclude that the combined norm-UIO based estimator should in general be more robust than the norm-based state estimator.

Furthermore, the state estimates may also contain iterative errors. Indeed, suppose there is a recovery error in $\tilde{\mathcal{E}}^k_{\tau p+1}$ through $\tilde{\mathcal{E}}^k_{\tau p+(\tau-1)(n_d+m)}$, then the delayed state estimate from (5.7) is incorrect. Since $\tilde{U}_2^k = \begin{bmatrix} \tilde{\mathcal{E}}^k_{\tau p+1} & \cdots & \tilde{\mathcal{E}}^k_{\tau p+(\tau-1)(n_d+m)} \end{bmatrix}^\top$ by (5.8), the state estimate given by (5.9) contains iterative errors.

The increased accuracy of the combined norm-UIO based estimator comes at the cost of additional processing and more stringent conditions for its existence. In particular, the matrix rank condition (5.17) may not be satisfied by some plants. For example, consider

an UAV model found in [48], which is remotely controlled as a CPS. For simplicity, we let $d[k] = 0$. The discrete state-space model has the form given by (5.2), where $B_2 = B_1$, and

$$A = \begin{bmatrix} 1 & 0.01 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0.01 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0.01 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, B_1 = \begin{bmatrix} 0.0001 & 0 & 0 \\ 0.1963 & 0 & 0 \\ 0 & 0.0001 & 0 \\ 0 & 0.0618 & 0 \\ 0 & 0 & 0.0002 \\ 0 & 0 & 0.3439 \end{bmatrix}, C^\top = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}.$$

In this example, the triple (A, B_1, C) is both reachable and observable, but $\operatorname{rank}(CB_2) \neq \operatorname{rank} B_2$. The combined norm-UIO based estimator does not exist for this plant model.

In the next subsection, we apply both estimators to a DT plant remotely controlled as a CPS. We compare the performance of the estimators discussed above.

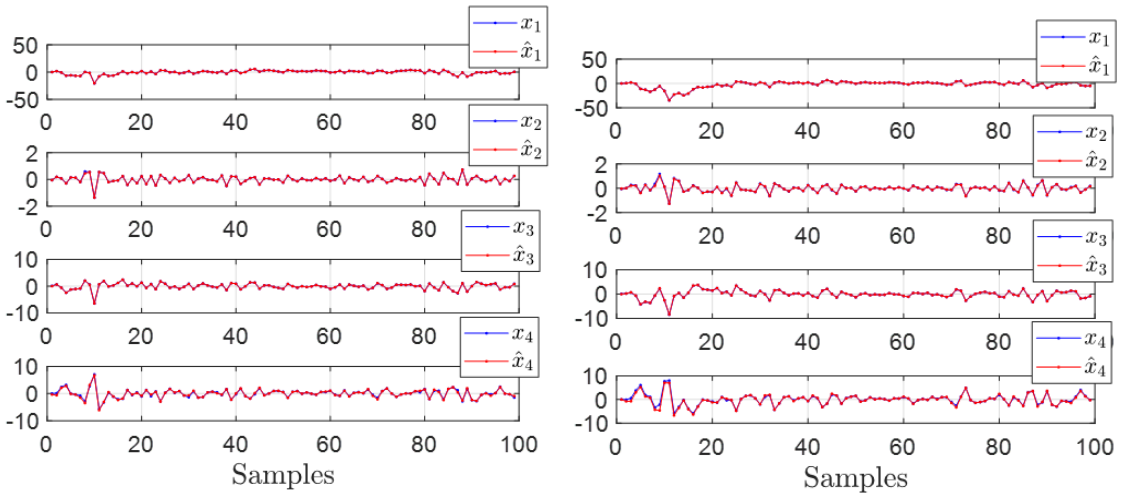
5.5.2 Example

We consider a CPS with the DT model of an inverted pendulum on a cart as the plant; see [55] for the modeling of this plant. The DT plant and the DT estimator are interconnected through unsecured communication networks as shown in Figure 5.1, where

$$A = \begin{bmatrix} 1 & -0.038 & 0.164 & -0.002 \\ 0 & 1.808 & 0.123 & 0.248 \\ 0 & -0.403 & 0.658 & -0.036 \\ 0 & 8.931 & 1.306 & 1.766 \end{bmatrix}, B_1 = \begin{bmatrix} 0.0057 \\ -0.0199 \\ 0.0550 \\ -0.2108 \end{bmatrix}, B_d = \begin{bmatrix} 0.953 \\ 0.073 \\ 0.207 \\ 0.775 \end{bmatrix}, C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}.$$

We assume that each element of $\mathrm{e}_a[k]$ and $\mathrm{e}_s[k]$ has 0.05 probability of being nonzero, and when the element is nonzero, the value of the element is normally distributed with 0 mean and 0.1 variance. We assume that the disturbance $d[k]$ is normally distributed also with 0 mean and 0.1 variance. As in [55], we use the initial state, $x[0] = \begin{bmatrix} 0.10 & -0.05 & 0.15 & 0.05 \end{bmatrix}^\top$. We choose $\tau = 10$ and assume that $y[-1]$ through $y[-9]$ and $u[-1]$ through $u[-9]$ are zero.

Simulation results are shown in Figure 5.2 and Figure 5.3. In Subfigures (5.2a) and (5.2b), the blue dot lines show the true states of the plant and the red dot lines show the estimated



(a) (b)

Figure 5.2. State estimates by (a) the norm-based estimator, and (b) the combined norm-UIO based estimator.

states generated by the estimators. In Subfigure (5.3a), the blue dot lines show the true sparse $\mathrm{e}_s[k]$ and the red dot lines show the recovered $\tilde{\mathrm{e}}_s[k]$ from the norm-based estimator. In Subfigures (5.3b) and (5.3c), the blue dot lines in the top and bottom subfigures are the true $d[k]$ and $\mathrm{e}_a[k]$, respectively. The red dot lines in Subfigure (5.3b) are the recovered $\tilde{d}[k]$ and $\tilde{\mathrm{e}}_a[k]$ from the norm-based estimator. The red dot lines in Subfigure (5.3c) are the recovered $\tilde{d}[k]$ and $\tilde{\mathrm{e}}_a[k]$ from the combined norm-UIO based estimator.

We can see in Subfigures (5.3a) and (5.3b) that the errors $\mathrm{e}_s[k]$, $d[k]$, and $\mathrm{e}_a[k]$ are not exactly recovered by the norm-based estimator. We hypothesize that the reason for not perfect recovery is that the presence of the disturbance $d[k]$ decreases the sparsity $s_{\mathcal{E}}$ of the vector $\mathcal{E}^k$ and thus it reduces the accuracy rate of the 1-norm approximation of the 0-norm optimization problem. Comparing Subfigure (5.3b) with Subfigure (5.3c), we see that the recovery of $\mathrm{e}_a[k]$ and $d[k]$ using the combined norm-UIO based estimator is superior compared with the norm-based estimator. To compare the performance of the estimators quantitatively, we evaluate the performance indices

$$\left.\begin{array}{rcl} J_d & = & \sum_{k=10}^{100} \|d[k] - \tilde{d}[k+1]\|_1 \\ J_{\mathrm{e}} & = & \sum_{k=10}^{100} \|\mathrm{e}_a[k] - \tilde{\mathrm{e}}_a[k+1]\|_1 \end{array}\right\} \tag{5.26}$$

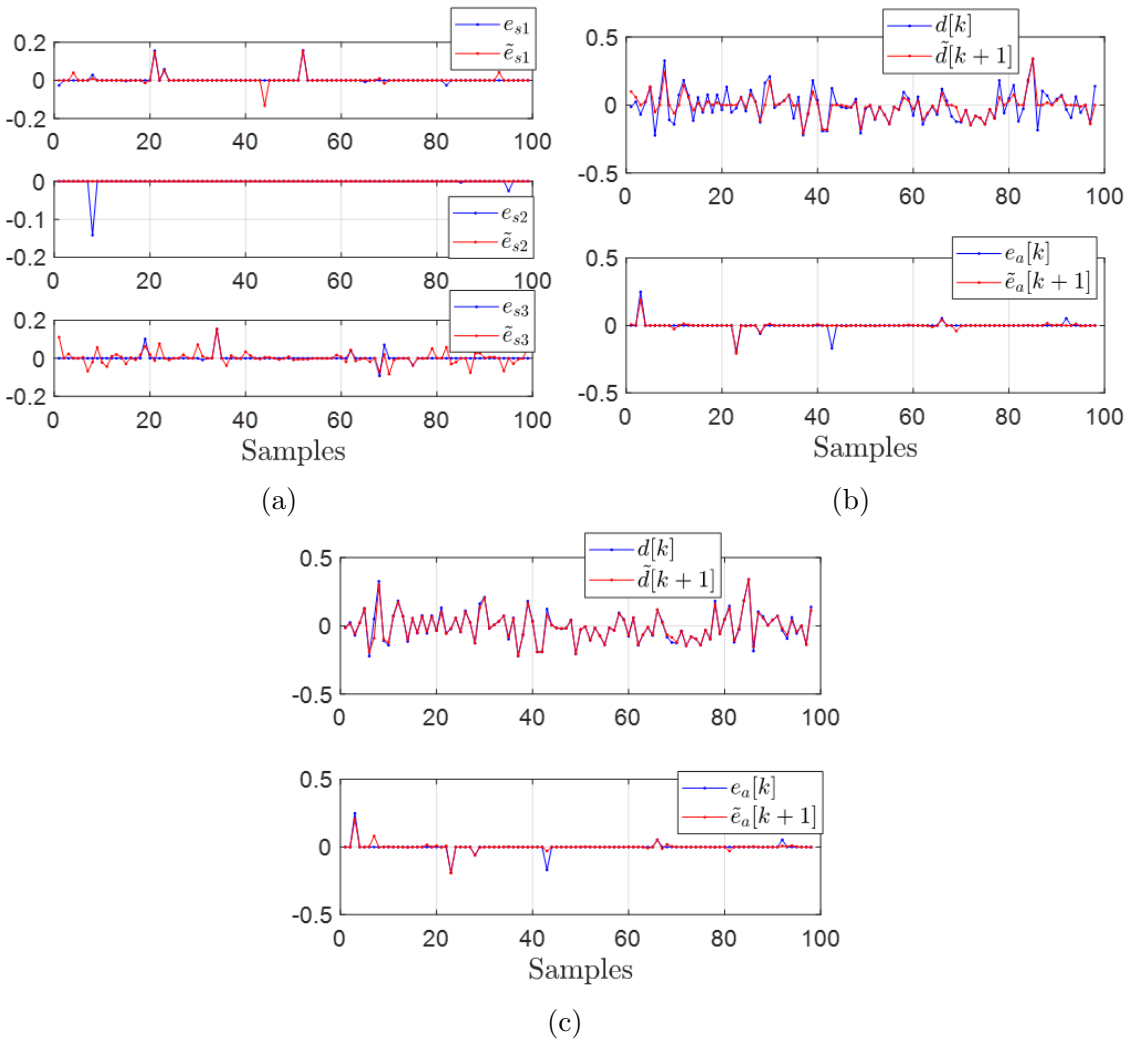


Figure 5.3. Sparse errors and disturbance recovery: (a) Recovery of $\mathrm{e}_s[k]$ by the norm-based estimator; (b) Recovery of $\mathrm{e}_a[k]$ and $d[k]$ by the norm-based estimator; and (c) Recovery of $\mathrm{e}_a[k]$ and $d[k]$ by the combined norm-UIO based estimator.

for the simulation results shown in Subfigures (5.3b) and (5.3c). The values of the indices for the simulations in Subfigure (5.3b) are: $J_d = 3.0902$ and $J_{\mathrm{e}} = 0.4572$, while the index values for the simulations of Subfigure (5.3c) are: $J_d = 0.6134$ and $J_{\mathrm{e}} = 0.3573$.

These results illustrate the superiority of the UIO-based estimator over the norm-based estimator discussed in Subsection 5.5.1.

5.6 Estimation Enhancement Using Fictitious Output Measurements

In this section, we propose a novel fictitious output measurements design method to improve the observers' performance for the system with sparse communication errors and arbitrary disturbance.

The technique of creating redundancies using linear combinations of signals is quite standard in communications and it is typically used to help identify and correct errors. The whole theory of error correcting codes is based on this idea. However, there is a price to be paid. In communications, as is the case here, the cost is bandwidth, delay, and processing.

5.6.1 Generating fictitious outputs

We conclude from Subsection 5.5.2 that the increased number of admissible nonzero elements in the vector $\mathcal{E}[k]$ caused by the arbitrary error $d[k]$ reduces the estimation accuracy. In other words, we need to increase the sparsity $s_{\mathcal{E}}$ of the vector $\mathcal{E}[k]$ in order to improve the estimators' performance. To this end, let $\mathbb{F}(y^s[k])$ denote a fictitious output vector whose elements are linear combinations of the output measurements $y^s[k]$. Let $y^s_{aug}[k] = \begin{bmatrix} y^s[k]^\top & \mathbb{F}(y^s[k])^\top \end{bmatrix}^\top$. The operation of the fictitious output measurements operator is depicted in Figure 5.4. On the plant side, the collected output measurements $y^s[k]$

Figure 5.4. Generating fictitious output measurements.

are transmitted to a coder. The coder generates $y^s_{aug}[k]$ and collects the information of all possible linear combinations of p sensor measurements that generate $\mathbb{F}(y^s[k])$. We use $\mathbb{L}(y^s)$ to denote the linear combinations information. Then, $y^s_{aug}[k]$ and $\mathbb{L}(y^s)$ are transmitted to a decoder on the estimator side through an unsecured communication network that introduces sparse errors $\mathrm{e}_{sn}[k]$ to the fictitious output measurements $y^s_{aug}[k]$. Note that $\mathbb{L}(y^s)$ is a fixed signal over time and therefore it can be easily recovered by comparing the copies

of this signal at the decoder side. The decoder decodes $\mathbb{L}(y^s)$ to obtain the matrix C_f and sends the corrupted output measurements $y^c_{aug}[k]$ and the matrix C_f to the estimator. Since the elements of $\mathbb{F}(y^s[k])$ are linear combinations of the elements of $y^s[k]$, the rows of matrix C_f are linear combinations of the rows of the plant output matrix C. We have the pair $(\mathbb{F}(y^s[k]), C_f)$ of the form

$$\mathbb{F}(y^s[k]) = \begin{bmatrix} y^s_1[k] + y^s_2[k] \\ \vdots \\ y^s_{p-1}[k] + y^s_p[k] \\ \vdots \\ y^s_1[k] + \cdots + y^s_p[k] \end{bmatrix}, C_f = \begin{bmatrix} C^{(1)} + C^{(2)} \\ \vdots \\ C^{(p-1)} + C^{(p)} \\ \vdots \\ C^{(1)} + \cdots + C^{(p)} \end{bmatrix}, \tag{5.27}$$

where $C^{(1)}, \ldots, C^{(p)}$ are the rows of the matrix C. Let p_f be the number of rows of C_f. Let $\mathcal{C}^{\mathrm{i}}_p$ denote the number of i-combinations from a given set of p elements, where $\mathcal{C}^{\mathrm{i}}_p = \frac{p!}{\mathrm{i}!(p-\mathrm{i})!}$. Then, in general, we can have C_f whose number of rows is given by

$$p_f = \sum_{\mathrm{i}=2}^{p} \mathcal{C}^{\mathrm{i}}_p.$$

Let $C_{aug} = \begin{bmatrix} C^\top & C_f^\top \end{bmatrix}^\top$. We then construct the augmented model with fictitious outputs of the form,

$$\left.\begin{aligned} x[k+1] &= Ax[k] + B_1 u_1[k] + B_2 u_2[k] \\ y^c_{aug}[k] &= C_{aug} x[k] + \mathrm{e}_{sn}[k]. \end{aligned}\right\} \tag{5.28}$$

In the next subsection, we show how to use the augmented model to improve the performance of the estimators.

5.6.2 Estimators' design using fictitious outputs

In this subsection, we first show that the original CPS given by (5.3) and the augmented CPS given by (5.28) share the same estimators' existence conditions. We then analyze the

reason why the proposed estimators' performances are improved after adding the fictitious output measurements.

We conclude from Subsection 5.3.2 that the norm-based estimator for the CPS given by (5.3) exists if

1. The pair (A, C) is observable;
2. $\text{spark}(W) > 2(\text{i}_{E_s} + \text{i}_{U_2})$.

We then conclude from Subsection 5.4 that the combined norm-UIO based estimator for (5.3) exists if

1. The pair (A, C) is observable;
2. $\text{spark}(W) > 2(\text{i}_{E_s} + \text{i}_{U_2})$;
3. $\text{rank}\, B_2 = \text{rank}(CB_2)$;
4. There exists a matrix $L \in \mathbb{R}^{n\times p}$ such that the matrix $(A_1 - LC)$ is Schur stable.

It is easy to verify that Condition 4) above is equivalent to the following linear matrix inequality (LMI):

$$\begin{bmatrix} -P & A_1^\top P - C^\top T^\top \\ PA_1 - TC & -P \end{bmatrix} \prec 0, \tag{5.29}$$

where $P = P^\top \succ 0$ and $T = PL$.

We now present the design lemma.

Lemma 16. *If the norm-based estimator and the combined norm-UIO based estimator exist for the CPS given by (5.3), then they also exist for the augmented CPS given by (5.28).*

Proof. Since C_f is a linear combination of the rows of matrix C and $C_{aug} = \begin{bmatrix} C^\top & C_f^\top \end{bmatrix}^\top$, it is obvious that the pair (A, C_{aug}) is observable if the pair (A, C) is observable. Let p_{aug} be the number of rows of matrix C_{aug}. Since $p_{aug} > p$, if we choose the same τ for both designs and if we have the same i_{E_s} and i_{U_2}, then $\tau p_{aug} - n > \tau p - n > 2(\text{i}_{E_s} + \text{i}_{U_2})$.

It is easy to see that (5.3) and (5.28) share the same conditions 1), 2), and 3). Let $T_{aug} = \begin{bmatrix} T & T_f \end{bmatrix}$ and use C_{aug} and T_{aug} in (5.29) instead of C and T to obtain

$$\begin{bmatrix} -P & A_1^\top P - \begin{bmatrix} C^\top & C_f^\top \end{bmatrix} \begin{bmatrix} T^\top \\ T_f^\top \end{bmatrix} \\ PA_1 - \begin{bmatrix} T & T_f \end{bmatrix} \begin{bmatrix} C \\ C_f \end{bmatrix} & -P \end{bmatrix},$$

which is equivalent to

$$\begin{bmatrix} -P & A_1^\top P - C^\top T^\top \\ PA_1 - TC & -P \end{bmatrix} + \begin{bmatrix} O_{n\times n} & -C_f^\top T_f^\top \\ -T_f C_f & O_{n\times n} \end{bmatrix}.$$

Let $T_f = O_{n\times p_f}$. Then we can take the same P and T as in (5.29) to design the estimators using the augmented model. □

5.6.3 Improving approximation with fictitious outputs

From Subsection 5.3.3, we conclude that the larger $s_{\mathcal{E}}$, the more accurate approximation of $\mathcal{E}^k$ by $\tilde{\mathcal{E}}^k$. Therefore, to show that adding the fictitious output measurements improves the performance of the estimators, we need to show that adding fictitious outputs increases the sparsity of $\mathcal{E}^k$. Note that

$$\mathcal{E}^k = \begin{bmatrix} E_s^k \\ U_2^k \end{bmatrix} = R \begin{bmatrix} \mathrm{e}_s[k] \\ \mathrm{e}_a[k-1] \\ d[k-1] \\ \vdots \end{bmatrix}, \tag{5.30}$$

where R is a row elementary matrix. We assume that $\mathrm{e}_{s_1}, \ldots, \mathrm{e}_{s_p}, \mathrm{e}_{a_1}, \ldots, \mathrm{e}_{a_m}, d_1, \ldots, d_{n_d}$ can be modeled as independent and identically distributed (i.i.d.) random variables. Let $\mathbb{P}((\cdot) \neq 0)$ denote the probability of a random variable not equal to 0. Since $\mathrm{e}_s[k]$ and $\mathrm{e}_a[k]$ are sparse and $d[k]$ is arbitrary, we assume $\mathbb{P}(\mathrm{e}_{s_h} \neq 0) = \mathcal{P}_1$, $\mathbb{P}(\mathrm{e}_{a_i} \neq 0) = \mathcal{P}_2$, and $\mathbb{P}(d_j \neq 0) = 1$, where $\mathcal{P}_1$ and $\mathcal{P}_2$ are positive negligibly small constants, and $h = 1, \ldots, p$,

$\mathrm{i} = 1, \ldots, m$, and $\mathrm{j} = 1, \ldots, n_d$. Let $\mu_{\mathcal{E}}$ denote the expectation of $\mathbb{P}(\mathcal{E}_l \neq 0)$, where $l = 1, \ldots, \tau p + (\tau - 1)(n_d + m)$. We then have

$$\begin{aligned}\mu_{\mathcal{E}} &= \frac{\sum_{h=1}^{p} \mathcal{P}_1 + \sum_{\mathrm{i}=1}^{m} \mathcal{P}_2 + n_d}{p + m + n_d} \\ &= \frac{p\mathcal{P}_1 + m\mathcal{P}_2 + n_d}{p + m + n_d} \approx \frac{n_d}{p + m + n_d}.\end{aligned}$$

Let $\mathcal{E}_{aug}^k$ denote the signal vector (5.30) for the augmented system given by (5.28). Since $p_{aug} > p$ and $p_{aug} = p + p_f$, choosing the same $\mathcal{P}_1$, $\mathcal{P}_2$, and $\mathbb{P}(d_{\mathrm{j}} \neq 1) = 1$, we obtain

$$\mu_{\mathcal{E}_{aug}} = \frac{(p + p_f)\mathcal{P}_1 + m\mathcal{P}_2 + n_d}{(p + p_f) + m + n_d} \approx \frac{n_d}{(p + p_f) + m + n_d} < \mu_{\mathcal{E}},$$

which implies the increased sparsity $s_{\mathcal{E}_{aug}}$ of $\mathcal{E}_{aug}^k$ after adding the fictitious output measurements.

5.6.4 Example

This example illustrates how we can improve the observers' performance using the augmented output matrix C_{aug} augmented with the matrix of fictitious outputs C_f, where

$$C_f = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 \end{bmatrix}.$$

We assume 0.05 probability of each element of $\mathrm{e}_a[k]$ and $\mathrm{e}_{sn}[k]$ to be nonzero and we assume $d[k]$ to be a normal distributed error with 0 mean and 0.1 variance. The simulation results are shown in Figure 5.5. The values of the performance indices for the simulations in Subfigure 5.5b are: $J_d = 5.3600 \times 10^{-10}$ and $J_{\mathrm{e}} = 2.8195 \times 10^{-10}$, while the index values for the simulations of Subfigure 5.5c are: $J_d = 3.5335 \times 10^{-10}$ and $J_{\mathrm{e}} = 2.3171 \times 10^{-10}$, which is a significant improvement compared with the model without the fictitious outputs. To further illustrate the effectiveness of the fictitious output method, we increase the probability of each element of $\mathrm{e}_a[k]$ and $\mathrm{e}_{sn}[k]$ to be nonzero from 0.05 to 0.20. As expected, the performance

indices J_d and J_e increase as the probability increases. The performance indices J_d and J_e have the values, $J_d = 0.8561$ and $J_e = 0.2181$ for the norm-based observer and $J_d = 0.7376$ and $J_e = 0.1506$ for the combined norm-UIO based observer. The simulation results of the observers' performance when the probability reaches 0.20 are shown in Figure 5.6.

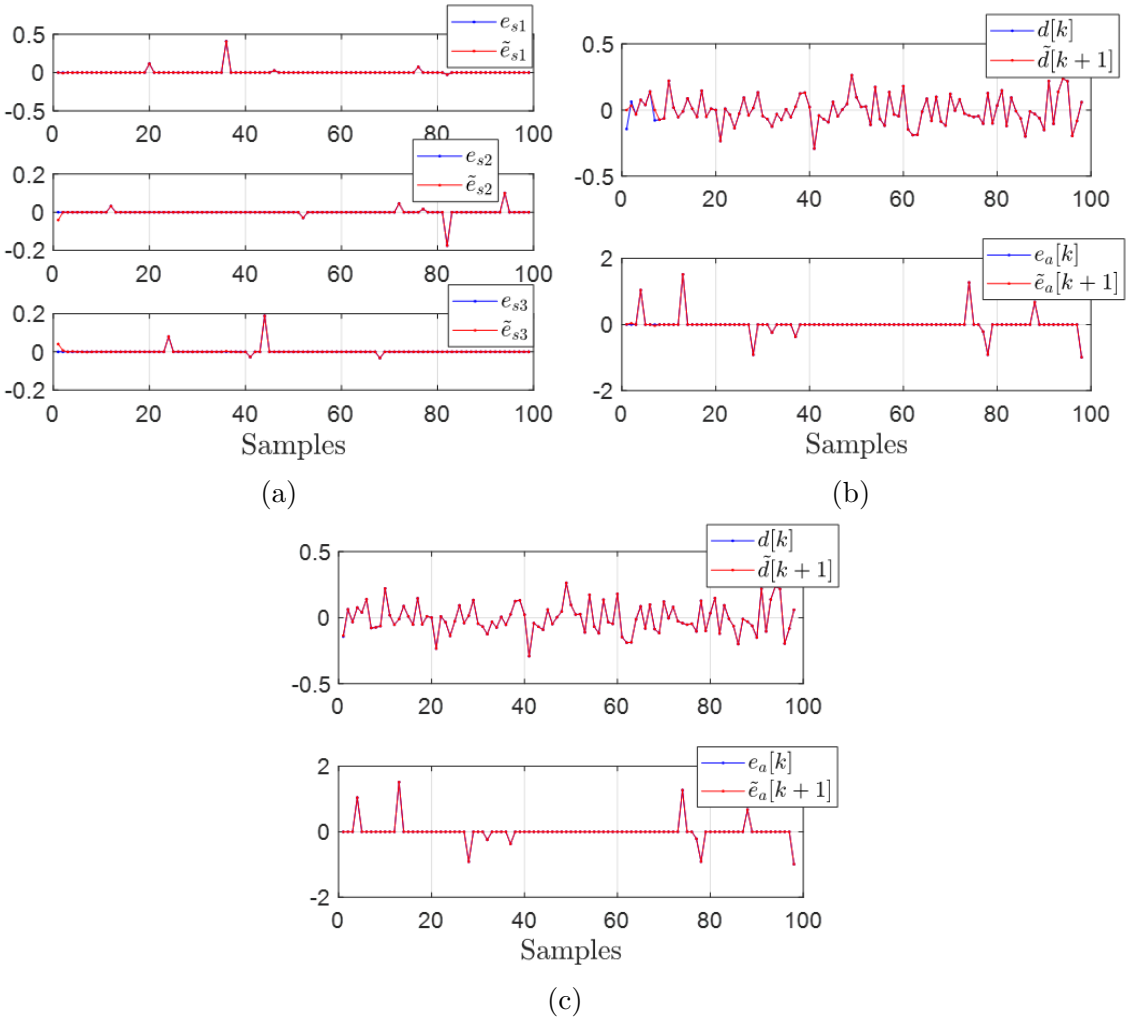


Figure 5.5. Sparse errors and disturbances recovery after adding the fictitious output measurements: (a) Recovery of $e_s[k]$ by the norm-based observer; (b) Recovery of $e_a[k]$ and $d[k]$ by the norm-based observer; and (c) Recovery of $e_a[k]$ and $d[k]$ by the combined norm-UIO based observer.

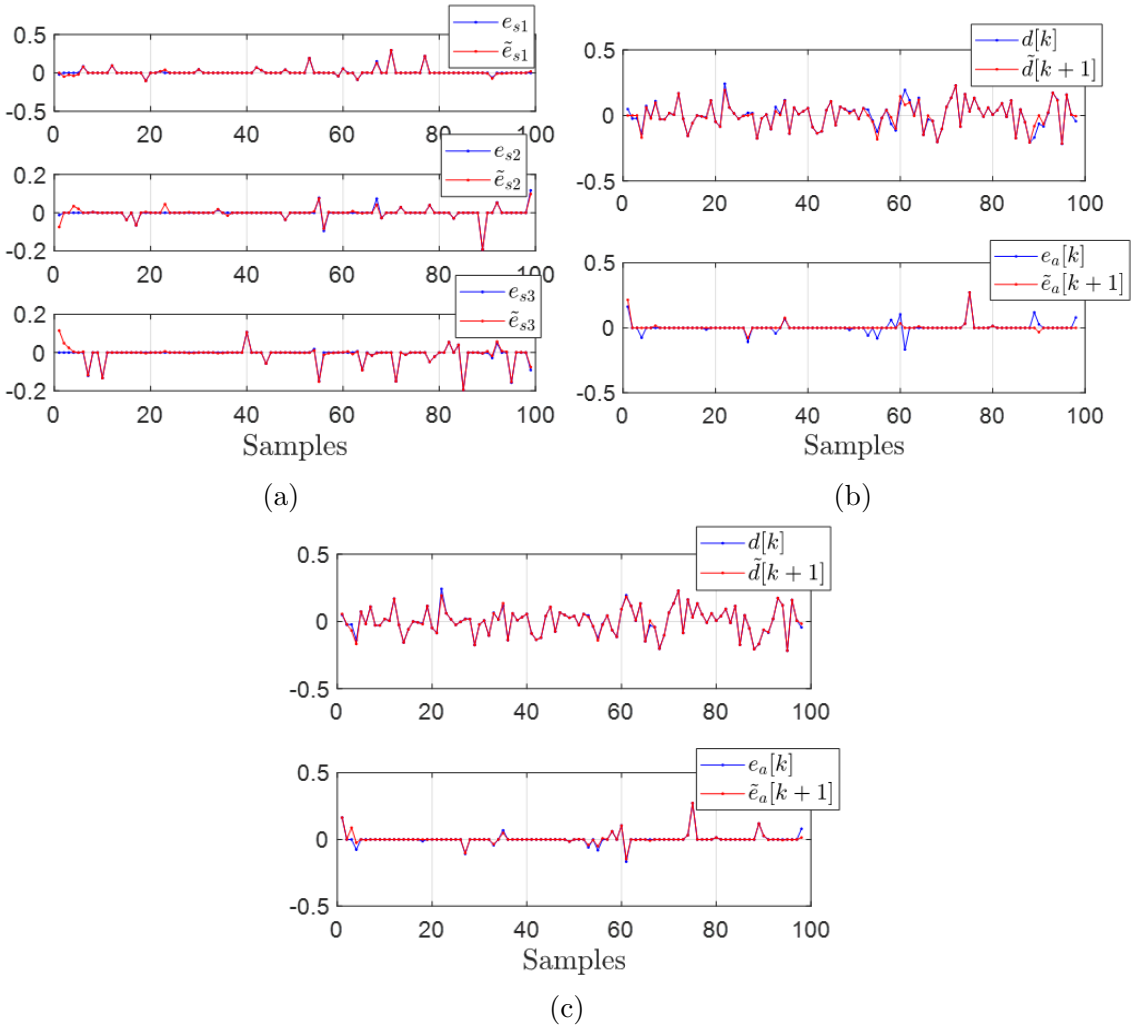


Figure 5.6. Sparse errors and disturbance recovery using the fictitious output measurements after decreasing the sparsity s_{e_a} and $s_{\mathrm{e}_{sn}}$ from 0.95 to 0.80.

5.6.5 Combined norm-UIO based observers for a CPS with sparse communication errors and subjected to arbitrary disturbances

In this subsection, we present a combined norm-UIO based observer to simultaneously estimate the state, the sparse communication errors, and arbitrary disturbances for a CPS depicted in Figure 5.1. In our design, we first check if the conditions for the existence of the UIO are satisfied. If the UIO does not exist, then we use only the norm-based observer to estimate the state as well as the communication errors and the disturbance. We summarize our discussion in Algorithm 1.

Algorithm 1: Norm-based and combined norm-UIO observer-controller compensator design

1 For a system given by (5.1), check if the triple (A, B_1, C) is reachable and observable; If not, STOP
2 Choose τ such that Corollary 2 is satisfied and compute $\tilde{\mathcal{E}}^k$ from (5.5) using a linear programming algorithm
3 Compute $\tilde{\mathrm{e}}_s[k]$ using (5.10)
4 Construct C_f and form $C_{aug} = [C^\top \;\; C_f^\top]^\top$
5 Check if the matrix rank condition given by (5.17) is satisfied, NEXT; If not, use (5.9), (5.12), and (5.13) to estimate $x[k]$, $d[k]$, and $\mathrm{e}_a[k]$, respectively
6 Let $C := C_{aug}$; Solve $(I - MC)B_2 = 0$ to obtain

$$M = B_2 \left((CB_2)^\dagger + M_0(I_p - (CB_2)(CB_2)^\dagger) \right)$$

7 Let $A_1 = (I - MC)A$ and $T = PL$; Compute P and T by solving (5.29)
8 Check if $P = P^\top \succ 0$, let $L = P^{-1}T$, NEXT; If not, use (5.9), (5.12), and (5.13) to estimate $x[k]$, $d[k]$, and $\mathrm{e}_a[k]$, STOP
9 Use (5.16) and (5.18) to estimate $x[k]$ and $u_2[k]$
10 Apply (5.12) and (5.13) to $\hat{u}_2[k-1]$ from (5.18) to estimate $d[k]$ and $\mathrm{e}_a[k]$.
11 Compute the control feedback gain K_d and let $u[k] = -K_d\hat{x}[k] + r$, where r is the reference signal.

5.7 Conclusions

We presented two observer architectures to simultaneously estimate the state, communication errors, and unknown disturbances in CPSs with sparse communication errors and subjected to arbitrary disturbances. We compared these two designs and showed the superiority of the combined norm-UIO based observer over the norm-based observer. We proposed using fictitious output measurements to improve the performance of the observers. A combined norm-UIO based observer design algorithm was formulated. The proposed observers can be used to design UIO-based fault detection and isolation (FDI) algorithms as well as the distributed fault-tolerant control for large-scale interconnected systems; see, for example, [25], [70] for related approaches.

6. CONTROLLER-OBSERVER COMPENSATOR SYNTHESIS FOR UNSECURED CYBER-PHYSICAL SYSTEMS

6.1 Introduction

Data and signal processing in CPSs are usually done in the discrete time (DT) domain. During the system analysis, a continuous time (CT) signal from the physical plant is sampled and then collected by DT sensors. The collected DT signal is transmitted to the control subsystem through communication networks. The control subsystem processes the received signal and generates a DT feedback signal that is sent to the physical plant through communication networks. On the plant side, the feedback signal is used by DT actuators to effect control commands. We refer to [71] for detailed discussion of the DT sensors and actuators, and [72] for the modeling of the CPS. The communication signals received by the CPS sensors and actuators may be corrupted by random disturbances and sensor and actuator faults. This necessities the analysis of the impact of this type of disturbances on CPS performance. In this chapter, we use ideas from the theory of error correcting codes (ECC) to design filters that mitigate and in many cases eliminate errors from sensor and actuator faults. In addition, an accurate state vector estimate must be available for effective control feedback implementation. A common approach to estimate the state of a system subjected to disturbances is through an unknown input observer (UIO), see for example, [26]–[28]. See also [31], [73], [74] for comparative study and different approaches for the UIO designs.

Since the CPSs are interconnected by communication networks that are not necessarily secure, the issue of reliable data communication must be addressed in the design of CPSs. In particular, sparse attacks in communication networks need to be addressed. As discussed in Chapter (5), the sparse attacks problem can be transformed into a 0-norm minimization problem that is usually solved using the ℓ_1-norm approximation method. One disadvantage of using the norm approximation method for sparse malicious attacks recovery in CPS is that it creates one sampling period time delay [22], [23]. Therefore, a robust control strategy needs to take this issue into account. A notion of robustness for cyber systems inspired by existing notions of input-output stability is introduced in [24]. In this chapter, we propose a

model reference robust control strategy to overcome the presence of malicious attacks during the control signal transmission.

In this chapter, we present a controller-observer compensator for a CPS with sensor and actuator faults and subjected to sparse malicious attacks. We propose a novel state observer architecture that combines a norm approximator with a bank of UIOs. We form an augmented CPS model using output signal accumulation with fictitious measurements to satisfy the UIO existence conditions. We propose novel sensor and actuator fault filters using error correcting code (ECC) approach. We present a model reference controller whose performance can be specified. The proposed controller design is presented in terms of linear matrix inequalities (LMIs) and its guaranteed performance is characterized.

6.2 Unsecured Cyber-Physical System

In this section, we discuss the modeling of a CPS with unsecured communication networks and subjected to sensor and actuator faults.

6.2.1 The CPS modeling

We consider a CT linear time-invariant (LTI) plant model of the form

$$\left.\begin{array}{rcl} \dot{x}(t) & = & A_c x(t) + B_c u(t) \\ y(t) & = & C_c x(t), \end{array}\right\} \tag{6.1}$$

where $A_c \in \mathbb{R}^{n\times n}$, $B_c \in \mathbb{R}^{n\times m}$, and $C_c \in \mathbb{R}^{p\times n}$. We assume the pairs (A_c, B_c) and (A_c, C_c) to be controllable and observable, and $p \geq m$. The CT plant is remotely controlled by the digital controller-observer compensator as shown in Figure 6.1. The compensator receives output measurement samples and generates the DT control signal that is sent to the plant through unsecured communication networks.

Our design of the digital combined controller-observer compensator is performed in the DT domain. We discretize the continuous plant model using the exact discretization method.

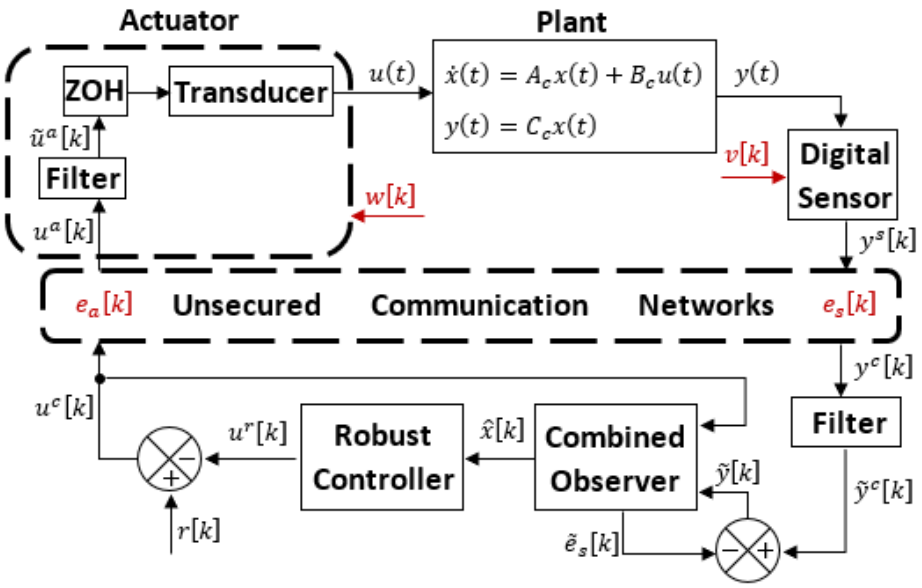


Figure 6.1. A block diagram of the CPS considered in this chapter.

Let T_s denote the sampling time period used to discretize the CT plant model. Then the discretized plant model has the form

$$\left.\begin{aligned} x[k+1] &= Ax[k] + Bu[k] \\ y[k] &= Cx[k], \end{aligned}\right\} \tag{6.2}$$

where $A = \mathrm{e}^{A_c T_s}$, $B = \int_0^{T_s} \mathrm{e}^{A_c \eta} B_c d\eta$, and $C = C_c$. See, for example [61, Subsection 1.1.2 and Chapter 2] or [62, Subsection 4.2.1] for discussions on modeling and the properties of DT systems. We select the discretization time T_s such that the pair (A, B) is reachable—see [62, p. 209] for a discussion of selecting the desired T_s. In our analysis and design, we will assume that the matrix B has full column rank, the output matrix C has full row rank, and $\operatorname{rank} B = \operatorname{rank}(CB)$. The plant output $y(t)$ from (6.1) is sensed by imperfect digital sensors to give an output $y^s[k] = y[k] + v[k]$, where $v[k]$ is the sensor fault. This corrupted sensor signal is transmitted to the controller-observer compensator through an unsecured communication network. We assume the network is under sparse maliciously injected attacks labeled $\mathrm{e}_s[k]$. The corrupted signal received by the compensator then has the form $y^c[k] = y[k] + v[k] + \mathrm{e}_s[k]$. The compensator generates the control signal $u^r[k]$, which

is then transmitted along with the reference command signal $r[k]$ to the plant. During the control signal transmission, the unsecured network is subjected to sparse maliciously injected attacks to the control signal denoted $\mathrm{e}_a[k]$. We also take into account the presence of the actuator fault labeled $w[k]$. The input to the filter is $u^a[k] = u^c[k] + \mathrm{e}_a[k] + w[k]$. The output of the filter is $\tilde{u}^a[k]$, which is then passed through the zero-order-hold (ZOH) element and applied to the transducer. The closed-loop CPS is modeled as the DT system of the form,

$$\left.\begin{array}{rcl} x[k+1] & = & Ax[k] + B(u^c[k] + \mathrm{e}_a[k] + w[k]) \\ y^c[k] & = & Cx[k] + v[k] + \mathrm{e}_s[k]. \end{array}\right\} \tag{6.3}$$

In the following sections, we will show how to design filters that are capable of filtering out actuator and sensor faults from system (6.3).

6.2.2 Modeling of sensor and actuator faults and sparse malicious attacks

In this subsection, we discuss how we model the sensor and actuator faults and the sparse malicious attacks. We assume that each component of the the sensor and actuator vectors are at fault with probabilities $\mathcal{P}_v$ and $\mathcal{P}_w$, respectively, and the nonzero fault values are given by continuous probability density functions (p.d.f.). To make this precise, let δ denote the Dirac delta function at 0 and let f_v, f_w be continuous p.d.f. on $\mathbb{R}$ (such as the uniform and normal distributions). We assume that for all i, k, $v_{\mathrm{i}}[k]$ are independent identically distributed (i.i.d.) real-valued random variables with p.d.f. $\mathbb{P}_v = (1-\mathcal{P}_v)\delta + \mathcal{P}_v f_v$ and that for all j, k, $w_{\mathrm{j}}[k]$ are i.i.d. real-valued random variables with p.d.f. $\mathbb{P}_w = (1-\mathcal{P}_w)\delta + \mathcal{P}_w f_w$, see for example [75, Section 4.3.2]. It then follows that $\mathrm{Prob}(v_{\mathrm{i}}[k] \neq 0) = \mathcal{P}_v$ and that nonzero values are given by the continuous p.d.f. f_v. Similar considerations apply to the random variables $w_{\mathrm{j}}[k]$ with $\mathrm{Prob}(w_{\mathrm{j}}[k] \neq 0) = \mathcal{P}_w$.

In our discussion, we assume that the malicious attacks $\mathrm{e}_s[k]$ and $\mathrm{e}_a[k]$ are sparse at every sample time k, that is, the 0-*norm* $\|\mathrm{e}_s[k]\|_0 < \frac{p}{2}$ and $\|\mathrm{e}_a[k]\|_0 < \frac{m}{2}$. As in the case for sensors and actuators, we also assume that $\mathrm{e}_{s_{\mathrm{i}}}[k]$ and $\mathrm{e}_{a_{\mathrm{j}}}[k]$ are i.i.d. with continuous probability density when they are nonzero. We assume the attacks $\mathrm{e}_s[k]$ and $\mathrm{e}_a[k]$ are maliciously injected to the unsecured communication networks during the signal transmission between the plant

and the controller-observer compensator. When different signals are transmitted through the same network at the same sample time, these signals are all corrupted by the same maliciously injected attacks.

We model the sensor and actuator faults and the sparse malicious attacks as uniformly bounded signals that are functions of k and the bounds are known to the designer.

In the following section, we discuss sensor and actuator fault filters design.

6.3 Sensor and Actuator Fault Filters Design

In this section, we use ideas from the theory of error correcting codes (ECC) to design filters that mitigate and in many cases eliminate errors from sensor and actuator faults. The well known fundamental idea of ECC is that the redundancy introduced allows a properly designed receiver to compare different "copies" to detect and eliminate errors [32, p. 355]. In this study, we use a repetition scheme as shown in the block diagram in Figure 6.2. The advantage of this scheme is that it is simple and, as we will show, quite effective. Before proceeding with our design, we first compute the probability of error for the repetition scheme applied to errors with p.d.f of the form $\mathbb{P} = (1 - \mathcal{P})\delta + \mathcal{P}f$, where f is a continuous p.d.f.

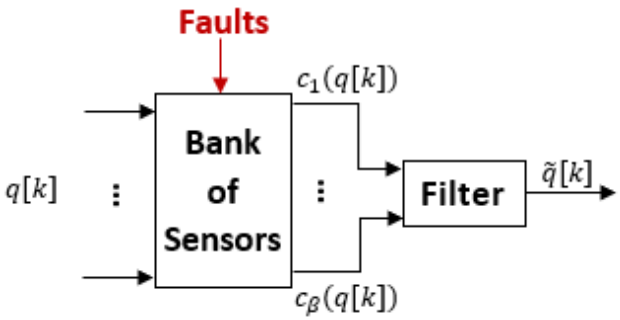


Figure 6.2. A block diagram of the sensor fault filter.

6.3.1 Error Probability for Repetition Filter

We first consider the scalar case. Suppose a device makes an additive error to a real-valued input a with the given p.d.f. We make β copies of a and pass it through β independent copies of the device. Then the outputs have the form $\alpha_{\mathrm{j}} = a + \xi_{\mathrm{j}}$, $\mathrm{j} = 1, \ldots, \beta$, where $\{\xi_{\mathrm{j}}\}_{\mathrm{j}=1}^{\beta}$

is a collection of i.i.d. real-valued random variable with the given p.d.f. Let η be the number of ξ_{j}'s that are nonzero. If $\eta \leq \beta - 2$, or equivalently at least two of the ξ_{j}'s are zero, then there are at least two copies of the original message a in the collection $\{\alpha_{\mathrm{j}}\}_{\mathrm{j}=1}^{\beta}$. From basic probability theory [44, p. 93], we have

$$\mathrm{Prob}\,(\eta \leq \beta - 2) = 1 - \beta(1-\mathcal{P})\mathcal{P}^{\beta-1} - \mathcal{P}^{\beta}.$$

For convenience, let $\mathfrak{p}_2 = \mathrm{Prob}\,(\eta \leq \beta - 2)$. For example, when $\mathcal{P} = 0.1$ and $\beta = 5$, we have $\mathfrak{p}_2 > 0.9995$. So we look for two elements that are equal in the collection $\{\alpha_{\mathrm{j}}\}_{\mathrm{j}=1}^{\beta}$. Note it follows from our assumption that the p.d.f. is continuous except at the origin that $\alpha_{\mathrm{i}} = \alpha_{\mathrm{j}} \neq a$ for $\mathrm{i} \neq \mathrm{j}$ can happen only with probability zero. It follows that if there are two or more equal elements in $\{\alpha_{\mathrm{j}}\}_{\mathrm{j}=1}^{\beta}$, they must equal a with probability one. We conclude that a can be recovered exactly with probability at least $\mathfrak{p}_2$. Once we have a, we obtain each $\xi_{\mathrm{j}} = \alpha_{\mathrm{j}} - a$ exactly.

If we do not find two elements that are equal in the collection $\{a + \xi_{\mathrm{j}}\}_{\mathrm{j}=1}^{\beta}$, which happens with probability one if $\mathcal{P} = 1$, we use the average of the copies as the estimate:

$$\bar{a} = \frac{1}{\beta}\sum_{\mathrm{j}=1}^{\beta} \alpha_{\mathrm{i}}.$$

The estimation error is $\sum_{\mathrm{j}=1}^{\beta} \xi_{\mathrm{j}}/\beta$, which has a variance that decreases at the rate of $1/\beta$. If the fault has zero mean, then $\bar{a}$ provides a good estimate of a for a reasonable β. With $\bar{a}$, we estimate ξ_{j} by $\bar{\xi}_{\mathrm{j}} = \alpha_{\mathrm{j}} - \bar{a}$.

We next consider the case when $a = (a_1, \ldots, a_l)$ is a real l-dimensional vector. We assume that the errors for each component of a are independent and apply the scalar case to each component a_{i} separately. As in the scalar case, we use β copies of a and β devices that each can handle l components independently. We obtain an estimate $\bar{a}$ using the scalar procedure on each component. Each component of $\bar{a}$ is either exactly correct or is an average of the β copies. By the independence assumption, we have

$$\mathrm{Prob}\,(\bar{a} = a) \geq \mathfrak{p}_2^l.$$

6.3.2 System Equations after Filtering

In our case, the input to the sensor is $u^c[k] + \mathrm{e}_a[k]$, which the sensor changes to $u^c[k] + \mathrm{e}_a[k] + w[k]$. The sensor filter either restores the original input to the sensor $u^c[k] + \mathrm{e}_a[k]$ exactly with probability at least $\mathfrak{p}_2^l$ or outputs an estimate whose accuracy depends on β. Similarly, the actuator filter either restores the original input to the actuator $y^s[k] + \mathrm{e}_s[k]$ exactly or gives an estimate with similar probability and condition.

We choose β large enough so that $\mathfrak{p}_2^l$ is close to 1 and that when the filter outputs only an estimate, it is accurate with high probability. Therefore, we will use a model that assumes that the filters are perfect in our design. Then the model of the closed-loop CPS, after the filtering operation, simplifies to

$$\left.\begin{aligned} x[k+1] &= Ax[k] + B(u^c[k] + \mathrm{e}_a[k]) \\ \tilde{y}^c[k] &= Cx[k] + \mathrm{e}_s[k]. \end{aligned}\right\} \tag{6.4}$$

6.3.3 An example illustrating the enhancement of the norm-UIO based observer performance by adding the sensor fault filter

In this subsection, we consider a CPS with the DT model of an inverted pendulum on a cart as the plant; see [55] for the modeling of this plant. Our objective is to show the significant estimation improvement of the norm-UIO based observer, which is given in Chapter 5, after adding the sensor fault filter.

The DT plant model's parameter matrices are:

$$A = \begin{bmatrix} 1 & -0.038 & 0.164 & -0.002 \\ 0 & 1.808 & 0.123 & 0.248 \\ 0 & -0.403 & 0.658 & -0.036 \\ 0 & 8.931 & 1.306 & 1.766 \end{bmatrix}, \; B = \begin{bmatrix} 0.0057 \\ -0.0199 \\ 0.0550 \\ -0.2108 \end{bmatrix}, C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}.$$

For simplicity, we let $u^c[k] = \mathrm{e}^a[k] = w[k] = 0$. In this example, the probability that a component of $y^s[k]$ contains random sensor fault is 0.7 and when the component is at fault, the value of the fault is uniformly distributed in the range $[0, 1]$. We assume that each

component of $e_s[k]$ has probability 0.15 of being nonzero and when the element is nonzero, the value of the component is normally distributed with zero mean and unit variance. We set the initial state $x[0] = \begin{bmatrix} 0 & 0 & 0 & 0 \end{bmatrix}^\top$. We choose $\tau = 10$ and assume that $y[-9]$ through $y[-1]$ and $u[-9]$ through $u[-1]$ are zero. The filter uses $\beta = 50$ signal copies. The simulation results are shown in Subfigure (6.3a), Subfigure (6.3b), and Figure 6.4. In

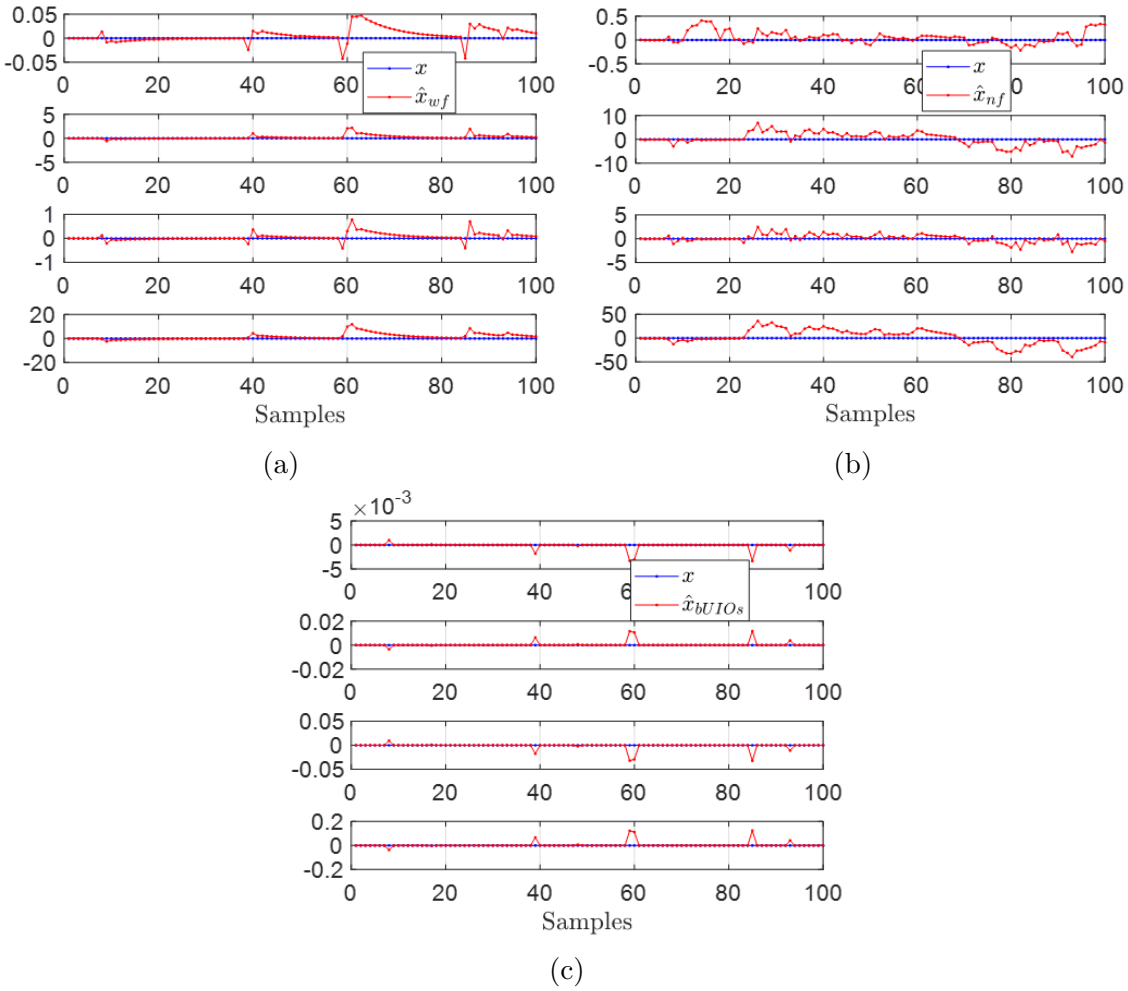


Figure 6.3. State estimates by (a) the combined observer with filter; (b) the combined observer only; (c) the bank of UIOs with filter. Note that the scale of the y-axis for the plots are different in order to show more detail.

Subfigures (6.3a) and (6.3b), the blue dot lines show the true states of the plant and the red dot lines show the estimated states generated by the combined observer with and without the filter, respectively. In Subfigures (6.4a) and (6.4b), the blue dot lines are the true $e_s[k]$

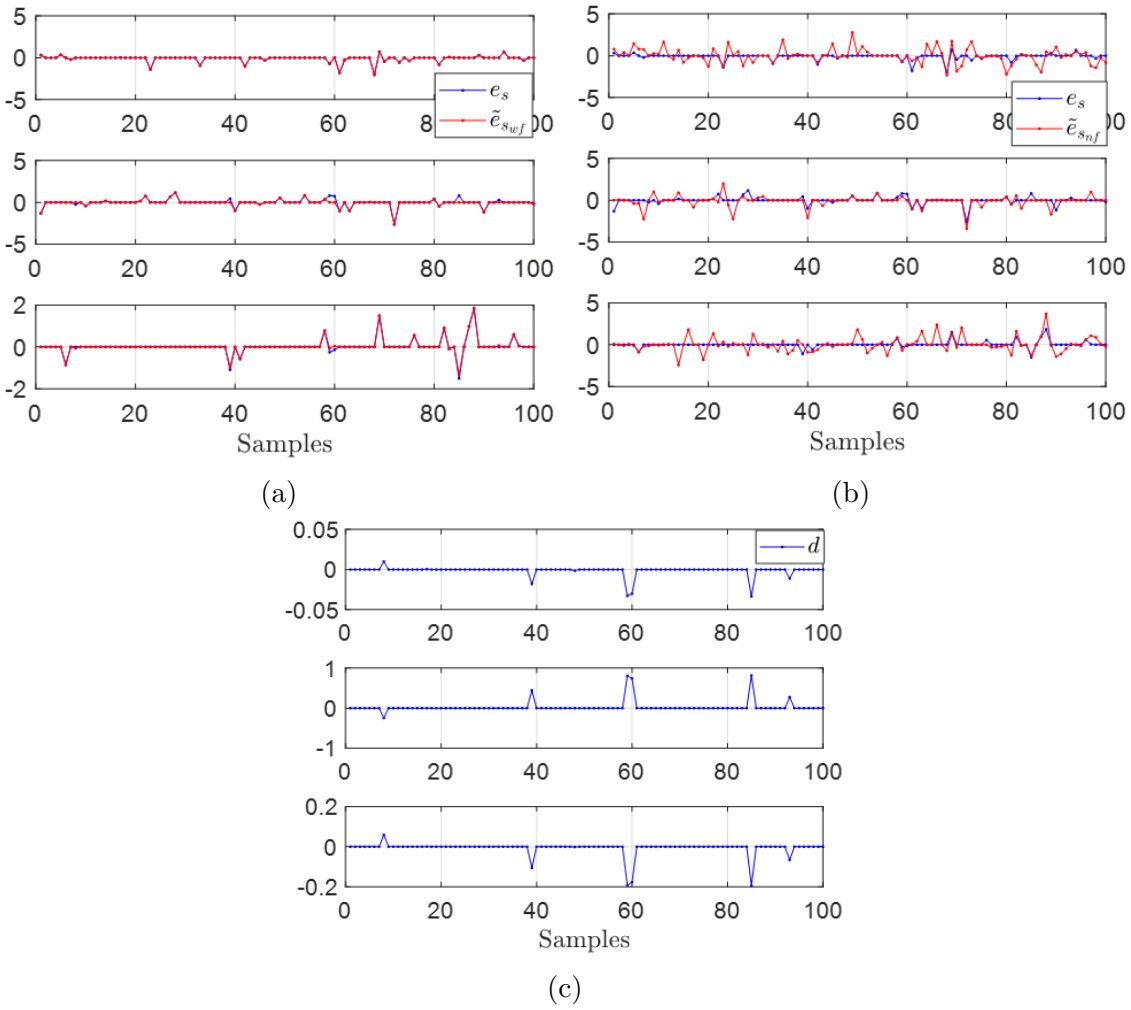


Figure 6.4. Sparse error recovery: (a) Recovery of $\mathrm{e}_s[k]$ by the norm-based approximator with filter; (b) Recovery of $\mathrm{e}_s[k]$ by the norm-based approximator only; and (c) Recovery error of the norm-based approximator with filter.

and the red dot lines are the recovered $\hat{\mathrm{e}}_s[k]$ generated by the norm-based approximator with and without filter, respectively. The blue dot lines in Subfigure (6.4c) show the recovery error $d[k]$ of the norm-based approximator with filter.

Subfigure (6.3a) and Subfigure (6.3b) show the significant state estimation improvement of the combined observer with the sensor faults filter. Subfigure (6.4a) and Subfigure (6.4b) show that the recovery of $\mathrm{e}_s[k]$ using the combined observer with filter is superior over the one using the combined observer only. The reason for this is that the sensor faults $v[k]$

are non-sparse, and without the filter, the 1-norm approximator cannot reliably recover the non-sparse combined error $\mathrm{e}_s[k] + v[k]$.

6.4 Estimation Enhancement with A Bank of UIOs

In this section, we propose a novel state observer architecture consisting of a bank of unknown input observers (UIOs). This novel observer architecture is depicted in Figure 6.5.

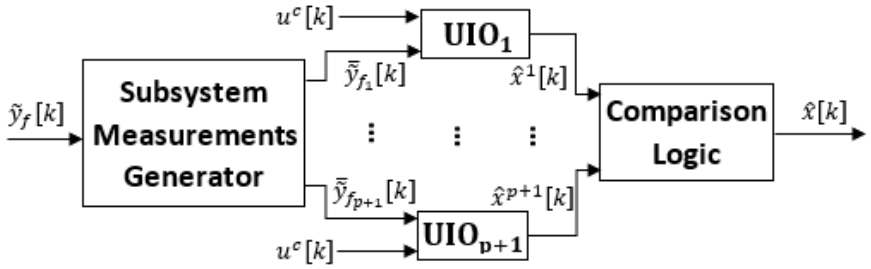


Figure 6.5. A block diagram of the UIO-based state observer.

6.4.1 Output signal accumulation with fictitious measurement

In this subsection, we introduce a fictitious measurement to augment the system. Using the output signal $y^s[k]$, collected by the sensors, we form the fictitious output $y^s_{f\mathrm{ic}}[k] = \sum_{\mathrm{i}=1}^{p} y^s_{\mathrm{i}}[k]$. Note that the fictitious output $y^s_{f\mathrm{ic}}[k]$ is a linear combination of the output sensor measurements. Both the output signal and the fictitious measurement are transmitted to the compensator. Using the fictitious measurement increases the dimension of the vector transmitted to the compensator by one. Also note that by taking the linear combination of the output sensor measurements, we form implicitly the linear combination of the sensor fault signals. To proceed, let $v_{f\mathrm{ic}}[k] = \sum_{\mathrm{i}=1}^{p} v_{\mathrm{i}}[k]$ and $C_{f\mathrm{ic}} = \sum_{\mathrm{i}=1}^{p} C_{(\mathrm{i})}$, where $C_{(\mathrm{i})}$ is the i-th row of the matrix C. Then, we have

$$y^s_f[k] = C_f x[k] + v_f[k],$$

where

$$y_f^s[k] = \begin{bmatrix} y^s[k] \\ y_{f\text{ic}}^s[k] \end{bmatrix}, \; C_f = \begin{bmatrix} C \\ C_{f\text{ic}} \end{bmatrix}, \; v_f[k] = \begin{bmatrix} v[k] \\ v_{f\text{ic}}[k] \end{bmatrix}.$$

The output measurement signal after filtering and received by the compensator has the form

$$\tilde{y}_f^c[k] = C_f x[k] + \mathrm{e}_f[k],$$

where $\tilde{y}_f^c[k] \in \mathbb{R}^{p+1}$, $C_f \in \mathbb{R}^{(p+1)\times n}$, and $\mathrm{e}_f[k] \in \mathbb{R}^{p+1}$ is the sparse malicious attack signal against the accumulated output measurement signal $y_f^s[k]$. Taking into account the augmented output measurement and (6.4), we obtain the following closed-loop CPS model:

$$\begin{aligned} x[k+1] &= Ax[k] + B(u^c[k] + \mathrm{e}_a[k]) \\ \tilde{y}_f^c[k] &= C_f x[k] + \mathrm{e}_f[k]. \end{aligned}$$

The system model after the norm approximation takes the form

$$\left.\begin{aligned} x[k+1] &= Ax[k] + B(u^c[k] + \mathrm{e}_a[k]) \\ \tilde{y}_f[k] &= C_f x[k] + d[k], \end{aligned}\right\} \tag{6.5}$$

where $\tilde{y}_f[k] = \tilde{y}_f^c[k] - \hat{\mathrm{e}}_f[k]$, $d[k] = \mathrm{e}_f[k] - \hat{\mathrm{e}}_f[k]$, and $\hat{\mathrm{e}}_f[k]$ is the estimation of $\mathrm{e}_f[k]$ generated by the norm-based approximator.

Remark 11. *Since $C_{f\text{ic}}$ is a linear combination of the rows of the matrix C, it is easy to check that the CPS model given by (6.4) and the augmented CPS model given by (6.5) have the same norm-based approximator design conditions.*

In the next subsection, we propose a novel state observer architecture consisting of a bank of UIOs for the augmented system given by (6.5).

6.4.2 State observer architecture comprising a bank of UIOs

For $\mathrm{i} = 1, \ldots, p+1$, let $\overline{\tilde{y}}_{f_\mathrm{i}}[k]$ be the vector $\tilde{y}_f[k]$ with the i-th component removed, $\overline{C}_{f_\mathrm{i}}$ be the matrix C_f with the i-th row removed, and $\overline{d}_\mathrm{i}[k]$ the vector $d[k]$ with the i-th component removed. Note that $\overline{\tilde{y}}_{f_\mathrm{i}}[k] \in \mathbb{R}^p$, $\overline{C}_{f_\mathrm{i}} \in \mathbb{R}^{p\times n}$, and $\overline{d}_\mathrm{i}[k] \in \mathbb{R}^p$.

For each $\mathrm{i} = 1, \ldots, p+1$, we use $\overline{\tilde{y}}_{f_\mathrm{i}}[k]$, $\overline{C}_{f_\mathrm{i}}$, and $\overline{d}_\mathrm{i}[k]$ to form a subsystem $\mathcal{S}_\mathrm{i}$ defined by

$$\mathcal{S}_\mathrm{i} \equiv \begin{cases} x[k+1] &= Ax[k] + B(u^c[k] + \mathrm{e}_a[k]) \\ \overline{\tilde{y}}_{f_\mathrm{i}}[k] &= \overline{C}_{f_\mathrm{i}} x[k] + \overline{d}_\mathrm{i}[k]. \end{cases} \tag{6.6}$$

There are $p+1$ subsystems and by its definition, each subsystem $\mathcal{S}_\mathrm{i}$ is insensitive to the disturbance $d_\mathrm{i}[k]$. We construct an UIO for the i-th subsystem using

$$\begin{aligned} z^\mathrm{i}[k+1] =& (I - M_\mathrm{i}\overline{C}_{f_\mathrm{i}}) \left(Az^\mathrm{i}[k] + AM_\mathrm{i}\overline{\tilde{y}}_{f_\mathrm{i}}[k] + Bu[k] \right) \\ &+ L_\mathrm{i} \left(\overline{\tilde{y}}_{f_\mathrm{i}}[k] - \overline{C}_{f_\mathrm{i}} z^\mathrm{i}[k] - \overline{C}_{f_\mathrm{i}} M_\mathrm{i} \overline{\tilde{y}}_{f_\mathrm{i}}[k] \right) \qquad (6.7a) \\ \hat{x}^\mathrm{i}[k] =& z^\mathrm{i}[k] + M_\mathrm{i}\overline{\tilde{y}}_{f_\mathrm{i}}[k], \qquad (6.7b) \end{aligned}$$

where $z^\mathrm{i}[k] \in \mathbb{R}^n$ is the state of the i-th UIO, $\hat{x}^\mathrm{i}[k] \in \mathbb{R}^n$ is the state estimate of (6.6), and $M_\mathrm{i} \in \mathbb{R}^{n\times p}$, $L_\mathrm{i} \in \mathbb{R}^{n\times p}$ are design parameter matrices.

We have the following theorem.

Theorem 7. *Let $A_M = (I - MC)A$. If $\mathrm{rank}\, B = \mathrm{rank}(CB)$ and there exists a matrix $P_\mathcal{O} = P_\mathcal{O}^\top \succ 0$ such that*

$$\begin{bmatrix} -P_\mathcal{O} & \star \\ P_\mathcal{O} A_M - TC & -P_\mathcal{O} \end{bmatrix} \prec 0, \tag{6.8}$$

where $T = P_\mathcal{O} L$, then (6.7) is an effective UIO for the subsystem (6.6) for each $\mathrm{i} = 1, \ldots, p+1$.

Proof. We will verify that each subsystem $\mathcal{S}_\mathrm{i}$ satisfies the standard UIO existence conditions. For $\mathrm{i} = p+1$, we have $\overline{C}_{f_\mathrm{i}} = C$. Therefore, the triple (A, B, C) and the triple $(A, B, \overline{C}_{f_\mathrm{i}})$ share the same UIO existence conditions. We next consider $\mathrm{i} = 1, \ldots, p$. For each $\mathrm{i} = 1, \ldots, p$, let $\overline{C}_\mathrm{i}$ and $\overline{I}_\mathrm{i}$ denote the matrix C and the $p \times p$ identity matrix with their respective i-th rows removed, and let $R_\mathrm{i} = \begin{bmatrix} \overline{I}_\mathrm{i} \\ 1 \cdots 1 \end{bmatrix}$. It is clear from the definition that R_i is $p \times p$ and has full rank. Then for each $\mathrm{i} = 1, \ldots, p$,

$$R_\mathrm{i} C = \begin{bmatrix} \overline{C}_\mathrm{i} \\ \sum_{\mathrm{i}=1}^{p} C_{(\mathrm{i})} \end{bmatrix} = \begin{bmatrix} \overline{C}_\mathrm{i} \\ C_{f\mathrm{i}c} \end{bmatrix} = \overline{C}_{f_\mathrm{i}}.$$

Since $\operatorname{rank}(CB) = \operatorname{rank} B$ and R_{i} has full rank, we have

$$\operatorname{rank}(\overline{C}_{f_{\mathrm{i}}} B) = \operatorname{rank}(R_{\mathrm{i}} CB) = \operatorname{rank}(CB) = \operatorname{rank} B,$$

and thus the rank condition is satisfied. We next verify condition (6.8). We have by assumption that $(I - MC)B = O$. Let $M_{\mathrm{i}} = MR_{\mathrm{i}}^{-1}$. Then

$$\left(I - M_{\mathrm{i}} \overline{C}_{f_{\mathrm{i}}}\right) B = \left(I - MR_{\mathrm{i}}^{-1} R_{\mathrm{i}} C\right) B = (I - MC)B = O.$$

With this M_{i}, we can use $A_{M_{\mathrm{i}}} = \left(I - M_{\mathrm{i}} \overline{C}_{f_{\mathrm{i}}}\right) A$ in condition (6.8). Then

$$A_{M_{\mathrm{i}}} = A_M,$$

and we can take $P_{\mathcal{O}_{\mathrm{i}}} = P_{\mathcal{O}}$ and $T_{\mathrm{i}} = TR_{\mathrm{i}}^{-1}$ in (6.8). Thus (6.8) for M_{i} and (6.8) for M are identical and the proof is complete. □

6.4.3 Comparison logic algorithm

In this subsection, we propose a comparison logic algorithm that compares the state estimates from the $p + 1$ UIOs. The objective is to find the state estimate, $\hat{x}^{\mathrm{i}}[k]$ that is insensitive to $d_{\mathrm{i}}[k]$. We then choose $\hat{x}^{\mathrm{i}}[k]$ as our state estimate of $x[k]$.

The UIO given by (6.7) for the i-th subsystem (6.6) is insensitive to $d_{\mathrm{i}}[k]$. The state estimate $\hat{x}^{\mathrm{i}}[k]$ from the i-th UIO is the correct estimate of the state $x[k]$. Note that the state estimates from the other p UIOs are affected by $d_{\mathrm{i}}[k]$. We use the state estimates from all $p+1$ UIOs to determine the state estimate $\hat{x}[k]$ as follows, where $\zeta > 0$ is a selected threshold and $\mathrm{i}, \mathrm{j} = 1, \ldots, p + 1$:

1. If there exists i such that $\|x^{\mathrm{i}}[k] - x^{\mathrm{j}}[k]\| > \zeta$ for all $\mathrm{j} \neq \mathrm{i}$, we let

$$\hat{x}[k] = \hat{x}^{\mathrm{i}}[k].$$

2. Otherwise, we let

$$\hat{x}[k] = \frac{1}{p+1} \sum_{\mathrm{i}=1}^{p+1} x^{\mathrm{i}}[k].$$

Algorithm 2 summarizes our discussion of the state observer design for the CPS shown in Figure 6.1. An illustration of the algorithm will be given in the next subsection.

Algorithm 2: State observer design using a bank of UIOs

1 Discretize the CT system given by (6.1) using the exact discretization method to obtain the DT system triple (A, B, C)
2 Check if the pair (A, B) is reachable and the pair (A, C) is observable; If not, STOP
3 Check that the conditions of Theorem 7 are satisfied. If not, STOP
4 Design the sensor and actuator fault filters using the method discussed in Section 6.3
5 Augment the sensor output measurement $y^s[k]$ with the fictitious output measurement $y^s_{f\mathrm{ic}}[k]$
6 Compute $\hat{\mathrm{e}}_f[k]$ and obtain the augmented system given by (6.5)
7 Compute $M_{\mathrm{i}} = B\left(\overline{C}_{f_{\mathrm{i}}} B\right)^{\dagger}$ and solve the LMI given by (6.8)
8 Calculate $x^{\mathrm{i}}[k]$ using (6.7b) and compare $x^{\mathrm{i}}[k]$ with $x^{\mathrm{j}}[k]$ using the comparison logic algorithm from Subsection 6.4.3
9 Find the state estimate $\hat{x}[k]$.

6.4.4 An example illustrating the enhancement of the combined observer performance using a bank of UIOs

To illustrate the state estimation improvement using the bank of UIOs, we compare the state estimation results from standard UIO with the ones using a bank of UIOs. In both simulations, we choose the same plant and the same parameters as discussed in Subsection 6.3.3. The simulation results are shown in Subfigures (6.3a) and (6.3c). Comparing Subfigure (6.3c) with Subfigure (6.3a), we can see the state estimation enhancement when using a bank of UIOs. To compare the performance of the observers quantitatively, we evaluate the performance indices

$$J_{\mathrm{e}} \quad = \quad \textstyle\sum_{k=10}^{100} \|x_{\mathrm{i}}[k] - \hat{x}_{\mathrm{i}}[k]\|_1, \tag{6.9}$$

where i $= 1, \ldots, n$. The values of the indices for the simulations in Subfigure (6.3a) and Subfigure (6.3c) are:

$$J_{\mathrm{e}_{UIO}} = \begin{bmatrix} 0.980 \\ 26.028 \\ 9.668 \\ 176.360 \end{bmatrix}, \; J_{\mathrm{e}_{bUIOs}} = \begin{bmatrix} 0.013 \\ 0.045 \\ 0.125 \\ 0.479 \end{bmatrix}.$$

These results illustrate the superiority of the observer using the bank of UIOs over the standard UIO-based observer.

In summary, the performance of the state observer is based on the sparsity of malicious attack vectors e_s and e_a. If s_{e_s} and s_{e_a} are sufficiently large, then $\hat{x}[k]$ is expected to be the correct estimate of $x[k]$. In general, however, we have

$$\mathrm{e}[k] = x[k] - \hat{x}[k] \neq 0. \tag{6.10}$$

In the next section, we propose a model-reference controller for the CPS considered in this chapter.

6.5 Model Reference Controller Design

In this section, we propose a controller design method for system (6.5) so that its state tracks the state of a given model-reference signal generator.

6.5.1 Controller design

The model-reference signal generator is described by

$$x^r[k+1] = A_r x^r[k] + B_r r[k], \tag{6.11}$$

where $A_r = A - BK_r$, $B_r = B$, $K_r \in \mathbb{R}^{m \times n}$ is a gain matrix, and $r[k] \in \mathbb{R}^m$ is a command signal that is chosen so that $x^r[k]$ represents a desired state trajectory of the plant. The

tracking error is given by $\mathrm{e}_r[k] = x[k] - x^r[k]$. Using the state equation of (6.5) and the fact that $u^c[k] = -u^r[k] + r[k]$ (see Figure 6.1), we obtain

$$\begin{aligned}
\mathrm{e}_r[k+1] &= x[k+1] - x^r[k+1] \\
&= Ax[k] + B\left(u^c[k] + \mathrm{e}_a[k]\right) - A_r x^r[k] - Br[k] \\
&= A_r \mathrm{e}_r[k] + B\left(K_r x[k] + u^c[k] + \mathrm{e}_a[k] - r[k]\right) \\
&= A_r \mathrm{e}_r[k] + B\left(K_r x[k] - u^r[k] + \mathrm{e}_a[k]\right).
\end{aligned}$$

In our design, we take $u^r[k] = K_r \hat{x}[k]$, which when substituted along with (6.10) into the above gives

$$\begin{aligned}
\mathrm{e}_r[k+1] &= A_r \mathrm{e}_r[k] + B\left(K_r \mathrm{e}[k] + \mathrm{e}_a[k]\right) \\
&= A_r \mathrm{e}_r[k] + B\xi[k],
\end{aligned} \tag{6.12}$$

where $\xi[k] = K_r \mathrm{e}[k] + \mathrm{e}_a[k]$. We need the following definition from [65].

Definition 5. *The system $\mathrm{e}_r[k+1] = f(k, \mathrm{e}_r[k], \xi[k])$ is globally uniformly l_∞-stable with performance level γ if the following conditions are satisfied:*

1. *If $\xi[k] = 0$ for all k, then the undisturbed system $\mathrm{e}_r[k+1] = f(k, \mathrm{e}_r[k])$ is globally uniformly exponentially stable with respect to the origin.*

2. *For zero initial condition, $\mathrm{e}_r[0] = 0$, and every bounded unknown input $\xi[k]$, we have $\|\mathrm{e}_r[k]\| \leq \gamma \|\xi[k]\|_\infty$.*

3. *For every initial condition and every bounded unknown input, we have*

$$\limsup_{k \to \infty} \|\mathrm{e}_r[k]\| \leq \gamma \|\xi[k]\|_\infty.$$

We have the following lemmas.

Lemma 17. *If A_r is Schur stable and that either condition 2) or condition 3) of Definition 5) is satisfied, then the tracking error dynamics given by (6.12) is globally uniformly l_∞-stable with performance level γ.*

Proof. It is easy to check that conditions 2) and 3) are equivalent for linear systems. Since A_r is Schur stable by assumption, the error dynamics (6.12) are globally uniformly l_∞-stable with performance level γ. □

Lemma 18. *Suppose, for the error governed by equation (6.12), there exist a continuous function $V : \mathbb{R}^n \to \mathbb{R}$ and scalars $\delta \in (0,1)$, ω_1, $\omega_2 > 0$, and $\lambda \geq 0$ such that*

$$\omega_1 \|\mathrm{e}_r[k]\|^2 \leq V(\mathrm{e}_r[k]) \leq \omega_2 \|\mathrm{e}_r[k]\|^2 \tag{6.13}$$

and

$$\Delta V[k] \leq -\delta \left(V(\mathrm{e}_r[k]) - \lambda \|\xi[k]\|^2 \right) \tag{6.14}$$

for all k, where $\Delta V[k] = V(\mathrm{e}_r[k+1]) - V(\mathrm{e}_r[k])$. Then the error is globally uniformly l_∞-stable with performance level $\gamma = \sqrt{\lambda/\omega_1}$ with respect to disturbance sequence $\xi[k]$.

Proof. See [76]. □

Using the above lemmas, we can show that if there exist a matrix $P_{\mathcal{C}} = P_{\mathcal{C}}^\top \succ 0$ and a scalar $\delta \in (0,1)$ such that

$$\begin{bmatrix} A_r^\top P_{\mathcal{C}} A_r - (1-\delta) P_{\mathcal{C}} & \star \\ B^\top P_{\mathcal{C}} A_r & B^\top P_{\mathcal{C}} B - \delta I \end{bmatrix} \preceq 0, \tag{6.15}$$

then the tracking error $\mathrm{e}_r[k]$ is l_∞-stable with performance level $\gamma = 1/\sqrt{\lambda_{\min}(P_{\mathcal{C}})}$.

6.5.2 Solving for the controller gain matrix

We now present a method to solve matrix inequality (6.15). Let $\tilde{P}_{\mathcal{C}} = P_{\mathcal{C}}^{-1}$. Then using standard arguments, we can show that (6.15) is equivalent to the linear matrix inequality (LMI)

$$\begin{bmatrix} \tilde{P}_{\mathcal{C}} & \star & \star \\ \tilde{P}_{\mathcal{C}} A_r^\top & (1-\delta)\tilde{P}_{\mathcal{C}} & \star \\ B^\top & O_{m\times n} & \delta I \end{bmatrix} \succeq 0. \tag{6.16}$$

Indeed, taking the Schur complement of $\tilde{P}_{\mathcal{C}}$, we obtain

$$\begin{bmatrix} (1-\delta)\tilde{P}_{\mathcal{C}} & \star \\ O_{m\times n} & \delta I \end{bmatrix} - \begin{bmatrix} \tilde{P}_{\mathcal{C}} A_r^\top \\ B^\top \end{bmatrix} \tilde{P}_{\mathcal{C}}^{-1} \begin{bmatrix} A_r \tilde{P}_{\mathcal{C}} & B \end{bmatrix}$$
$$= \begin{bmatrix} (1-\delta)\tilde{P}_{\mathcal{C}} - \tilde{P}_{\mathcal{C}} A_r^\top P_{\mathcal{C}} A_r \tilde{P}_{\mathcal{C}} & \star \\ -B^\top P_{\mathcal{C}} A_r \tilde{P}_{\mathcal{C}} & \delta I - B^\top P_{\mathcal{C}} B \end{bmatrix} \succeq 0.$$

Let $\Xi_2 = \begin{bmatrix} P_{\mathcal{C}} & O_{n\times m} \\ O_{m\times n} & I \end{bmatrix}$. We then have $\Xi_2 = \Xi_2^\top \neq O$. Premultiplying and postmultiplying the above inequality by $\Xi_2^\top$ and Ξ_2, respectively, gives

$$\begin{bmatrix} (1-\delta)P_{\mathcal{C}} - A_r^\top P_{\mathcal{C}} A_r & \star \\ -B^\top P_{\mathcal{C}} A_r & \delta I - B^\top P_{\mathcal{C}} B \end{bmatrix} \succeq 0,$$

which is equivalent to (6.15). We now substitute $A_r = A - BK_r$ into (6.16) to obtain

$$\begin{bmatrix} \tilde{P}_{\mathcal{C}} & \star & \star \\ \tilde{P}_{\mathcal{C}} A^\top - Z^\top B^\top & (1-\delta)\tilde{P}_{\mathcal{C}} & \star \\ B^\top & O_{m\times n} & \delta I \end{bmatrix} \succeq 0, \tag{6.17}$$

where $Z = K_r \tilde{P}_{\mathcal{C}}$. Since (6.17) is an LMI, the parameter matrices $\tilde{P}_C$ and Z can be obtained by solving (6.17) using any standard LMI toolbox. We then calculate the controller gain matrix, $K_r = Z\tilde{P}_{\mathcal{C}}^{-1}$.

We summarize the above discussion of the robust model reference controller design in the following algorithm.

Remark 12. *A possible method for computing the parameter δ_{i} in each line search iteration is a bisection method presented, for example, in [77, p. 209].*

6.6 Application to Self-Driving Vehicle

In this section, we apply the proposed controller-observer compensator to a self-driving vehicle. We first discuss the modeling of the plant of the ground vehicle and then analyze

Algorithm 3: Controller design

1 Set $\mathrm{i} = 1$
2 Construct the parameter δ_{i} using any 1-D line search and substitute δ_{i} into (6.17)
3 Compute parameter matrices $\tilde{P}_{\mathcal{C}}$ and Z by solving (6.17)
4 If there are no feasible solutions, set $\mathrm{i} = \mathrm{i} + 1$ and go to STEP 2; If there is a feasible solution, compute $K_r = Z\tilde{P}_{\mathcal{C}}^{-1}$, NEXT
5 Construct the control law, $u^c[k] = -K_r\hat{x}[k] + r[k]$, where $\hat{x}[k]$ is the CPS state estimate obtained from the observer constructed using Algorithm 2.

the controller performance in the presence of sensor and actuator faults and sparse malicious attacks. Our objective is to control the ground vehicle to follow a desired reference trajectory.

6.6.1 Ground vehicle model

In this subsection, we discuss the dynamics of a ground vehicle that we will use as a dynamical plant of the CPS described in Subsection 6.2.1. The dynamical model that we consider is known in the literature as the bicycle model. Our modeling process follows that of [45, p. 22] and of [78].

We use the Society of Automotive Engineering (SAE) standard coordinate system for the body-fixed coordinate system of the ground vehicle as shown in Fig. 6.6. The nomenclature

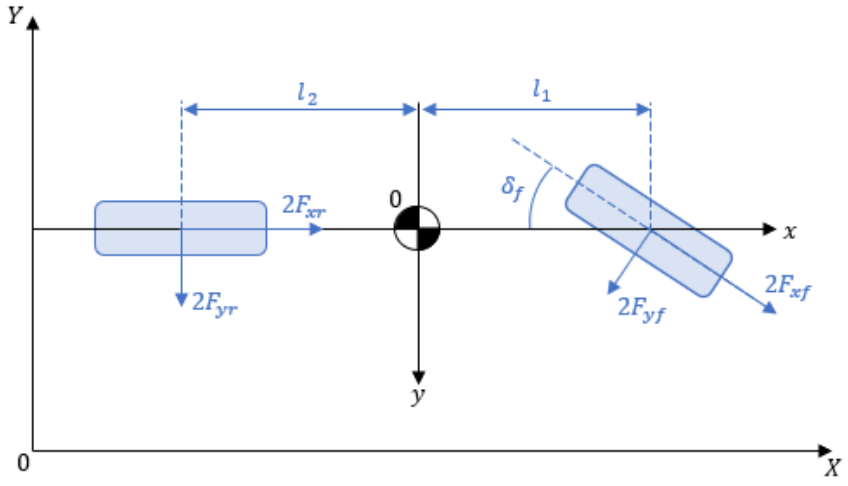


Figure 6.6. Free-body diagram of bicycle model in body-fixed coordinates.

used in the model description is given in Table 6.1.

Table 6.1. Parameters and nomenclature for the bicycle model.

Parameter	Definition	Unit
v_x	Longitudinal velocity	m/s
Y	Global position	m
ψ	Yaw angle	rad
m	Vehicle mass	kg
I_z	Mass moment of inertia	$\mathrm{kg}\cdot\mathrm{m}^2$
l_1	Distance from CG to front axle	m
l_2	Distance from CG to rear axle	m
F_f, F_r	Front and rear tire force	N
$C_{\alpha f}$, $C_{\alpha r}$	Front and rear tire stiffness	N/rad
δ_f	Front steering angle	rad

We make the following simplifying assumptions:

1. The vehicle is symmetrical along its longitudinal axis;
2. No motion exists in the roll and pitch directions;
3. The vehicle is steered by the front wheels;
4. The vehicle has a constant longitudinal velocity, v_x.

Following [45], we choose the state of the vehicle dynamical model as $x = \begin{bmatrix} \dot{y} & \theta & \omega_z & Y \end{bmatrix}^\top$, where $\dot{y}$ is the car velocity along the lateral direction y out of the right-hand side of the vehicle, $\theta = \psi$, and ω_z is the yaw angular velocity. The vehicle modeling equations in a state-space format can then be represented as (6.1), where

$$A_c = \begin{bmatrix} -\frac{2C_{\alpha f}+2C_{\alpha r}}{mv_x} & 0 & -v_x - \frac{2C_{\alpha f}l_1 - 2C_{\alpha r}l_2}{mv_x} & 0 \\ 0 & 0 & 1 & 0 \\ -\frac{2C_{\alpha f}l_1 - 2C_{\alpha r}l_2}{I_z v_x} & 0 & -\frac{2l_1^2 C_{\alpha f} + 2l_2^2 C_{\alpha r}}{I_z v_x} & 0 \\ -1 & -v_x & 0 & 0 \end{bmatrix}, \; B_c = \begin{bmatrix} \frac{2C_{\alpha f}}{m} \\ 0 \\ \frac{2l_1 C_{\alpha f}}{I_z} \\ 0 \end{bmatrix}, \; C_c = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

In our simulations, we use the following vehicle parameters: $C_{\alpha f} = C_{\alpha r} = 30000.00\,\mathrm{N/rad}$, $l_1 = 1.20\,\mathrm{m}$, $l_2 = 1.22\,\mathrm{m}$, $m = 1280\,\mathrm{kg}$, $I_z = 2500\,\mathrm{kg}\cdot\mathrm{m}^2$, and $v_x = 5\,\mathrm{m/sec}$. Note that the system is controllable and observable but not stable.

6.6.2 Simulations

In our simulations, we use the sampling time period $T_s = 0.05\,\text{sec}$ to discretize the plant model with the exact discretization method. Following Algorithm 3, we compute the controller gain,

$$K_r = \begin{bmatrix} 0.49 & 3.92 & -0.06 & -8.46 \end{bmatrix}.$$

We simulate two scenarios. We first consider a self-driving vehicle moving along the X axis with a constant longitudinal speed $v_x = 5\,\text{m/s}$. We compare the performance of the proposed closed-loop controller-observer compensator against the open-loop strategy. In the open-loop strategy, we set $u^c[k] = r[k] = 0$ for all samples k. We assume that each component of the sensor output, $y^s[k]$, and the actuator input, $u^a[k]$, has 0.7 probability of being contaminated by random sensor and actuator faults. The faults are uniformly distributed with values in the interval $[0, 1]$. We assume that each element of $\mathrm{e}_s[k]$ and $\mathrm{e}_a[k]$ has 0.15 and 0.05 probability, respectively, of being nonzero. The sparse attacks are assumed to be normally distributed with mean 0 and variance 1.

In Subfigure (6.7a), the blue dot lines depict the true states of the plant and the red dot lines show the estimated states. In this simulation, the moving vehicle is controlled by the proposed controller-observer compensator. In Subfigure (6.7b), the blue dot lines show the true states of the plant controlled by the open-loop strategy. The blue dot line in the top subplot in (6.7c) shows the sparse attack signal $\mathrm{e}_a[k]$, and the bottom subplot shows the reference control signal, $r[k] = 0$. The desired state of this driving scenario is $x^r[k] = \begin{bmatrix} 0 & 0 & 0 & 0 \end{bmatrix}^\top$ at all samples k. Since the plant is unstable, we can see from Subfigures (6.7b) and (6.7c) that when the reference control signal is corrupted by sparse malicious attacks at sample $k = 38$ and sample $k = 90$, the state of the plant operating in the open loop mode is unable to follow the desired state. In Subfigure (6.7a), we can see that when the closed-loop control strategy is applied, the plant state follows desired state after a few samples in the presence of sparse attacks. These simulations illustrate the expected performance of the proposed controller as discussed in Subsection 6.5.1.

We next simulate the case when the vehicle is to follow a more complex path shown in Figure 6.8. The blue dot line in Figure 6.8 shows the desired trajectory of the self-driving

vehicle. We first let the simulated vehicle move along the desired trajectory and collect the reference signal $r[k]$ at every sample k. We then simulate the vehicle motion in the CPS environment as in the previous case when the vehicle is driven by the proposed controller-observer compensator. The simulation results are shown in Figure 6.9. As can be seen, the proposed closed-loop controller-observer compensator performs as desired in the presence of sensor and actuator faults and sparse malicious attacks.

6.7 Conclusion

In this chapter, we are inspired by ideas from the theory of error correcting codes to propose a repetition scheme to compare different copies of the actuator and sensor signals to eliminate errors from sensor and actuator faults. The advantage of this scheme is that it is simple and quite effective. However it requires a bank of identical sensors which in some cases may be prohibitively expensive. This issue calls for further investigation.

An open problem is to investigate the performance of the proposed controller-observer compensator on a CPS with more detailed model with modeling uncertainties.

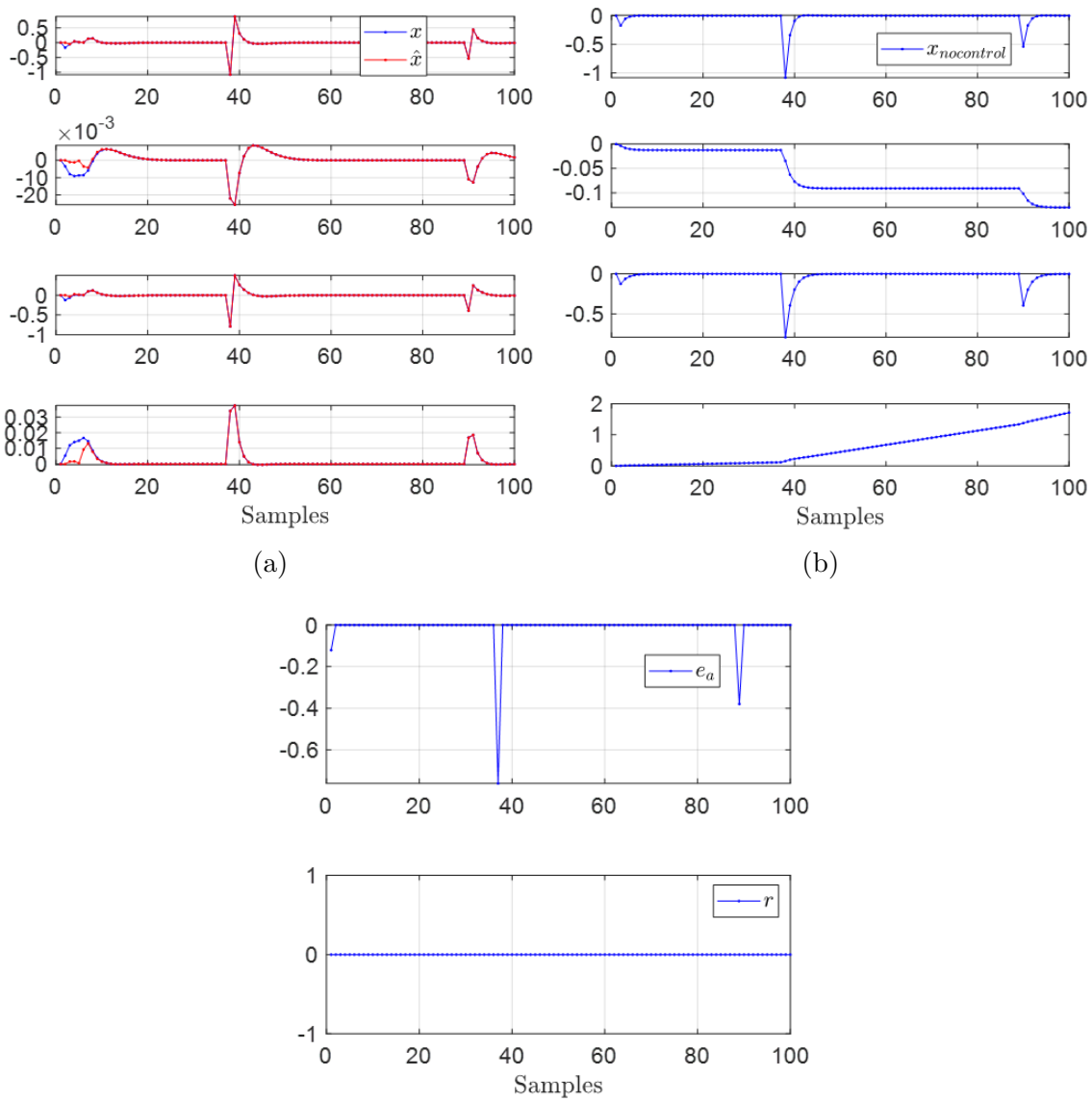


Figure 6.7. (a) State and state estimates of the moving vehicle driven by the proposed robust controller-observer compensator; (b) State of the self-driving vehicle in the open-loop mode; (c) Sparse malicious attack signal and the reference signal.

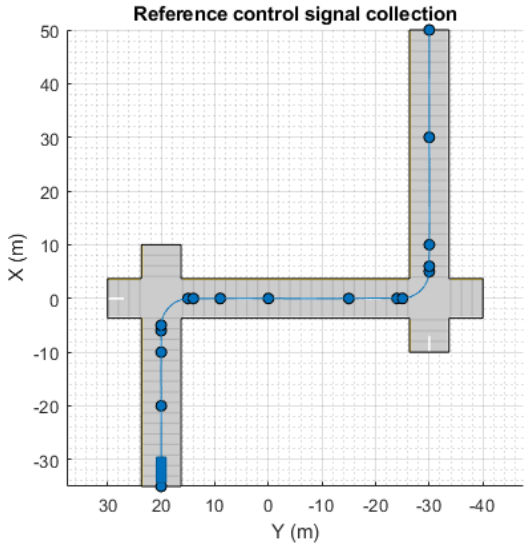
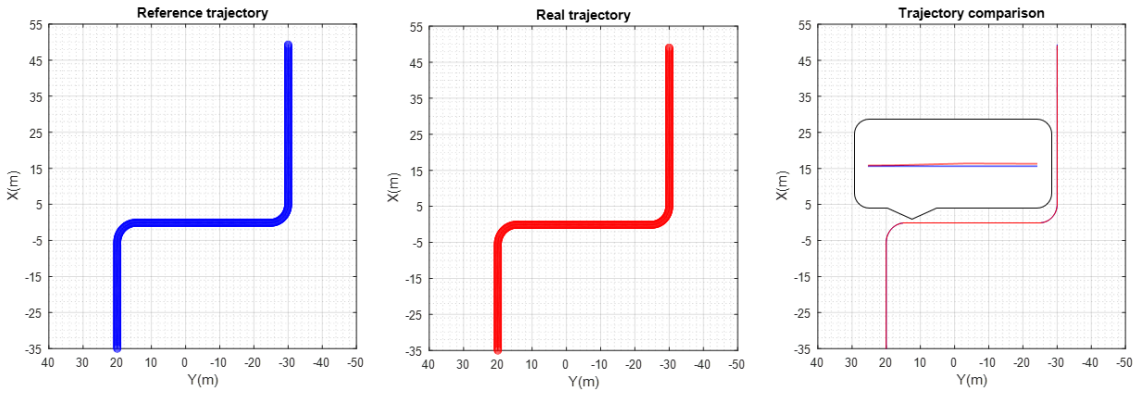


Figure 6.8. A driving scenario in the second simulation.



(a) (b) (c)

Figure 6.9. (a) The desired reference trajectory; (b) The actual trajectory of the self-driving car driven by the proposed robust controller-observer compensator; (c) Comparison between the reference trajectory and the actual trajectory.

7. SUMMARY AND OPEN PROBLEMS

7.1 Summary

In this thesis, we propose methods for simultaneously estimating the state, the sparse attacks, and the disturbance of a CPS under sparse adversarial attacks and subjected to arbitrary disturbance.

In Chapter 2, we analyze a scenario where the output measurements of the DT linear CPS are subjected to sparse malicious packet drop attacks. A sparse error recovery method was proposed to overcome the problem.

In Chapter 3, a general case of a CPS under malicious attacks is considered. That is, sparse malicious packet drop attacks corrupt both control signals and output measurements of the DT linear CPS. A novel DT observer-controller compensator architecture is proposed.

In Chapter 4, we propose a method of applying the DT observer-controller compensator to a CT nonlinear CPS. We give an example of systems showing the continuous-discrete UIO existence dichotomy.

In Chapter 5, we apply the proposed observer to a CPS under sparse adversarial attacks and subjected to arbitrary disturbance. A novel design method is proposed that uses fictitious output measurements to enhance the observer's performance.

In Chapter 6, a controller-observer compensator is proposed for CPSs with sensor and actuator faults and unsecured communication networks. Sensor and actuator fault filters are proposed that use ECC approach to enhance the proposed observer's performance. A model reference controller with a performance level that can be calculated is given.

7.2 Open Problems

In this section, we describe methods of applying DT decentralized combined observer-controller compensators to different types of CT decentralized cyber-physical systems. In the design of the compensators for these systems, we propose to use the design methods presented in this thesis.

7.2.1 Review of different types of decentralization methods

A large-scale CPS can be composed of several smaller interconnected units. Many modern large-scale cyber-physical systems consist of several subsystems coupled through their dynamics, controllers, or performance objectives. For examples, aircraft, satellite, and mobile robot formations [79]–[81]; automated highways and other shared infrastructures [82], [83]; flexible structures [84] and supply chains [85], [86]. When regulating these systems, it is often advantageous to adopt a decentralized control architecture in which the overall controller is composed of interconnected sub-controllers, each of which accesses a subset of the plant's state measurements. In the following, we discuss different types of decentralization.

Completely overlapping decentralization

We begin by considering an unstructured large-scale linear time-invariant (LTI) system model,

$$\left.\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t) \\ y(t) &= Cx(t), \end{aligned}\right\} \tag{7.1}$$

where $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$, and $C \in \mathbb{R}^{p\times n}$. Following [87], we apply the so called completely overlapping decomposition to system (7.1) using the following steps:

- Decompose the input-output pair (u, y) into N input-output sub-pairs $(u_{\mathrm{i}}, y_{\mathrm{i}})$ such that (i) the coupling between u_{i} and y_{i} is maximized, for all $\mathrm{i} = 1, \ldots, N$; (ii) the coupling between u_{j} and y_{i} is minimized, $\mathrm{i}, \mathrm{j} = 1, \ldots, \mathrm{j} \neq \mathrm{i}$.

- Decompose matrices B and C according to the partition obtained on u and y, respectively. Specifically, $B = [B_1 \;\; B_2 \;\; \cdots \;\; B_N]$ and $C = [C_1^\top \;\; C_2^\top \;\; \cdots \;\; C_N^\top]^\top$. Hence, $Bu = \sum_{\mathrm{i}=\mathrm{i}}^{N} B_{\mathrm{i}} u_{\mathrm{i}}$ and $y_{\mathrm{i}} = C_{\mathrm{i}} x$ for all i.

The completely overlapping decentralization of (7.1) can be presented as

$$\begin{cases} \dot{x}(t) &= Ax(t) + \sum_{\mathrm{i}=1}^{N} B_{\mathrm{i}} u_{\mathrm{i}}(t) \\ y_{\mathrm{i}}(t) &= C_{\mathrm{i}} x(t), \qquad \mathrm{i} = 1, 2, \cdots, N, \end{cases} \tag{7.2}$$

where $B_{\mathrm{i}} \in \mathbb{R}^{n \times m_{\mathrm{i}}}$, and $C_{\mathrm{i}} \in \mathbb{R}^{p_{\mathrm{i}} \times n}$. Note that $p = \sum_{\mathrm{i}=1}^{N} p_{\mathrm{i}}$, $m = \sum_{\mathrm{i}=1}^{N} m_{\mathrm{i}}$, $B = [B_1 \ B_2 \ \cdots \ B_N]$, and $C = [C_1^\top \ C_2^\top \ \cdots \ C_N^\top]^\top$.

The completely overlapping decentralized system can be regarded as a N-channel interconnected system with the input $u_{\mathrm{i}}(t)$ and the output $y_{\mathrm{i}}(t)$ for its i-th control channel. We use the following example to show the application of this type of decentralization.

Example 3. Consider the mass-spring system from [88] shown in Figure 7.1. We view this

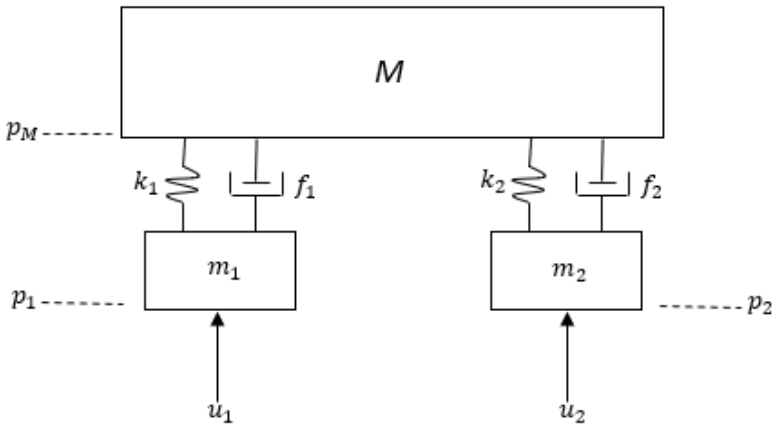


Figure 7.1. An example of a completely overlapping decentralized system. The example is taken from [88].

system as a two-channel interconnected system with the input $u_{\mathrm{i}}(t)$ and the output $y_{\mathrm{i}}(t)$ for its i-th control channel, where $\mathrm{i} = 1, 2$. The state of the system is defined as

$$x(t) = [p_M(t) \ \ \dot{p}_M(t) \ \ p_1(t) \ \ \dot{p}_1(t) \ \ p_2(t) \ \ \dot{p}_2(t)]^\top,$$

where p_M, p_1, and p_2 are the positions of masses M, m_1, and m_2, respectively. f_1 and f_2 are the damping force parameters, k_1 and k_2 are the spring constants, and u_1 and u_2 are the input forces.

The system modeling equations are:

$$\begin{cases} \dot{p}_M &= \dot{p}_M \\ M\ddot{p}_M &= -(k_1+k_2)p_M - (f_1+f_2)\dot{p}_M + k_1 p_1 + f_1\dot{p}_1 + k_2 p_2 + f_2\dot{p}_2 \\ \dot{p}_1 &= \dot{p}_1 \\ m_1\ddot{p}_1 &= k_1 p_M + f_1\dot{p}_M - k_1 p_1 - f_1\dot{p}_1 + u_1 \\ \dot{p}_2 &= \dot{p}_2 \\ m_2\ddot{p}_2 &= k_2 p_M + f_2\dot{p}_M - k_2 p_2 - f_2\dot{p}_2 + u_2. \end{cases}$$

Let $m_1 = m_2 = 1$ kg, $M = 10$ kg, $k_1 = k_2 = 1$ N/m, $f_1 = f_2 = 0.1$ N·s/m. Choose p_1 and p_2 as the outputs. The state space of the system has the form (7.2), where

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ -0.2 & -0.02 & 0.1 & 0.01 & 0.1 & 0.01 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0.1 & -1 & -0.1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0.1 & 0 & 0 & -1 & -0.1 \end{bmatrix},$$

$$B_1 = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix}^\top, \quad B_2 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}^\top,$$

$$C_1 = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}, \quad C_2 = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}.$$

Non-overlapping decentralization

Following [87, Chapter 8], we apply the so called non-overlapping decomposition method to the unstructured LTI system model given by (7.1). Let $x_\mathrm{i} \in \mathbb{R}^{n_\mathrm{i}}$ be a non-overlapping partition of x (possibly under a suitable reordering of the state variables), i.e., $x = \begin{bmatrix} x_1 & \cdots & x_N \end{bmatrix}^\top$

with $\sum_{\mathrm{i}=1}^{N} n_{\mathrm{i}} = n$ and where x_{i} defines the state of subsystem $\mathcal{S}_{\mathrm{i}}$. The unstructured model takes the form:

$$\begin{bmatrix} \dot{x}_1 \\ \vdots \\ \dot{x}_N \end{bmatrix} = \begin{bmatrix} A_{11} & \cdots & A_{1N} \\ \vdots & \ddots & \vdots \\ A_{N1} & \cdots & A_{NN} \end{bmatrix} \begin{bmatrix} x_1 \\ \vdots \\ x_N \end{bmatrix} + \begin{bmatrix} B_{11} & \cdots & B_{1N} \\ \vdots & \ddots & \vdots \\ B_{N1} & \cdots & B_{NN} \end{bmatrix} \begin{bmatrix} u_1 \\ \vdots \\ u_N \end{bmatrix}$$
$$\begin{bmatrix} y_1 \\ \vdots \\ y_N \end{bmatrix} = \begin{bmatrix} C_{11} & \cdots & C_{1N} \\ \vdots & \ddots & \vdots \\ C_{N1} & \cdots & C_{NN} \end{bmatrix} \begin{bmatrix} x_1 \\ \vdots \\ x_N \end{bmatrix}.$$

The model equation of subsystem $\mathcal{S}_{\mathrm{i}}$ is

$$\begin{cases} \dot{x}_{\mathrm{i}} &= A_{\mathrm{ii}} x_{\mathrm{i}} + B_{\mathrm{ii}} u_{\mathrm{i}} + \sum_{\mathrm{i}\neq\mathrm{j}} (A_{\mathrm{ij}} x + B_{\mathrm{ij}} u_{\mathrm{j}}) \\ y_{\mathrm{i}} &= C_{\mathrm{ii}} x_{\mathrm{i}} + \sum_{\mathrm{i}\neq\mathrm{j}} C_{\mathrm{ij}} x_{\mathrm{j}}. \end{cases}$$

The non-overlapping decentralized system can be regarded as an interaction of N subsystems $\mathcal{S}_{\mathrm{i}}$, $\mathrm{i} = 1, \ldots, N$. We use the following example to illustrate the application of this type of decentralization.

Example 4. Consider the system from [89] shown in Figure 7.2, consisting of a cascade interconnection of three water tanks. The modeling equations are:

$$\begin{cases} \dot{x}_1 &= k_3\sqrt{x_3} - k_1\sqrt{x_1} \\ \dot{x}_2 &= -k_2\sqrt{x_2} + u_2 \\ \dot{x}_3 &= k_2\sqrt{x_2} - k_3\sqrt{x_3} + u_1 \\ y_1 &= x_2 \\ y_2 &= x_1, \end{cases}$$

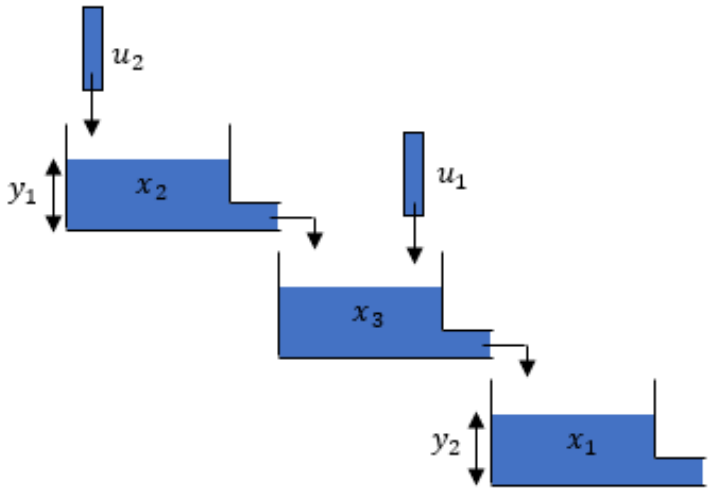


Figure 7.2. An example of a non-overlapping decentralized system.

where x_{i}, $\mathrm{i} = 1, 2, 3$ is the water level; u_1 and u_2 are the input water volume flows; $k_1 = k_2 = k_3 = 2$. The linearized model about the equilibrium pair $(x_{\mathrm{e}}, u_{\mathrm{e}})$, where $x_{\mathrm{e}} = [1 \;\; 1 \;\; 1]^\top$ and $u_{\mathrm{e}} = [0 \;\; 2]^\top$, has the form,

$$A = \begin{bmatrix} -1 & 0 & 1 \\ 0 & -1 & 0 \\ 0 & -1 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 0 & 0 \\ 0 & 1 \\ 1 & 0 \end{bmatrix}, \quad C = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix}.$$

Using the non-overlapping decomposition method, the above unstructured linearized system can be partitioned into two interconnected subsystems as

$$\mathcal{S}_1 : \begin{cases} \dot{x}_2 &= -x_2 + u_2 \\ y_1 &= x_2, \end{cases}$$

$$\mathcal{S}_2 : \begin{cases} \begin{bmatrix} \dot{x}_1 \\ \dot{x}_3 \end{bmatrix} &= \begin{bmatrix} -1 & 1 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_3 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} x_2 + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u_1 \\ y_2 &= \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_3 \end{bmatrix}. \end{cases}$$

7.2.2 Secure state estimation and control using non-overlapping decentralized CPS model

In this subsection, we propose to design a class of observer-controller compensators for secrue state estimation and control of N interconnected cyber-physical systems under non-overlapping decentralization form.

Problem statement and preliminary analysis

Consider N interconnected physical systems are remotely controlled as N cyber-physical systems as shown in Figure 7.3.

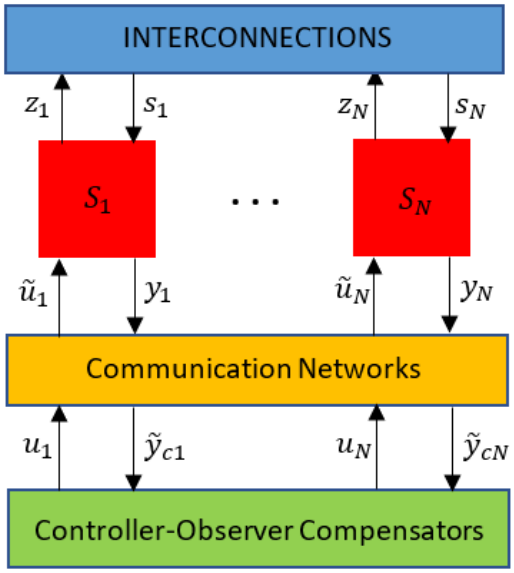


Figure 7.3. Non-overlapping interconnected decentralized CPS.

The dynamic model for system $\mathcal{S}_\mathrm{i}$ is:

$$\mathcal{S}_\mathrm{i}: \begin{cases} \dot{x}_\mathrm{i} &= A_\mathrm{ii}x_\mathrm{i} + B_\mathrm{i}u_\mathrm{i} + E_\mathrm{i}s_\mathrm{i} \\ y_\mathrm{i} &= C_\mathrm{ii}x_\mathrm{i} + F_\mathrm{i}s_\mathrm{i} \\ z_\mathrm{i} &= C_{z\mathrm{i}}x_\mathrm{i} + D_{z\mathrm{i}}u_\mathrm{i}, \end{cases}$$

where x_i, u_i, and y_i are the state, input, and output of system $\mathcal{S}_\mathrm{i}$, respectively. Additional input variables s_i and output variables z_i are used to describe the interconnection terms with

the neighboring systems. We define the interaction model as $s_i = \sum_{j=1}^{N} L_{ij} z_j$, where L_{ij} are interconnection matrices.

As shown in Figure 7.3, for each physical system $\mathcal{S}_i$, we assume the malicious packet drop attacks occur during the control signals and output measurements transmission between the controller-observer compensator and the physical system. We define $u_1, \ldots, u_N$ as the control signals sent by the compensators; $\tilde{u}_1, \ldots, \tilde{u}_N$ as the corrupted control signals received by the physical systems; $y_1, \ldots, y_N$ as the output measurements sent by the physical systems; and $\tilde{y}_{c1}, \ldots, \tilde{y}_{cN}$ as the corrupted output measurements received by the compensators.

Our first objective is to estimate the state x_i of each system $\mathcal{S}_i$ using the corrupted output measurement y_i. We then use each state estimate $\tilde{x}_i$ in the local controllers instead of the true state.

Since we assumed that each system $\mathcal{S}_i$ is interconnected with its neighboring systems, the state observer $\mathcal{E}_i$ for each system $\mathcal{S}_i$ is to be designed taking into account interconnections. The implementation of the designed observer is shown in Figure 7.4. The inputs of the

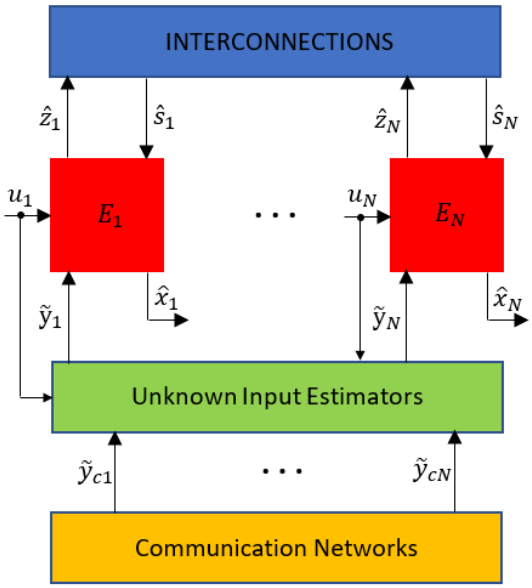


Figure 7.4. Interconnected state observers.

state observer $\mathcal{E}_i$ are $\tilde{y}_i$, u_i, and $\hat{s}_i$, where $\tilde{y}_i$ is calculated using the vector recovery strategy discussed in Section 3.3. The control signal u_i is generated by the controller that we discuss

later. The interaction between the observer $\mathcal{E}_\mathrm{i}$ and its neighboring observers is denoted as $\hat{s}_\mathrm{i}$. We define $\hat{s}_\mathrm{i} = \sum_{\mathrm{j}=1}^{N} L_{\mathrm{ij}}\hat{z}_\mathrm{j}$, where $\hat{z}_\mathrm{j} = C_{z\mathrm{j}}\hat{x}_\mathrm{j} + D_{z\mathrm{j}}u_\mathrm{j}$. The output of the state observer $\mathcal{E}_\mathrm{i}$ is $\hat{x}_\mathrm{i}$, which is the state estimate of the system $\mathcal{S}_\mathrm{i}$. The state estimate $\hat{x}_\mathrm{i}$ is then sent to the decentralized controllers.

We next design the interconnected decentralized controllers. The interconnected controller implementation is shown in Figure 7.5. The input of the interconnected decentralized

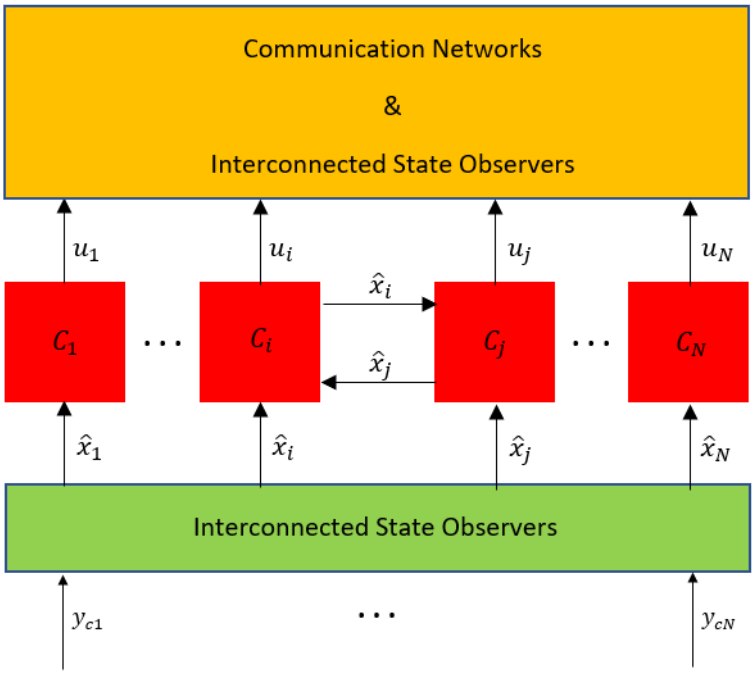


Figure 7.5. Interconnected decentralized controllers.

controller $\mathcal{C}_\mathrm{i}$ is $\hat{x}_\mathrm{i}$, which is state estimate of the system $\mathcal{S}_\mathrm{i}$ generated by the state observer $\mathcal{E}_\mathrm{i}$. The controller $\mathcal{C}_\mathrm{i}$ can exchange the input information with its neighboring controllers. The output of the controller $\mathcal{C}_\mathrm{i}$ is u_i, which is the control signal of system $\mathcal{S}_\mathrm{i}$. The control signal u_i is sent back to the interconnected state observer to create a closed-loop observer-controller compensator. The control signal u_i is sent to the system $\mathcal{S}_\mathrm{i}$ through a communication network.

We show the observer-controller compensator implementation of non-overlapping CPS in Figure 7.6.

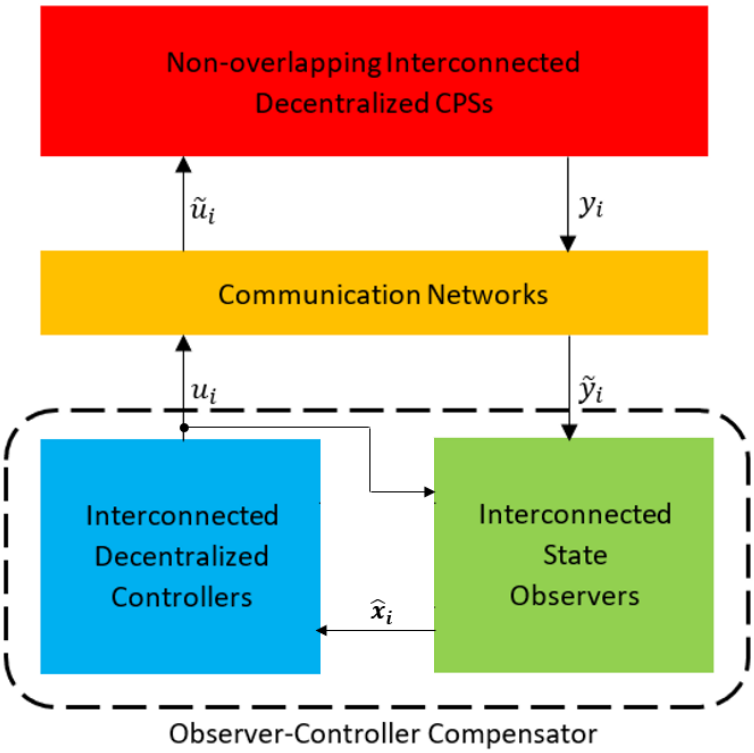


Figure 7.6. Observer-controller compensator implementation for non-overlapping CPS.

REFERENCES

[1] E. A. Lee and S. A. Seshia, *An Introductory Textbook on Cyber-Physical Systems*. New York: WESE, 2010.

[2] F. Molaei, E. Rahimi, H. Siavoshi, S. G. Afrouz, and V. Tenorio, "A comprehensive review on internet of things and its implications in the mining industry," *American Journal of Engineering and Applied Sciences*, vol. 13, no. 3, pp. 499–515, 2020.

[3] H. Xu, W. Yu, D. Griffith, and N. Golmie, "A survey on industrial internet of things: A cyber-physical systems perspective," *IEEE Access*, vol. 6, pp. 78 238–78 259, 2018. DOI: 10.1109/ACCESS.2018.2884906.

[4] X. Yu and Y. Xue, "Smart grids: A cyber–physical systems perspective," *Proceedings of the IEEE*, vol. 104, no. 5, pp. 1058–1070, 2016. DOI: 10.1109/JPROC.2015.2503119.

[5] P. Galambos, "Disruptive robotics and cyber-physical control," in *2018 IEEE 22nd International Conference on Intelligent Engineering Systems (INES)*, 2018, pp. 1–2. DOI: 10.1109/INES.2018.8523849.

[6] Z. Zhao, Y. Li, Y. Yang, L. Li, Y. Xu, and J. Zhou, "Sparse undetectable sensor attacks against cyber-physical systems: A subspace approach," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 67, no. 11, pp. 2517–2521, 2020. DOI: 10.1109/TCSII.2019.2953238.

[7] A. Lu and G. Yang, "Secure luenberger-like observers for cyber–physical systems under sparse actuator and sensor attacks," *Automatica*, vol. 98, pp. 124–129, 2018, ISSN: 0005-1098. DOI: https://doi.org/10.1016/j.automatica.2018.09.003. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S000510981830428X.

[8] Y. Li, L. Shi, P. Cheng, J. Chen, and D. E. Quevedo, "Jamming attacks on remote state estimation in cyber-physical systems: A game-theoretic approach," *IEEE Transactions on Automatic Control*, vol. 60, no. 10, pp. 2831–2836, Oct. 2015.

[9] M. S. Chong, M. Wakaiki, and J. P. Hespanha, "Observability of linear systems under adversarial attacks," in *2015 American Control Conference (ACC)*, Jul. 2015, pp. 2439–2444.

[10] G. Fiore, Y. H. Chang, Q. Hu, M. D. Di Benedetto, and C. J. Tomlin, "Secure state estimation for cyber physical systems with sparse malicious packet drop," in *2017 American Control Conference (ACC)*, Seattle, WA, May 2017, pp. 1898–1903.

[11] Y. H. Chang, Q. Hu, and C. J. Tomlin, "Secure estimation based kalman filter for cyber–physical systems against sensor attacks," *Automatica*, vol. 95, pp. 399–412, 2018.

[12] G. Fiore, A. Iovine, E. De Santis, and M. D. Di Benedetto, "Secure state estimation for dc microgrids control," in *2017 13th IEEE Conference on Automation Science and Engineering (CASE)*, Aug. 2017, pp. 1610–1615.

[13] Y. Luo, K. Chakrabarty, and T. Ho, "Real-time error recovery in cyberphysical digital-microfluidic biochips using a compact dictionary," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 32, no. 12, pp. 1839–1852, Dec. 2013. DOI: 10.1109/TCAD.2013.2277980.

[14] F. Pasqualetti, F. Dörfler, and F. Bullo, "Attack detection and identification in cyber-physical systems," *IEEE Transactions on Automatic Control*, vol. 58, no. 11, pp. 2715–2729, Nov. 2013. DOI: 10.1109/TAC.2013.2266831.

[15] H. Fawzi, P. Tabuada, and S. Diggavi, "Secure estimation and control for cyber-physical systems under adversarial attacks," *IEEE Transactions on Automatic Control*, vol. 59, no. 6, pp. 1454–1467, Jun. 2014.

[16] L. An and G. Yang, "Secure state estimation against sparse sensor attacks with adaptive switching mechanism," *IEEE Transactions on Automatic Control*, vol. 63, no. 8, pp. 2596–2603, Aug. 2018. DOI: 10.1109/TAC.2017.2766759.

[17] S. Nateghi, Y. Shtessel, J. Barbot, G. Zheng, and L. Yu, "Cyber-attack reconstruction via sliding mode differentiation and sparse recovery algorithm: Electrical power networks application," in *2018 15th International Workshop on Variable Structure Systems (VSS)*, Jul. 2018, pp. 285–290.

[18] D. L. Donoho and M. Elad, "For most large underdetermined systems of linear equations the minimal l_1-norm solution is also the sparsest solution," *SIAM Review*, vol. 56, no. 6, pp. 797–829, 2006.

[19] L. Kolev, "Iterative algorithm for the minimum fuel and minimum amplitude problems for linear discrete system," *International Journal of Control*, vol. 21, no. 5, pp. 779–784, 1975.

[20] S. Hui, W. E. Lillo, and S. H. Żak, "Solving minimum norm problems using penalty function and the gradient method," *Automatica*, vol. 31, no. 1, pp. 115–124, 1995.

[21] E. J. Candes and T. Tao, "Decoding by linear programming," *IEEE Transactions on Information Theory*, vol. 51, no. 12, pp. 4203–4215, 2005.

[22] M. Zhang, B. Alenezi, S. Hui, and S. H. Żak, "State estimation of networked control systems corrupted by unknown input and output sparse errors," in *2020 American Control Conference (ACC)*, Denver, CO, Jul. 2020, pp. 4393–4398.

[23] M. Zhang, B. Alenezi, S. Hui, and S. H. Żak, "Unknown input observers for discretized systems with application to networked systems corrupted by sparse malicious packet drops," *IEEE Control Systems Letters*, vol. 5, no. 4, pp. 1261–1266, 2021. DOI: 10.1109/LCSYS.2020.3031454.

[24] P. Tabuada, S. Y. Caliskan, M. Rungger, and R. Majumdar, "Towards robustness for cyber-physical systems," *IEEE Transactions on Automatic Control*, vol. 59, no. 12, pp. 3151–3163, 2014. DOI: 10.1109/TAC.2014.2351632.

[25] C. Deng and C. Wen, "Distributed resilient observer-based fault-tolerant control for heterogeneous multiagent systems under actuator faults and dos attacks," *IEEE Transactions on Control of Network Systems*, vol. 7, no. 3, pp. 1308–1318, 2020. DOI: 10.1109/TCNS.2020.2972601.

[26] M. E. Valcher, "State observers for discrete-time linear systems with unknown inputs," *IEEE Transactions on Automatic Control*, vol. 44, no. 2, pp. 397–401, Feb. 1999, ISSN: 0018-9286. DOI: 10.1109/9.746275.

[27] S. Sundaram and C. N. Hadjicostis, "Delayed observers for linear systems with unknown inputs," *IEEE Transactions on Automatic Control*, vol. 52, no. 2, pp. 334–339, Feb. 2007.

[28] F. Xu, J. Tan, X. Wang, V. Puig, B. Liang, and B. Yuan, "A novel design of unknown input observers using set-theoretic methods for robust fault detection," in *2016 American Control Conference (ACC)*, Jul. 2016, pp. 5957–5961.

[29] G. Basile and G. Marro, "On the observability of linear, time-invariant systems with unknown inputs," *Journal of Optimization Theory and Applications*, vol. 3, no. 6, pp. 410–415, 1969.

[30] R. Patton and J. Chen, *Robust model-based fault diagnosis for dynamic systems*. New York: Springer, 1999.

[31] S. Hui and S. H. Żak, "Observer design for systems with unknown inputs," *International Journal of Applied Mathematics and Computer Science*, vol. 15, no. 4, pp. 431–446, 2005.

[32] J. Han, M. Kamber, and J. Pei, *Data Mining: Concepts and Techniques*, Third. Waltham, Mass: Morgan Kaufmann Publishers, 2012.

[33] G. H. Golub and C. F. Van Loan, *Matrix Computations*. Baltimore, Maryland: The John Hopkins University Press, 1983.

[34] E. Schechter, *Handbook of Analysis and Its Foundations*. London, UK: Academic Press, 1997.

[35] B. K. Natarajan, "Sparse approximate solutions to linear systems," *SIAM Journal on Computing*, vol. 24, no. 2, pp. 4203–4215, 1995.

[36] T. Hastie, R. Tibshirani, and J. H. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. New York: Springer, 2001.

[37] Z. Wei, J. Zhang, Z. Xu, Y. Huang, Y. Liu, and X. Fan, "Gradient projection with approximate l_0 norm minimization for sparse reconstruction in compressed sensing," *Sensors*, vol. 18, no. 10, pp. 3373–3388, 2018.

[38] J. Tropp, "Greed is good: Algorithmic results for sparse approximation," *IEEE Transactions on Information Theory*, vol. 50, no. 10, pp. 2231–2242, 2004.

[39] A. M. Tillmann and M. E. Pfetsch, "The computational complexity of the restricted isometry property, the nullspace property, and related concepts in compressed sensing," *IEEE Transactions on Information Theory*, vol. 60, no. 2, pp. 1248–1259, 2014.

[40] D. L. Donoho and M. Elad, "Optimally sparse representation in general (nonorthogonal) dictionaries via l_1 minimization," *Proceedings of the National Academy of Sciences*, vol. 100, no. 5, pp. 2197–2202, 2003.

[41] Y. Zhang, "Theory of compressive sensing via 1-norm minimization: A non-rip analysis and extensions," *Journal of the Operations Research Society of China*, vol. 1, no. 1, pp. 79–105, 2013.

[42] D. Donoho and J. Tanner, "Counting faces of randomly-projected polytopes when the projection radically lowers dimension," *Journal of the American Mathematical Society*, vol. 22, pp. 1–53, Aug. 2006.

[43] X. Feng and Z. Zhang, "The rank of a random matrix," *Applied Mathematics and Computation*, vol. 185, pp. 689–694, Feb. 2007.

[44] A. Papoulis and S. U. Pillai, *Probability, Random Variables and Stochastic Processes*, Fourth. New York: McGraw-Hill, 2002.

[45] S. H. Żak, *Systems and Control*. New York: Oxford University Press, 2003.

[46] Y. H. Chang, J. W. Gray, and C. J. Tomlin, "Exact reconstruction of gene regulatory networks using compressive sensing," *BMC Bioinformatics*, vol. 15, no. 1, p. 400, Dec. 2014.

[47] E. K. P. Chong and S. H. Żak, *An Introduction to Optimization*, Fourth. Hoboken, New Jersey: John Wiley & Sons, Inc., 2013.

[48] I. Grujic and R. Nilsson, "Model-based development and evaluation of control for complex multi-domain systems: Attitude control for a quadrotor uav," *Technical Report Electronics and Computer Engineering*, vol. 4, no. 23, Mar. 2016.

[49] Y. H. Chang, Q. Hu, and C. J. Tomlin, "Secure estimation based Kalman filter for cyber-physical systems against adversarial attacks," *CoRR*, vol. abs/1512.03853, 2016.

[50] Q. Hu, D. Fooladivanda, Y. H. Chang, and C. J. Tomlin, "Secure state estimation and control for cyber security of the nonlinear power systems," *IEEE Transactions on Control of Network Systems*, vol. 5, no. 3, pp. 1310–1321, Sep. 2018. DOI: 10.1109/TCNS.2017.2704434.

[51] L. Li and X. Song, "State estimation for systems with packet dropping and state equality constraints," *IEEE Transactions on Circuits and Systems II: Express Briefs*, 2018, In print.

[52] Z. Jin, C. Ko, and R. M. Murray, "Estimation for nonlinear dynamical systems over packet-dropping networks," in *2007 American Control Conference*, Jul. 2007, pp. 5037–5042.

[53] M. Hou and P. C. Müller, "Design of observers for linear systems with unknown inputs," *IEEE Transactions on Automatic Control*, pp. 632–635, June 1992.

[54] S. Hui and S. H. Żak, "Stress estimation using unknown input observer," in *Proceedings of American Control Conference (ACC), Washington, DC*, Jun. 2013, pp. 259–264.

[55] K. Kalsi, S. Hui, and S. H. Żak, "Unknown input and sensor fault estimation using sliding-mode observers," *2011 American Control Conference*, pp. 1364–1369, 2011.

[56] A. Termehchy, A. Afshar, and M. Javidsharifi, "A novel design of unknown input observer for fault diagnosis in the Tennessee-Eastman process system to solve non-minimum phase problem," in *2013 IEEE International Conference on Smart Instrumentation, Measurement and Applications (ICSIMA)*, Nov. 2013, pp. 1–6.

[57] J. Anzurez-Marin, N. Pitalua-Diaz, O. Cuevas-Silva, and J. Villar-García, "Unknown inputs observers design for fault detection in a two-tank hydraulic system," in *2008 Electronics, Robotics and Automotive Mechanics Conference (CERMA '08)*, Sep. 2008, pp. 373–378.

[58] C. Edwards, S. K. Spurgeon, and R. J. Patton, "Sliding mode observers for fault detection and isolation," *Automatica*, vol. 36, no. 4, pp. 541–553, 2000.

[59] J. Zarei and S. Ahmadizadeh, "LMI-based unknown input observer design for fault detection," in *The 2nd International Conference on Control, Instrumentation and Automation*, Dec. 2011, pp. 1130–1135.

[60] W. Gritli, H. Gharsallaoui, and M. Benrejeb, "Fault detection based on unknown input observers for switched discrete-time systems," in *2017 International Conference on Advanced Systems and Electric Technologies (IC ASET)*, Jan. 2017, pp. 436–441.

[61] T. Kaczorek, K. M. Przyłuski, and S. H. Żak, *Wybrane Metody Analizy Liniowych Układów Dynamicznych (Selected Methods of Analysis of Linear Dynamical Systems)*. Warszawa (Warsaw): Państwowe Wydawnictwo Naukowe (Polish Scientific Publishers), 1984.

[62] C.-T. Chen, *Linear System Theory and Design*, Fourth. New York: Oxford University Press, 2013.

[63] K. J. Astrom and R. M. Murray, *Feedback Systems: An Introduction for Scientists and Engineers*. Princeton, NJ, USA: Princeton University Press, 2008, ISBN: 0691135762, 9780691135762.

[64] M. L. J. Hautus, "Strong detectability and observers," *Linear Algebra and Its Applications*, vol. 50, pp. 353–368, 1983.

[65] A. Chakrabarty, S. H. Żak, and S. Sundaram, "State and unknown input observers for discrete-time nonlinear systems," in *2016 IEEE 55th Conference on Decision and Control (CDC)*, Dec. 2016, pp. 7111–7116.

[66] M. Zhang, S. Hui, M. R. Bell, and S. H. Żak, "Vector recovery for a linear system corrupted by unknown sparse error vectors with applications to secure state estimation," *IEEE Control Systems Letters*, vol. 3, no. 4, pp. 895–900, Oct. 2019. DOI: 10.1109/LCSYS.2019.2918751.

[67] J. W. Brown and R. V. Churchill, *Complex Variables and Applications*. McGraw-Hill, 2003.

[68] B. Alenezi, M. Zhang, S. Hui, and S. H. Żak, "State observers and unknown input estimators for discrete-time nonlinear systems characterized by incremental multiplier matrices," in *Proceedings of the 59-th Conference on Decision and Control, Jeju Island, Republic of Korea*, Dec. 2020.

[69] P. D. Lax, *Functional Analysis*. New York: Wiley-Interscience, 2002.

[70] F. Boem, A. J. Gallo, D. M. Raimondo, and T. Parisini, "Distributed fault-tolerant control of large-scale systems: An active fault diagnosis approach," *IEEE Transactions on Control of Network Systems*, vol. 7, no. 1, pp. 288–301, 2020.

[71] H. Tzou and C. Chou, "Sensors and actuators," in *Encyclopedia of Vibration*, S. Braun, Ed., Oxford: Elsevier, 2001, pp. 1134–1144, ISBN: 978-0-12-227085-7. DOI: https://doi.org/10.1006/rwvb.2001.0141.

[72] Q. Wang, X. Zhou, G. Yang, and Y. Yang, "Behavior modeling of cyber-physical system based on discrete hybrid automata," in *2013 IEEE 16th International Conference on Computational Science and Engineering*, 2013, pp. 680–684. DOI: 10.1109/CSE.2013.105.

[73] A. Saberi, A. Stoorvogel, and P. Sannuti, *Filtering Theory - With Applications to Fault Detection, Isolation, and Estimation*, ser. Systems & Control: Foundations & Applications. Switzerland: Birkhäuser, 2007, ISBN: 978-0-8176-4301-0.

[74] D. Ichalal and S. Mammar, "On unknown input observers of linear systems: Asymptotic unknown input decoupling approach," *IEEE Transactions on Automatic Control*, vol. 65, no. 3, pp. 1197–1202, 2020. DOI: 10.1109/TAC.2019.2924375.

[75] H. Pishro-Nik, *Introduction to Probability, Statistics, and Random Processes*. Kappa Research LLC, 2014.

[76] A. Chakrabarty, M. J. Corless, G. T. Buzzard, S. H. Żak, and A. E. Rundell, "Sufficient conditions for exogenous input estimation in nonlinear systems," in *Proceedings of the 2016 American Control Conference, Boston, MA*, Jul. 2016.

[77] C. Aboky, G. Sallet, and J.-C. Vivalda, "Observers for Lipschitz non-linear systems," *International Journal of Control*, vol. 75, no. 3, pp. 204–212, 2002.

[78] P. Stankiewicz, "Vehicle control for collision avoidance and rollover prevention using the zero-moment point," Master's thesis, The Pennsylvania State University, May 2015.

[79] F. Giulietti, L. Pollini, and M. Innocenti, "Autonomous formation flight," *IEEE Control Systems Magazine*, vol. 20, no. 6, pp. 34–44, Dec. 2000.

[80] V. Kapilal, A. G. Sparks, J. M. Buffington, and Q. Yan, "Spacecraft formation flying: Dynamics and control," in *Proceedings of the 1999 American Control Conference (Cat. No. 99CH36251)*, vol. 6, Jun. 1999, pp. 4137–4141.

[81] A. Nguyen, Q. Ha, S. Huang, and H. Trinh, "Observer-based decentralized approach to robotic formation control," in *Australasian Conference on Robotics and Automation*, ARAA Australian Robotics & Automation Association, 2004.

[82] D. Swaroop and J. K. Hedrick, "Constant spacing strategies for platooning in automated highway systems," *Journal of Dynamic Systems, Measurement, and Control*, vol. 121, no. 3, pp. 462–470, 1999.

[83] R. R. Negenborn and H. Hellendoorn, "Intelligence in transportation infrastructures via model-based predictive control," in. Springer, Nov. 2010, pp. 3–24.

[84] S. M. Joshi, *Control of Large Fesible Space Structures*, ser. Lecture Notes in Control and Information Sciences. Berlin: Springer, 1989.

[85] A. Alessandri, M. Gaggero, and F. Tonelli, "Min-max and predictive control for the management of distribution in supply chains," *IEEE Transactions on Control Systems Technology*, vol. 19, no. 5, pp. 1075–1089, Sep. 2011.

[86] W. B. Dunbar, "Distributed receding horizon control of dynamically coupled nonlinear systems," *IEEE Transactions on Automatic Control*, vol. 52, no. 7, pp. 1249–1263, Jul. 2007.

[87] D. D. Šiljak, Ed., *Decentralized Control of Complex Systems*, ser. Mathematics in Science and Engineering. Elsevier, 1991.

[88] J. Lavaei, "Decentralized implementation of centralized controllers for interconnected systems," *IEEE Transactions on Automatic Control*, vol. 57, no. 7, pp. 1860–1865, 2012.

[89] J. Farlow, J. E. Hall, J. M. McDill, and B. H. West, *Differential Equation and Linear Algebra*, Second. Upper Saddle River, NJ: Prentice Hall, 2007.

VITA

Mukai Zhang received the B.Eng degree in automation from Nanjing University of Science and Technology, Nanjing, China, in 2013 and the M.Sc degree in advanced control and system engineering from the University of Manchester, Manchester, U.K., in 2015. His research interests include estimation of unsecured cyber-physical systems, unknown input observer application for network control systems, and convex optimization.