Politecnico di Torino

Master's Degree in Mechatronic Engineering
Dipartimento di Automatica e Informatica

Master's Degree Thesis

Observer-based secure state estimation for multi-agents systems subjected to adversial sensor attacks

Relatore:
Prof. Diego Regruto Tomalino

Correlatori:
Prof.ssa Sophie Fosson
Ing. Francesco Ripa

Candidato:
Simone Vincenzo Cilia

Anno Accademico 2024/2025

Abstract

Cyber-Physical Systems (CPSs) represent a critical class of interconnected systems where physical processes are tightly integrated with computation and communication layers. Their growing deployment in safety-critical infrastructures, such as autonomous transportation, industrial automation, and smart energy grids, makes them highly exposed to malicious intrusions. Among the most harmful threats are adversarial sensor attacks, which can stealthily corrupt measurement data and undermine the reliability of state estimation, control performance, and safety.

Traditional observer-based approaches, such as the Luenberger observer, provide accurate state estimation under nominal conditions but lack robustness against adversarial corruption. In this context, sparsity-aware estimation techniques inspired by compressed sensing and convex optimization have recently emerged as promising tools. This thesis explores these methods by revisiting Secure State Estimation (SSE) through the lens of sparse optimization and developing observer-based counterparts capable of jointly reconstructing both the system state and the attack vector. Particular attention is given to the Sparse Soft Observer (SSO) and its Deadbeat variant (D-SSO), which extend proximal-based iterative algorithms for sparse optimization into recursive observer structures for dynamic CPSs.

While practically effective, the stability and convergence of SSO and D-SSO have not been proven in the literature. Building on these insights, the main contribution of this work is the analysis of the stability of observer design in the presence of adversarial sensor attacks. The proposed analysis introduces a novel error dynamics representation and leverages Lyapunov-based tools to establish formal stability guarantees.

Overall, this thesis shows that sparsity-promoting observers, reinforced by the newly developed theoretical foundation, constitute a robust approach. The results pave the way for real-time implementations and distributed extensions in multi-agent CPSs, highlighting the potential impact of the proposed methodology on the resilience of future cyber-physical infrastructures.

Contents

Chapter 1

Introduction

Cyber-Physical Systems and Their Challenges

Cyber-Physical Systems (CPSs) are large-scale interconnected systems where physical processes are deeply integrated with computation, communication, and control. They appear in a wide variety of applications, ranging from autonomous driving and industrial automation to smart grids, medical devices, and robotics [1, 2, 3]. The tight interaction between the cyber and physical layers makes CPSs powerful, but also vulnerable to failures and malicious intrusions.

In infrastructures that are critical to safety, the reliability of the state estimation process is very important because controllers depend on having accurate information about the system state. Any manipulation with sensor data can spread through the control loop, causing instability, unsafe actions, or a drop in performance.

Adversarial Attacks and Secure State Estimation

When discussing the main threats in the context of CPSs, malicious sensor attacks must be mentioned. To integrate robustness to external attacks into the design of an observer, it is necessary to understand the characteristics of attacks that differ from normal measurement noise. Unlike the latter, which is random and often modeled probabilistically, attacks are generated by external agents that aim to mislead the system, and it is not always possible to find a mathematical model that can characterize them. This difference requires the development of new methodologies that go beyond classic estimation.

In the last decade, a wide body of literature has addressed this problem under the term *Secure State Estimation (SSE)*. A common assumption is that attacks are *sparse*, meaning that only a limited number of sensors are compromised at each

time. This assumption is reasonable in large-scale networks, where an attacker may have limited resources, and it has enabled the adoption of sparsity-aware techniques inspired by compressed sensing [4]. Notably, ℓ_1-based formulations such as the Lasso have been successfully applied to reconstruct both the state and the attack vector, see for example [5].

It is important to highlight that the stability of ℓ_1 formulations has not yet been formally demonstrated in the scientific literature. This limitation encourages the need to find alternative methodologies that can guarantee stability without requiring excessively stringent assumptions about the attack structure.

Sparsity in Secure State Estimation

In the literature about secure state assessment, a central thought is that malicious attacks are sparse. This suggests that, at any moment, only a small number of sensors are affected, and most data remains trustworthy. This modeling choice is motivated by the fact that in large-scale CPSs, such as power grids or distributed sensor networks, it is unrealistic for an attacker to compromise all sensors simultaneously, due to resource, access, or synchronization limitations.

Denoting the attack vector $a(k) \in \mathbb{R}^m$ affecting the measurement at time instant k, the sparsity assumption states that

$$\|a(k)\|_0 \ll m,$$

where $\|\cdot\|_0$ indicates the number of non-zero elements of a vector. This means that the number of attacked sensors is small compared to the total number of sensors present in the system. This assumption allows us to study the effect of external attacks using sparsity-promoting optimization algorithms [4], as will be analyzed in Chapter 4.

It is important to remark, however, that sparsity is not an intrinsic property of attacks, but rather a *working hypothesis* that makes the estimation problem tractable.

Observer-Based Approaches

Parallel to optimization-based methods, observer design remains a fundamental tool in control theory for state estimation. Classical structures, such as the Luenberger observer, provide asymptotic or optimal estimation in the absence of adversarial perturbations. More recently, extensions such as the Sparse Soft Ob-

server (SSO) and its deadbeat variant (D-SSO) have been introduced, combining recursive observer design with sparsity-promoting regularization; **see [5] for details**. These approaches can be interpreted as observer counterparts of iterative optimization algorithms like ISTA.

Structure of the Thesis

The remainder of the thesis is organized as follows. Chapter 2 introduces the mathematical preliminaries, including definitions of stability and ISS theory. Chapter 3 presents CPS modeling and discusses security challenges. Chapter 4 reviews secure state estimation methods, focusing on sparsity assumptions and their limitations and introduces alternative formulations such as Elastic Net. Chapter 5 develops the new observer design and provides a detailed stability analysis. Chapter 6 illustrates an application to localization in the presence of adversarial attacks.

Chapter 2

Mathematical Preliminaries

Definition 1 (Discrete-Time Linear Time-Invariant (LTI) System)**.** *A discrete-time system is called* Linear Time-Invariant (LTI) *if it satisfies the following two properties:*

1. ***Linearity:*** *For any inputs $u_1(k)$ and $u_2(k)$ producing outputs $y_1(k)$ and $y_2(k)$, respectively, and for any scalars $\alpha, \beta \in \mathbb{R}$:*

$$\alpha u_1(k) + \beta u_2(k)(k) \;\longrightarrow\; \alpha y_1(k) + \beta y_2(k), \quad \forall k \in \mathbb{Z}.$$

2. ***Time Invariance:*** *If the input $u(k)$ produces the output $y(k)$, then the shifted input $u(k-s)$ produces the shifted output $y(k-s)$, for any integer shift s.*

Proposition 1 (State-Space Representation of Discrete LTI Systems)**.** *A discrete-time LTI system can be represented in the state-space form:*

$$\begin{cases} x(k+1) = A\,x(k) + B\,u(k), \\ y(k) = C\,x(k) + D\,u(k), \end{cases} \tag{2.1}$$

where $x(k) \in \mathbb{R}^n$ is the state, $u(k) \in \mathbb{R}^m$ is the input and $y(k) \in \mathbb{R}^p$ is the measurement vector. $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$, $C \in \mathbb{R}^{p\times n}$, and $D \in \mathbb{R}^{p\times m}$ are constant matrices.

Proof. The linearity and time-invariance properties imply that the system's evolution is fully described by constant matrices acting on the current state and input. The representation above is the most general finite-dimensional form of a discrete-time LTI system. □

Remark on notation: It is important to highlight that throughout the thesis, the notation $\|\cdot\|$ denotes the ℓ_2 norm.

Proposition 2 (Young's Inequality for Inner Products)**.** *Let $a, b \in \mathbb{R}^n$ and $\gamma > 0$. Then:*

$$2\, a^\top b \leq \gamma\, a^\top a + \frac{1}{\gamma}\, b^\top b. \tag{2.2}$$

Proof. From the Cauchy–Schwarz inequality:

$$|a^\top b| \leq \|a\|\, \|b\|, \tag{2.3}$$

and from the arithmetic–geometric mean inequality (with $x = \sqrt{\gamma}\, \|a\|$ and $y = \frac{\|b\|}{\sqrt{\gamma}}$):

$$2\, \|a\|\, \|b\| \leq \gamma \|a\|^2 + \frac{1}{\gamma} \|b\|^2. \tag{2.4}$$

Combining these results and noting that $a^\top a = \|a\|^2$ and $b^\top b = \|b\|^2$ yields the stated inequality. □

2.1 Stability of Discrete-Time Systems via Lyapunov Theory

Stability is an important concept in the study of dynamical systems, since it describes the system's ability to maintain or recover a desired behavior if small perturbations occur. The subsequent definitions have been excerpted from [6].

A stability analysis is essential to predict the behavior of the system when subjected to perturbations around the equilibrium point, determining whether the system converges or remains bounded to it. A straightforward approach to study the stability of a system is the Lyapunov's direct method, which allows one to avoid solving the equations of the system explicitly.

Stability analysis plays a key role in cyber-physical systems (CPS), although the following discussion is general, the results obtained can also be applied to our case study.

Before introducing Lyapunov theory, it is convenient to recall the following fundamental definition.

Definition 2. *A function $f(t, x)$ is said to be* Lipschitz continuous *in $(\bar{t}, \bar{x})$ if there exists a constant $L > 0$ such that*

$$\|f(t, x) - f(t, y)\| \leq L \|x - y\|$$

for all (t, x) *and* (t, y) *in a neighbourhood of* $(\bar{t}, \bar{x})$. *The constant* L *is referred to as the* Lipschitz constant.

Now, consider the case $f(t, x) = f(x)$, i.e., when f does not explicitly depend on time:

- If every point in an open and connected domain $D \subset \mathbb{R}^n$ admits a neighborhood D_0 where the aforementioned inequality holds with some constant $L_0 > 0$, then f is said to be *locally Lipschitz* on the domain D
- f is said to be *Lipschitz* on a set W if the same inequality holds for all points in W with a common Lipschitz constant L. Moreover, any locally Lipschitz function on D is Lipschitz on every compact subset of D.
- f is *globally Lipschitz* if it satisfies the inequality on the whole space $\mathbb{R}^n$.

The Lipschitz constant L does not change with time if $f(t, x)$ explicitly depends on time t. In this case, the same definitions hold *uniformly in* t over a given time interval. Specifically, if $f(t, x)$ is globally Lipschitz on $\mathbb{R}^n$ with a constant L independent of t, then it is said to be globally uniformly Lipschitz.
Lastly, keep in mind that any function that is continuously differentiable on a domain D is likewise Lipschitz continuous on that same domain.

2.1.1 Basic Definitions of Stability

Consider the general nonlinear discrete-time system:

$$x(k+1) = f(x(k)), \qquad x(0) = x_0, \tag{2.5}$$

where $f : D \to \mathbb{R}^n$ is locally Lipschitz in $D \subset \mathbb{R}^n$ and suppose $f(0) = 0$, that is $x = 0$ is an equilibrium point for the system (2.5) (all this can be extended for an equilibrium point different from 0).

Definition 3 (Stability)**.** *The equilibrium point* $x = 0$ *is said to be* stable in the sense of Lyapunov *if, for each* $\varepsilon > 0$, *there exists a* $\delta = \delta(\varepsilon)$ *such that*

$$\|x_0\| < \delta \implies \|x(k)\| < \varepsilon, \quad \forall k \geq 0.$$

Definition 4 (Instability)**.** *The equilibrium point* $x = 0$ *is said to be* unstable *if it is not stable.*

Definition 5 (Asymptotic Stability)**.** *The equilibrium $x = 0$ is* asymptotically stable *if it is Lyapunov stable and δ can be chosen such that*

$$\|x_0\| < \delta \implies \lim_{k\to\infty} x(k) = 0$$

2.1.2 Lyapunov Direct Method for Discrete-Time Systems

Lyapunov's direct method provides a powerful tool to establish stability without explicitly solving the system equations. The idea is to construct a scalar *Lyapunov function* that behaves like an energy measure, decreasing along the system trajectories.

Theorem 1 (Existence of a Lyapunov function implies stability)**.** *Let $x = 0$ be an equilibrium point for the system 2.5 where $f : D \to \mathbb{R}^n$ is locally Lipschitz in $D \subset \mathbb{R}^n$ and $0 \in D$. Suppose there exists a continuous function $V : D \to \mathbb{R}$ such that:*

$$V(0) = 0, \quad V(x) > 0, \quad \forall x \in D \setminus \{0\};$$
$$V(f(x)) - V(x) \leq 0, \quad \forall x \in D.$$

Then $x = 0$ is stable*. Moreover, if*

$$V(f(x)) - V(x) < 0, \quad \forall x \in D \setminus \{0\},$$

then $x = 0$ is asymptotically stable*.*

Theorem 2 (Global asymptotic stability from Lyapunov)**.** *Let consider the system (2.5), where $f : D \to \mathbb{R}^n$ is locally Lipschitz in $D \subset \mathbb{R}^n$ and $0 \in D$. Let $V : \mathbb{R}^n \to \mathbb{R}$ be a continuous function such that:*

1. *$V(0) = 0$ and $V(x) > 0$, $\forall\ x \in D \setminus \{0\}$;*
2. *$\|x\| \to \infty \Rightarrow V(x) \to \infty$;*
3. *The forward difference $\Delta V(x) = V(f(x)) - V(x) < 0$, $\forall x \in D$.*

Then the equilibrium point $x = 0$ is globally asymptotically stable.

If an equilibrium point is globally asymptotically stable, then it is the only possible equilibrium point of the system (2.5).

Remark. The previous theorem can be explained intuitively if we think of the function $V(x)$ as an "energy" function. If the energy decreases ($\Delta V(x) < 0$) along the trajectory, the system dissipates energy and consequently the state converges to equilibrium.

2.1.3 Linear Systems and the Discrete Lyapunov Equation

Consider the linear time-invariant (LTI) system:

$$x(k+1) = Ax(k), \tag{2.6}$$

where $A \in \mathbb{R}^{n\times n}$ is the state transition matrix. The origin is an equilibrium point of the system.

Theorem 3 (Eigenvalue Condition for Stability)**.** *The equilibrium $x = 0$ is stable if and only if all the eigenvalues of A satisfy $|\lambda_i| < 1$ and the algebraic and geometric multiplicity of the eigenvalues with absolute value 1 coincide. The equilibrium point $x = 0$ is globally asymptotically stable if and only if all the eigenvalues of A are such that $|\lambda_i| < 1$.*

A matrix A with all the eigenvalues in absolute value smaller than 1 is called a Schur matrix, and it holds that the origin is asymptotically stable if and only if matrix A is Schur.

In this case, a quadratic Lyapunov function can be chosen as

$$V(x) = x^\top Px, \quad P = P^\top > 0.$$

The corresponding Lyapunov difference is

$$\Delta V(x) = V(f(x)) - V(x) = x^\top A^\top PAx - x^\top Px = x^\top (A^\top PA - P)x := -x^\top Qx$$

Using Theorem 1 we have that if Q is positive-semidefinite the origin is stable, whether if Q is positive definite the origin is asymptotically stable. Fixing a positive definite matrix Q, if the solution of the Lyapunov equation

$$A^\top PA - P = -Q, \tag{2.7}$$

with respect to P is positive definite, then the trajectories converge to the origin. Equation (2.7) is known as the *discrete Lyapunov equation.*

Remark. For every Schur matrix A, the discrete Lyapunov equation admits a unique solution $P > 0$ for each $Q > 0$. This gives the asymptotic stability of linear systems a clear algebraic criterion.

2.1.4 Summary

Lyapunov stability theory provides a rigorous mathematical framework to analyze the convergence and robustness of discrete-time systems and observers. In CPSs, it serves as the foundation for the design of secure estimation algorithms capable of guaranteeing stability even in the presence of malicious attacks.

2.2 Input-to-State Stability

2.2.1 Introduction

Input-to-State Stability (ISS) is a fundamental concept in modern control theory, introduced by Eduardo Sontag in the 1980s [7, 8] and later extended to discrete-time systems by Jiang and Wang [9]. ISS's main goal is to create a theoretical framework that explains how a system's state changes over time, both in relation to its initial conditions and when it is affected by external inputs or disturbances. This method combines the ideas of internal stability and robustness, which are especially helpful for studying nonlinear systems, control networks, and cyber-physical systems that are affected by changes.

An ISS system is intuitively such that the impact of inputs on the dynamics of the state is bounded and measurable: the state converges to zero if the input tends to zero, and the state stays bounded if the input is bounded.

2.2.2 Comparison Functions

For the following stability analyses, it is necessary to introduce the following classes of functions to characterize the comparison functions and their asymptotic behavior: The subsequent definitions have been excerpted from Isidori's seminal work [10].

Definition 6. *A continuous function $\alpha : [0, \alpha) \to [0, \infty)$ is said to belong to a class $\mathcal{K}$ if it is strictrly increasing and $\alpha(0) = 0$. If $\alpha(0) = \infty$ and $\lim_{r\to\infty} \alpha(r) = \infty$, the function is said to belong to class $\mathcal{K}_\infty$.*

Definition 7. *A continuous function $\beta : [0, \alpha) \to [0, \infty)$ is said to belong to class $\mathcal{KL}$ if, for each fixed s, the function*

$$\begin{aligned} \alpha &: [0, \alpha) \to [0, \infty) \\ r &\mapsto \beta(r, s) \end{aligned} \tag{2.8}$$

belongs to a class $\mathcal{K}$ and, for each fixed r, the function

$$\begin{aligned} \phi &: [0,\infty) \to [0,\infty) \\ s &\mapsto \beta(r,s) \end{aligned} \tag{2.9}$$

is decreasing and $\lim_{s\to\infty} \phi(s) = 0$,

2.2.3 Formal Definition

Consider a discrete-time system

$$x(k+1) = f(x(k), u(k)), \quad x(0) = x_0, \tag{2.10}$$

with $x(k) \in \mathbb{R}^n$, $u(k) \in \mathbb{R}^m$, and $f : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^n$ locally Lipschitz and such that $f(0,0) = 0$.

Definition 8 (Input-to-State Stability)**.** *System* (2.10) *is said to be* Input-to-State Stable (ISS) *if there exist a function $\beta \in \mathcal{KL}$ and a function $\gamma \in \mathcal{K}$ such that, for every initial state x_0 and for every bounded input $u(\cdot)$,*

$$\|x(k)\| \leq \beta(\|x_0\|, k) + \gamma\left(\sup_{t\geq 0} \|u(k)\|\right), \quad \forall k \geq 0 \tag{2.11}$$

Equation (2.11) formally expresses that the state $x(k)$ is bounded by two contributions: the decay of the internal dynamics (first term) and the effect of the input (second term).

2.2.4 ISS-Lyapunov Function

Using specific Lyapunov functions, it is possible to verify the ISS, following the following approach:

Definition 9 (ISS-Lyapunov Function)**.** *A continuous function $V : \mathbb{R}^n \to \mathbb{R}_{\geq 0}$ is an* ISS-Lyapunov function *for system* (2.10) *if there exist $\alpha_1, \alpha_2, \alpha_3 \in \mathcal{K}_\infty$ and $\sigma \in \mathcal{K}$ such that*

$$\alpha_1(\|x_0\|) \leq V(x_0) \leq \alpha_2(\|x_0\|), \tag{2.12}$$

$$V(f(x_0,u)) - V(x_0) \leq -\alpha_3(\|x_0\|) + \sigma(\|u\|), \tag{2.13}$$

for all $x_0 \in \mathbb{R}^n$ and for all $u \in \mathbb{R}^m$.

Theorem 4 (ISS-Lyapunov Equivalence, Jiang & Wang, 2001)**.** *System* (2.10) *is ISS if and only if there exists an ISS-Lyapunov function.*

The proof relies on systematic constructions of Lyapunov functions based on the system's response to bounded inputs.

2.2.5 Robustness and Applications

An essential tool for assessing system robustness is ISS, which measures a system's capacity to reduce noise and disturbances. This is essential in real-world scenarios where sensors and actuators are subject to uncertainties, measurement noise, and malicious attacks.

Chapter 3

Cyber-Physical Systems: Modeling and Security

The integration of computation, communication, and control has led to the emergence of *Cyber-Physical Systems* (CPSs), which represent the structure of modern technological infrastructures. These systems are characterized by the tight coupling between the physical processes governed by the laws of physics and the cyber components that process information and make control decisions.

To fully understand the functioning of CPSs, a multidisciplinary approach is required, as it requires expertise in systems theory, control theory, and communication networks. Two aspects play an important role in CPSs. One is related to system modeling, which describes the system dynamics from both a physical and computer perspective. The second concerns security analysis, which ensures adequate system behavior even under adverse conditions.

This chapter provides a comprehensive overview of the architecture and mathematical modeling of CPSs. Finally, the security challenges arising from cyber attacks, which can compromise the proper functioning of the system, are also presented. These concepts serve as an introduction to Chapter 4, which explores secure state estimation techniques in greater depth.

3.1 Definition and Architecture of CPSs

Cyber-Physical Systems (CPSs) are systems in which computation, networking, and physical processes are deeply intertwined. The National Institute of Standards and Technology (NIST) defines CPSs as "smart systems that include engineered interacting networks of physical and computational components" [11]. These systems operate by collecting data from the physical world via sensors, processing the data

through computational units, and acting on the environment through actuators. This forms a tight feedback loop between the cyber and physical layers. CPSs, are Sensors, actuators, computational units and communication infrastructure are the main parts that make up a CPS, each performing a different function:

- **Sensors:** Devices for gathering information from the physical world;
- **Actuators:** Equipment that responds to control inputs by performing physical actions;
- **Computational units:** Data processing and decision-making modules;
- **Communication infrastructure:** Data transmission channels, frequently wireless, between components.

3.2 Examples of CPSs

Cyber-Physical Systems are widely deployed across several domains of engineering and technology. One prominent example is that of autonomous vehicles, which integrate numerous sensors such as LiDAR, GPS, and cameras, along with vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) communication systems, to enable safe and intelligent navigation in dynamic environments [12].

Another important application is found in industrial automation, where CPSs consist of networks of programmable logic controllers and robots that perform synchronized tasks in real time, often under strict safety and timing constraints [13, 14].

CPSs are also exemplify by smart grids. In order to ensure efficiency and resilience, these systems integrate digital sensors and control units with physical infrastructure for electricity distribution. This allows for real-time monitoring, demand-response tactics, and fault detection [15]. CPSs are found in medical equipment like surgical robots, insulin pumps, and pacemakers. These systems, continuously monitor physiological signals and modify their operation accordingly [16, 17, 18].

Finally, wireless sensor networks (WSNs) represent a class of CPSs where spatially distributed nodes, each equipped with sensors and limited computing power, collaborate to monitor environmental, structural, or industrial parameters. These systems are essential for applications such as indoor localization, forest fire detection, and infrastructure health monitoring [19, 20].

3.3 Mathematical Modeling of CPSs

CPSs can be modeled using several frameworks. As a starting point, we can model CPSs as discrete-time (DT) LTI systems:

$$x(k+1) = Ax(k) + Bu(k), \quad y(k) = Cx(k) + Du(k) \tag{3.1}$$

for $k = 0, 1, 2, ... \in \mathbb{R}$; $x(k) \in \mathbb{R}^n$; $u(k) \in \mathbb{R}^p$; $y(k) \in \mathbb{R}^q$.
Where $x(k)$ is the system state, $u(k)$ the input, and $y(k)$ the output. For instance $x(k)$ can represent the position of a moving target at time instant k and $y(k)$ is the vector of the measurements taken by q different sensor nodes.
In this scenario, A, B, C, and D are constant matrices describing the dynamics and measurement relations. For instance, in a localization problem, $x(k)$ may represent the position and velocity of a target, while $y(k)$ corresponds to the measurements collected by multiple sensor nodes.
Subsequently, we will address the problem from a broader perspective: that of multi-agent systems. We refer to a network of agents modeled as dynamical systems, each with a local state and inputs, communicating over a graph to achieve a global objective such as consensus.

3.4 Security Challenges in CPSs

The integration of networking and computing makes CPSs vulnerable to attacks. It is important to highlight that external attacks cannot be treated as simple noise or disturbances. Therefore, it is necessary to develop appropriate strategies to counter them. The main characteristics of an external attack include sparsity, meaning that only a limited number of sensors are affected, since the sensor nodes are physically distributed; consequently, only a small subset of them can be targeted. Furthermore, a "good attack" is stealthy, meaning it is designed to mimic noise. Moreover, detection is made more difficult by the fact that a well-designed attack is not easily recognizable with a precise mathematical model. Such attacks can be introduced on both the system's sensors and actuators. The original discrete-time system can be modified by adding additive terms to the state and measurement equations:

$$x(k+1) = Ax(k) + Bu(k) + b(k), \quad y(k) = Cx(k) + Du(k) + a(k) \tag{3.2}$$

for $k = 0, 1, 2, ... \in \mathbb{R}$; $x(k) \in \mathbb{R}^n$; $u(k) \in \mathbb{R}^p$; $y(k) \in \mathbb{R}^q$; $b(k) \in \mathbb{R}^n$; $a(k) \in \mathbb{R}^q$.
Where $a(k)$ represents attacks on sensors and $b(k)$ the attacks on actuators.

These attacks can significantly compromise state estimation and control performance if not properly addressed.

Chapter 4

Secure state estimation for CPSs

As discussed in the previous chapter, cyber-physical systems (CPS) are inherently vulnerable to external attacks that can compromise the correct functioning of sensors and actuators. Traditional estimation and control techniques, such as the Luenberger observer, fail in the presence of malicious attacks.

The goal of this chapter is to formalize the problem of *secure state estimation* under sparse sensor attacks.Our goal is to create estimation algorithms that can reliably recover the system state even in the case that an adversary arbitrarily perturbs with a portion of the measurements.

We first define the attacked system model and characterize the fundamental limitations imposed by observability attacks. Next, we present formulations based on optimization that draw inspiration from sparse recovery theory. Lastly, we introduce observer dynamics that use the sparsity assumption to estimate malicious attacks.

4.1 State estimation of an LTI system

Before addressing the problem of state estimation in presence of attacks, it is important to understand the basics concepts of state estimation and observability in a simpler and more general context. The goal is to clearly define the concepts of observability and observers (in particular the Luenberger observer), providing a rigorous starting point to extend the theory to secure state estimation in more realistic scenarios subject to disturbances or attacks.

Let us consider the discrete-time LTI system:

$$x(k+1) = Ax(k), \quad y(k) = Cx(k) \tag{4.1}$$

where $x(k) \in \mathbb{R}^n$; $y(k) \in \mathbb{R}^q$; $A \in \mathbb{R}^{n,n}$; $C \in \mathbb{R}^{q,n}$.

When we talk about estimation, we refer to the process of inferring the system state from the available measurements. If the system under consideration is known (i.e., the matrices A and C are known), then estimating the initial state is sufficient to determine all future states of the system. A system is observable if there exists a finite time T such that $x(0)$ can be recovered from measurements $y(k)$. We can also determine whether a system is observable from a mathematical standpoint:

$$x(k+1) = Ax(k), \quad y(k) = Cx(k)$$

$$\Rightarrow \begin{cases} y(0) = Cx(0) \\ y(1) = Cx(1) = CAx(0) \\ y(2) = Cx(2) = CAx(1) = CA^2x(0) \\ \vdots \\ y(T-1) = \cdots = CA^{T-1}x(0) \end{cases}$$

In matrix form:

$$\begin{pmatrix} y(0) \\ y(1) \\ \vdots \\ y(T-1) \end{pmatrix} = \underbrace{\begin{pmatrix} C \\ CA \\ \vdots \\ CA^{T-1} \end{pmatrix}}_{\mathcal{O}_\mathcal{T} \in \mathbb{R}^{qT,n}} x(0) \tag{4.2}$$

Whenever the equation (4.2) has a unique solution, the system is observable.
If $qT \geq n$, and $rank(\mathcal{O}_T) = n$, we can solve the equation by (pseudo)inverting it.
If $T = n$, we define the observability matrix O_n as:

$$O_n = \begin{pmatrix} C \\ CA \\ \vdots \\ CA^{n-1} \end{pmatrix}$$

The following fundamental result, originally proved by Kálmán in the 1960s, characterizes the observability of the system.

Theorem [Kálmán, 1960s,[21]]
An LTI system is observable if and only if the observability matrix O_n has full rank, i.e.,

$$\text{rank}(O_n) = n.$$

This condition ensures that the initial state of the system can be uniquely determined

from the output measurements over a finite time horizon.

4.2 Luenberger state observer

For a discrete-time LTI system:

$$\begin{cases} x(k+1) = Ax(k) \\ y(k) = Cx(k) \end{cases}$$

a standard structure for state estimation is the Luenberger observer, defined as:

$$\begin{cases} \hat{x}(k+1) = A\hat{x}(k) + L\left[y(k) - \hat{y}(k)\right] \\ \hat{y}(k) = C\hat{x}(k) \end{cases}$$

where $\hat{x}(k)$ is the estimated state, and $L \in \mathbb{R}^{n \times q}$ is the observer gain matrix to be designed.

Error Dynamics

Let $e(k) = \hat{x}(k) - x(k)$ denote the estimation error.

$$\begin{aligned} e(k+1) &= \hat{x}(k+1) - x(k+1) \\ &= A\hat{x}(k) + L\big(y(k) - \hat{y}(k)\big) - Ax(k) \\ &= A(\hat{x}(k) - x(k)) - LC(\hat{x}(k) - x(k)) \\ &= (A - LC)\, e(k). \end{aligned}$$

From the error dynamics expressed above, it is straightforward to verify that the observer is asymptotically stable if and only if $A - LC$ is Schur.

A fundamental result guarantees that an appropriate gain L exists under the observability assumption:

Theorem 5. *If the pair (A, C) is observable, then there exists a gain matrix L such that the matrix $A - LC$ is asymptotically stable.*

This ensures that the estimation error $e(k)$ converges to zero as $k \to \infty$, i.e., the observer asymptotically tracks the true state.

Dead-beat observer

The speed of convergence of the Luenberger observer depends on the eigenvalues of $A - LC$. In some cases, it is desirable to choose these eigenvalues to be all zero; this leads to the so-called *deadbeat observer*, which converges in exactly n steps.

Pole placement techniques can be used for observer design.

The Luenberger observer offers a baseline structure for secure state estimation, but it is susceptible to sensor attacks. We need to exploit a new strategy to address this issue.

4.3 Secure state estimation

Let us consider the following CPS model with attacks on sensors:

$$x(k+1) = Ax(k), \quad y(k) = Cx(k) + a(k)$$

We assume that each sensor i takes a single (scalar) measurement: $y_i \in \mathbb{R}$. The goal of **Secure State Estimation (SSE)** is to estimate the system state $x(k)$ from output measurements $y(k)$, even in the presence of external attacks on the sensors. From the perspective of attack identification, we do not focus on estimating the magnitude of the attack, but rather on identifying which sensors are under attack. Secure State Estimation is challenging due to the lack of any prior information about $a(k)$, such as its dynamics or probabilistic distribution.

4.3.1 SSE for static systems

The Optimal Decoder D_0

To address the problem of secure state estimation, we need to account for the presence of attacks, which represents the key difference with respect to the previous (non-adversarial) case. To simplify the analysis, we first consider a static system with $A = I$, so that we can isolate and study the problem associated with the attack component. The attacks can be reasonably assumed to be sparse, and this property can be exploited to recover their effect and correctly estimate the state. Following the approach introduced in [22], we define the optimal decoder D_0 as the solution to the following optimization problem:

$$D_0(y) := \arg \min_{x \in \mathbb{R}^n,\, a \in \mathbb{R}^q} \|a\|_0 \quad \text{subject to } y = Cx + a, \tag{4.3}$$

where $\|a\|_0$ denotes the ℓ_0 norm, i.e., the number of nonzero entries in the attack vector a.
This decoder aims to reconstruct the state x by identifying the sparsest possible attack vector a that explains the discrepancy between the measurement y and the nominal output Cx. Although D_0 achieves optimal performance in terms of resilience against sparse attacks, the associated optimization problem is *combinatorial* in nature, meaning that it requires searching over all possible combinations of nonzero entries in a. Specifically, minimizing the ℓ_0 norm is an *NP-hard* problem, i.e., a problem for which no known algorithm can find the exact global solution in polynomial time. As a consequence, this approach becomes computationally intractable for large-scale systems.

Relaxed Decoder D_1 via ℓ_1 Minimization

To address the issue related to the intractability of the optimal decoder D_0, a more computationally efficient formulation is required. A common approach is to replace the non-convex ℓ_0 norm with its best convex approximation: the ℓ_1 norm. This leads to a relaxed formulation of the decoding problem.

We define the decoder $D_1 : \mathbb{R}^q \to \mathbb{R}^n$ as the solution of:

$$\min_{x \in \mathbb{R}^n,\, a \in \mathbb{R}^q} \|a\|_1 \quad \text{subject to } y = Cx + a, \tag{4.4}$$

where $\|a\|_1$ denotes the sum of the absolute values of the entries of the vector a.

This optimization problem is convex, and therefore computationally feasible using standard convex optimization techniques. The decoder D_1 can be viewed as a relaxation of the original problem defined by D_0, and as such, it is generally suboptimal compared to D_0 in terms of attack correction capabilities.

However, D_1 is a useful tool in practice for secure state estimation in the presence of sparse adversarial attacks because it balances tractability and performance.

Compressed Sensing Correlation

The relaxation introduced with the decoder D_1 naturally leads to an interesting connection with the theory of *compressed sensing* (CS). By reformulating the optimization problem in a more compact form, it becomes apparent that the estimation of the system state in the presence of sparse attacks can be viewed through the same mathematical lens as the recovery of sparse signals from limited measurements. Compressed sensing provides a well-established theoretical and algorithmic framework for reconstructing sparse vectors from an underdetermined set of linear

equations [23, 24]. This framework relies on the observation that, under suitable conditions on the sensing matrix, a sparse signal can be exactly recovered by solving an ℓ_1-minimization problem instead of the original ℓ_0-minimization one, which is computationally intractable.

This analogy is especially relevant to secure state estimation: the measurement matrix C acts as the sensing operator, and the attack vector a represents the sparse signal in CS. The task of recovering both the attack vector and the system state is thus closely related to sparse recovery in compressed sensing, and the measurement corruption introduced by sparse attacks can be interpreted as a sparse perturbation of the true output.

This correspondence not only provides valuable theoretical insight into the properties of the relaxed decoder D_1 but also allows the application of well-known CS results, such as recovery guarantees and algorithmic approaches, to the problem of secure state estimation. In this section we formalize this connection and introduce the *partial* compressed sensing formulations that arise in this context.

The optimization problem in (4.4) can be equivalently rewritten in a compact form by defining the augmented matrix $G = (C \ I) \in \mathbb{R}^{q\times(n+q)}$ and the stacked variable $z = \begin{pmatrix} x \\ a \end{pmatrix} \in \mathbb{R}^{n+q}$. The constraint becomes:

$$y = Gz. \tag{4.5}$$

Thus, the optimization problem becomes:

$$\min_{z\in\mathbb{R}^{n+q}} \|a\|_1 \quad \text{subject to } y = Gz, \tag{4.6}$$

where the objective still involves only the a component of z. This compact formulation reveals a strong connection with the theory of *compressed sensing* (CS).

According to CS theory, a k-sparse vector $x \in \mathbb{R}^n$, with $k \ll n$, can be accurately recovered from a set of compressed linear measurements:

$$y = \Theta x + (\text{noise}), \quad y \in \mathbb{R}^q, \quad q < n, \tag{4.7}$$

provided that the sensing matrix $\Theta \in \mathbb{R}^{q\times n}$ satisfies specific conditions.

The measurements y are corrupted by a sparse attack a, and the goal is to recover the original state x despite this sparse corruption.

The original sparse recovery problem is expressed as:

$$\min_{x \in \mathbb{R}^n} \|x\|_0 \quad \text{subject to } \Theta x = y. \tag{4.8}$$

which is combinatorial in nature and NP-hard.

To make the problem tractable, a common approach is to relax the ℓ_0-norm to its best convex approximation, the ℓ_1-norm. This leads to the well-known *Basis Pursuit* formulation:

$$\min_{x \in \mathbb{R}^n} \|x\|_1 \quad \text{subject to } \Theta x = y. \tag{4.9}$$

An alternative and widely used variant is the *Lasso* (Least Absolute Shrinkage and Selection Operator) formulation introduced by Tibshirani in 1996 [25]. It introduces a regularization term that penalizes the ℓ_1-norm while allowing for **noisy measurements**:

$$\min_{x \in \mathbb{R}^n} \frac{1}{2}\|\Theta x - y\|_2^2 + \lambda \|x\|_1, \quad \lambda > 0. \tag{4.10}$$

Here, λ is a hyperparameter that controls the trade-off between sparsity and data fidelity.

In the context of secure state estimation, the idea of sparsity is applied not to the state vector x, but to the attack vector $a \in \mathbb{R}^q$. This leads to what can be considered a "partial" version of the standard CS formulations.

The *Partial Lasso* formulation introduces a trade-off between fitting the measurements and promoting sparsity in the attack vector:

$$\min_{x \in \mathbb{R}^n,\, a \in \mathbb{R}^q} \frac{1}{2} \left\| y - G \begin{pmatrix} x \\ a \end{pmatrix} \right\|_2^2 + \lambda \|a\|_1, \quad \lambda > 0. \tag{4.11}$$

where $G = (C\ I)$ is the augmented measurement matrix.

These formulations allow for secure state estimation in scenarios with measurement corruption.

Partial Lasso for SSE

In the context of *Secure State Estimation* (SSE), we now address a formulation that explicitly accounts for both adversarial attacks and measurement noise. Consider the static model

$$y = Cx + a + \eta,$$

where $x_e \in \mathbb{R}^n$ denotes the unknown system state, $a_e \in \mathbb{R}^q$ represents a sparse attack vector, and $\eta \in \mathbb{R}^q$ models measurement noise. The goal is to estimate the true state

x_e despite the presence of these perturbations.

A widely used and effective approach to this problem is the *Partial Lasso* (P-Lasso) formulation, which casts the estimation as a convex optimization problem:

$$\min_{x\in\mathbb{R}^n,\, a\in\mathbb{R}^q} F(x,a) + G(a),$$

where the objective is composed of a data-fitting term

$$F(x,a) := \frac{1}{2}\|Cx + a - y\|_2^2,$$

and a regularization term promoting sparsity in the attack vector:

$$G(a) := \lambda\|a\|_1,$$

with $\lambda > 0$ being a design hyperparameter that balances data fidelity and sparsity. The presence of the ℓ_1-norm in the cost function encourages sparse solutions for a, making this formulation particularly suited to scenarios where only a small subset of sensors may be compromised.

The P-Lasso problem is convex, but standard gradient descent methods cannot be applied because the ℓ_1-norm is non-differentiable. Iterative first-order optimization algorithms that take advantage of the *proximal mapping* [26] concept can be used to solve this.

Considering the convex function $G : \mathbb{R}^q \to \mathbb{R}$ defined before, its proximal mapping is defined as:

$$\text{prox}_G(z) = \arg\min_{a\in\mathbb{R}^q} \left\{ G(a) + \frac{1}{2}\|a - z\|_2^2 \right\}.$$

In our case, with $G(a) = \lambda\|a\|_1$, the proximal mapping becomes:

$$\text{prox}_{\lambda\|\cdot\|_1}(z) = \arg\min_{a\in\mathbb{R}^q} \left\{ \lambda\|a\|_1 + \frac{1}{2}\|a - z\|_2^2 \right\}.$$

This operator is known as the *soft thresholding* or *shrinkage* operator and is separable across the components of the vector. In other words, it can be applied component-wise:

$$\mathcal{S}_{\lambda_i}(z_i) = \begin{cases} z_i - \text{sign}(z_i)\lambda_i & \text{if } |z_i| > \lambda_i, \\ 0 & \text{otherwise,} \end{cases} \qquad \text{for } i = 1, \ldots, q. \tag{4.12}$$

This operation reduces the magnitude of each component by λ, setting it to zero if its absolute value is smaller than λ.

This efficient computation of the proximal operator allows the use of proximal gradient methods for solving the P-Lasso problem, ensuring a computationally feasible approach to secure state estimation even in the presence of sparse adversarial corruption and measurement noise.

Elastic Net for Secure State Estimation

An extension of the partial Lasso involves the elastic net model, which also uses the ℓ_2-norm optimization variable. It is defined as:

$$(\hat{x}, \hat{a}) = \arg\min_{x,a} \frac{1}{2}\|y - Cx - a\|_2^2 + \lambda\|a\|_1 + \frac{\mu}{2}\|a\|_2^2, \tag{4.13}$$

where λ,$\mu \geq 0$ are regularization parameters.

This formulation retains a sparsity-promoting component by minimizing the ℓ_1 norm derived from Lasso.

The Elastic Net can be interpreted as a natural extension of Lasso and Ridge regression, both of which belong to the family of proximal optimization methods [4]. Throughout the thesis, this regularization algorithm will be applied in the theoretical discussion presented in Chapter 5.

4.3.2 Iterative Algorithms for Solving the P-Lasso Problem

A limitation of the P-lasso problem is its non-differentiability. Specific optimization algorithms are used to solve this optimization problem. This section presents two such algorithms, both belonging to the class of first-order methods: the Iterative Soft Thresholding Algorithm (ISTA) and the Inertial Jacobi Alternating Minimization (IJAM). These algorithms are presented in [4].

Inertial Jacobi Alternating Minimization (IJAM)

The first algorithm proposed is IJAM, which, as described previously, finds a solution to the P-Lasso problem. It uses a parameter $\nu \in (0, 1]$, which must be tuned correctly to stabilize the convergence of the algorithm.

The algorithm proceeds as follows:

- **Initialization:** Set $x(0) = 0 \in \mathbb{R}^n$, $a(0) = 0 \in \mathbb{R}^q$, and choose an inertia parameter $\nu \in (0, 1]$.

- **Iterative step:** For $k = 0, 1, \ldots, T_{\max}$:

$$x(k+1) = \arg\min_{x \in \mathbb{R}^n} F(x, a(k)) = C^{\dagger}(y - a(k)),$$
$$a(k+1) = \mathcal{S}_{\nu\lambda}\left[a(k) - \nu(Cx(k) + a(k) - y)\right],$$

  where $C^{\dagger}$ denotes the Moore-Penrose pseudoinverse of C.

Iterative Soft Thresholding Algorithm (ISTA)

An alternative approach to solving the P-Lasso problem is offered by the Iterative Soft Thresholding Algorithm (ISTA). This method leverages the fact that the data-fitting term $F(x, a)$ is differentiable with respect to x, enabling a gradient descent update, while the term $G(a) = \lambda\|a\|_1$ is treated via its proximal mapping.

The ISTA algorithm follows these steps:

- **Initialization:** Set $x(0) = 0$, $a(0) = 0$, and choose a step size $\nu > 0$.

- **Iterative step:** For $k = 0, 1, \ldots, T_{\max}$:

$$x(k+1) = x(k) - \nu\nabla_x F(x(k), a(k)) = x(k) - \nu C^{\top}(Cx(k) + a(k) - y),$$
$$a(k+1) = \mathcal{S}_{\nu\lambda}\left[a(k) - \nu(Cx(k) + a(k) - y)\right].$$

The ISTA algorithm can be interpreted as a gradient-proximal method, where each iteration alternates between a gradient descent step on the smooth part of the objective and a proximal step to handle the non-smooth ℓ_1-regularization. The use of the soft-thresholding operator in both IJAM and ISTA highlights the key role of sparsity-promoting regularization in robust state estimation.

Summary. Both IJAM and ISTA are provably convergent under standard assumptions, with IJAM offering greater stability through its inertial parameter and ISTA providing a natural proximal gradient interpretation. These methods enable effective and scalable solutions to secure state estimation problems in the presence of sparse adversarial disturbances.

4.3.3 SSE for dynamic systems

Having previously studied and discussed the problem of secure state estimation in the context of static systems, we now extend our analysis to dynamic systems. This allows us to incorporate the effect of state evolution over time, addressing

dynamic estimation problems where the system matrix A differs from the identity matrix.

We consider a discrete-time linear model of a Cyber-Physical System (CPS), where the measurements may be affected by sparse malicious attacks. The dynamics of the system are described by the following equations:

$$\begin{cases} x(k+1) = Ax(k), \\ y(k) = Cx(k) + a(k), \end{cases}$$

where $x(k) \in \mathbb{R}^n$ and $a(k) \in \mathbb{R}^q$ represent the system state and the attack, respectively at time step k, $y(k) \in \mathbb{R}^q$ is the measurement vector, $A \in \mathbb{R}^{n \times n}$ is the state transition matrix, and $C \in \mathbb{R}^{q \times n}$ is the output matrix. As described before, the scenario considered promotes the sparsity of the attack.

We now extend the model of the CPS with sparse sensor attacks over a finite observation horizon of T time steps. Assuming no process noise, the evolution of the system state is given recursively by the state equation $x(k+1) = Ax(k)$, and the output is affected by a sparse attack vector $a(k)$, resulting in the measured output:

$$y(k) = Cx(k) + a(k), \quad \text{for } k = 0, 1, \ldots, T-1.$$

By recursively applying the state equation, we can express each measurement $y(k)$ in terms of the initial state $x(0)$ as follows:

$$\begin{cases} y(0) = Cx(0) + a(0), \\ y(1) = CAx(0) + a(1), \\ y(2) = CA^2x(0) + a(2), \\ \vdots \\ y(T-1) = CA^{T-1}x(0) + a(T-1). \end{cases} \tag{4.14}$$

By stacking all the output equations into a single expression, we obtain the following compact matrix form:

$$\begin{pmatrix} y(0) \\ y(1) \\ \vdots \\ y(T-1) \end{pmatrix} = \underbrace{\begin{pmatrix} C \\ CA \\ \vdots \\ CA^{T-1} \end{pmatrix}}_{O_T} x(0) + \begin{pmatrix} a(0) \\ a(1) \\ \vdots \\ a(T-1) \end{pmatrix},$$

where $O_T \in \mathbb{R}^{qT \times n}$. This batch formulation reveals that the overall measurement vector over time is a linear function of the initial state $x(0)$, plus the stacked sparse attack vector.

We thus obtain a system of qT linear equations with $n + qT$ unknowns, i.e., the n-dimensional initial state and the qT-dimensional attack vector. This structure underpins sparse state estimation techniques such as Partial Lasso, where the recovery of $x(0)$ is made possible despite the presence of sparse adversarial corruptions in the output.

Dynamic CPS Model under Constant Sparse Sensor Attack

Let us now consider a particular case of the cyber-physical system (CPS) model in which the sensor attack vector remains constant over time. Specifically, we assume that for each time step $k = 0, 1, \ldots$, the attack vector satisfies $a(k) = a$ for some unknown but fixed vector $a \in \mathbb{R}^q$. The system dynamics are still governed by the linear discrete-time state-space equations:

$$\begin{cases} x(k+1) = Ax(k), \\ y(k) = Cx(k) + a, \end{cases} \tag{4.15}$$

where $x(k) \in \mathbb{R}^n$ denotes the state of the system, $y(k) \in \mathbb{R}^q$ the measured output, $A \in \mathbb{R}^{n \times n}$ the state transition matrix, and $C \in \mathbb{R}^{q \times n}$ the output matrix. The vector a models a sparse adversarial signal injected into the sensor measurements.

By unrolling the system outputs over time, we obtain:

$$\begin{cases} y(0) = Cx(0) + a, \\ y(1) = CAx(0) + a, \\ y(2) = CA^2x(0) + a, \\ \quad \vdots \\ y(T-1) = CA^{T-1}x(0) + a. \end{cases} \tag{4.16}$$

This set of equations can be compactly expressed in matrix form as:

$$\begin{pmatrix} y(0) \\ y(1) \\ \vdots \\ y(T-1) \end{pmatrix} = \mathcal{O}_T x(0) + \begin{pmatrix} a \\ a \\ \vdots \\ a \end{pmatrix}, \tag{4.17}$$

where $\mathcal{O}_T$ denotes the standard observability matrix of order T, defined as:

$$\mathcal{O}_T = \begin{pmatrix} C \\ CA \\ \vdots \\ CA^{T-1} \end{pmatrix} \in \mathbb{R}^{qT \times n}. \tag{4.18}$$

We may then rewrite the system in the form:

$$\begin{pmatrix} y(0) \\ y(1) \\ \vdots \\ y(T-1) \end{pmatrix} = \begin{pmatrix} C & I \\ CA & I \\ \vdots & \vdots \\ CA^{T-1} & I \end{pmatrix} \begin{pmatrix} x(0) \\ a \end{pmatrix}, \tag{4.19}$$

where $I \in \mathbb{R}^{q \times q}$ is the identity matrix. We define the *augmented observability matrix* $\mathcal{O}'_T$ of the extended system as:

$$\mathcal{O}'_T = \begin{pmatrix} C & I \\ CA & I \\ \vdots & \vdots \\ CA^{T-1} & I \end{pmatrix} \in \mathbb{R}^{qT \times (n+q)}. \tag{4.20}$$

The augmented system can thus be interpreted as a linear system with state $\begin{pmatrix} x(0)^\top & a^\top \end{pmatrix}^\top \in \mathbb{R}^{n+q}$ and observability matrix $\mathcal{O}'_T$.

Now, we aim to determine whether this system is observable and under which conditions observability holds.

Observability of Dynamic CPSs under Constant Sensor Attacks

We now consider the observability of a dynamic cyber-physical system (CPS) subject to constant sensor attacks. Recall the system model (4.15) The analysis of observability in such an augmented setting is closely related to the study of composite systems, as discussed by **Davison and Wang** in [27].

Proposition 3 (Observability under Constant Attacks)**.** *Let us assume that the attack-free system (A, C) is observable. If the matrix A has an eigenvalue equal to* 1*, then the dynamic CPS under constant attacks is* not observable.

Proof. If A has an eigenvalue equal to 1, then the matrix $I - A \in \mathbb{R}^{n \times n}$ has a

corresponding eigenvalue equal to 0, which implies that $I - A$ is not full rank.

We analyze the observability of the augmented system:

$$\begin{pmatrix} y(0) \\ y(1) \\ \vdots \\ y(T-1) \end{pmatrix} = \mathcal{O}'_T \begin{pmatrix} x \\ a \end{pmatrix}, \quad \text{where} \quad \mathcal{O}'_T = \begin{pmatrix} C & I \\ CA & I \\ \vdots & \vdots \\ CA^{T-1} & I \end{pmatrix}. \tag{4.21}$$

We consider $T = n + q$, the minimal time horizon needed to ensure observability in the nominal (attack-free) case. We want to study whether the matrix $\mathcal{O}'_T$ has full rank. Define the equation:

$$\mathcal{O}'_T \begin{pmatrix} x \\ a \end{pmatrix} = 0. \tag{4.22}$$

By applying a change of basis and row operations, the system can be written equivalently as:

$$\begin{pmatrix} C & I \\ \mathcal{O}_{T-1}(A-I) & 0 \end{pmatrix} \begin{pmatrix} x \\ a \end{pmatrix} = \begin{pmatrix} Cx + a \\ \mathcal{O}_{T-1}(A-I)x \end{pmatrix} = 0. \tag{4.23}$$

This yields the two conditions:

$$Cx + a = 0, \tag{4.24}$$

$$\mathcal{O}_{T-1}(A-I)x = 0. \tag{4.25}$$

Because (A, C) is observable, the matrix $\mathcal{O}_{T-1}$ has full column rank. Therefore,

$$\mathcal{O}_{T-1}v = 0 \iff v = 0.$$

However, from (4.25), we know that $(A-I)x = 0$ is a non-trivial equation due to the presence of an eigenvalue at 1, which implies that $A - I$ is not full rank. Therefore, there exist non-zero solutions $x \neq 0$ that satisfy $(A - I)x = 0$.

Plugging this into (4.24), we get:

$$a = -Cx,$$

which defines infinitely many possible values for a corresponding to the null space of $A - I$.

Hence, the kernel of $\mathcal{O}'_T$ is non-trivial: there exist non-zero vectors $\begin{pmatrix} x \\ a \end{pmatrix}$ such

that $\mathcal{O}'_T \begin{pmatrix} x \\ a \end{pmatrix} = 0$. This implies that $\mathcal{O}'_T$ does not have full column rank and the system is not observable.

□

Consequences for Observer Design

The above result highlights a critical limitation: the existence of a constant attack vector may make the augmented system unobservable, even if the system is observable in the attack-free scenario.

This situation is particularly critical when the system matrix A has an eigenvalue at 1. In these cases, classical observer designs, such as the Luenberger observer, fails, since they rely on the assumption of full observability of the augmented system.

Nevertheless, if additional structural information about the attack is available—for example, the knowledge that it is *sparse*—this limitation can be addressed by exploiting such information in the observer design. Sparse estimation techniques provide a natural framework in this context, as they aim to jointly recover both the system state and the attack vector by minimizing a specific norm (commonly the ℓ_1 norm) of the attack signal. By incorporating this sparsity assumption into the estimation process, it becomes possible to reconstruct the true state even when classical observability conditions fail. This insight motivates the development of *sparse observers*, a topic we explore in detail in the following sections.

4.4 Sparse Soft Observer for CPS with Sparse Sensor Attacks

In cyber-physical systems (CPS), sensor attacks can severely compromise the ability to accurately observe the system state. When such attacks are *sparse*, meaning only a small subset of sensors are affected at any time, it is possible to design observers that explicitly leverage this sparsity to improve state estimation accuracy. One effective approach in this context is the *Sparse Soft Observer* (SSO), which is closely related to optimization methods such as the *Iterative Shrinkage-Thresholding Algorithm* (ISTA).

4.4.1 Sparse Soft Observer (SSO)

The Sparse Soft Observer is a recursive algorithm that estimates both the system state and the sparse attack vector simultaneously. At each time step, it receives the system output, which is corrupted by the sparse attack, and updates its estimates according to the following rules:

- The state estimate is updated by applying the system dynamics and correcting it based on the residual between the predicted and measured outputs. This correction is modulated by a gain term that includes the system matrix.
- The attack estimate is updated via a *soft-thresholding* operation, which promotes sparsity by shrinking small components towards zero. This step exploits the knowledge that attacks are sparse.

Formally, the update equations can be written as:

$$\hat{x}(k+1) = A\hat{x}(k) - \nu AC^\top(\hat{y}(k) - y(k)),$$

$$\hat{a}(k+1) = S_{\nu\lambda}\left[\hat{a}(k) - \nu(\hat{y}(k) - y(k))\right],$$

where $\hat{y}(k) = C\hat{x}(k) + \hat{a}(k)$ is the predicted output, $\nu > 0$ is a step size parameter, λ controls the sparsity level, and $S_{\nu\lambda}$ is the element-wise soft-thresholding operator.

- **Initialization:** $\tau > 0$, $\hat{x}(0) \in \mathbb{R}^n$, $\hat{a}(0) \in \mathbb{R}^q$.
- Per $k = 0, \ldots, T_{\max}$:

$$\begin{aligned} y(k) &= Cx(k) + a(k) \\ \hat{y}(k) &= C\hat{x}(k) + \hat{a}(k) \\ \hat{x}(k+1) &= A\hat{x}(k) - \nu AC^\top(\hat{y}(k) - y(k)) \\ \hat{a}(k+1) &= \mathcal{S}_{\nu\lambda}\left[\hat{a}(k) - \nu(\hat{y}(k) - y(k))\right] \\ x(k+1) &= Ax(k) \end{aligned}$$

4.4.2 Relation to ISTA

When the system matrix A is the identity, the Sparse Soft Observer reduces exactly to the classical ISTA method for solving the underlying sparse recovery problem. ISTA performs a gradient descent step on the differentiable part of the objective function, followed by a proximal mapping via soft-thresholding to enforce sparsity in the attack estimate.

4.4.3 Deadbeat Sparse Soft Observer (D-SSO)

An upgraded version of the Sparse Soft Observer, called the *Deadbeat Sparse Soft Observer* (D-SSO), incorporates a correction gain L designed so that the eigenvalues of the matrix $(A - LC)$ are zero. This choice enables the state estimation error to be driven to zero in a finite number of steps, hence the term "deadbeat." The update equations for the D-SSO are:

$$\hat{x}(k+1) = A\hat{x}(k) - L(\hat{y}(k) - y(k)),$$

$$\hat{a}(k+1) = S_{\nu\lambda}\left[\hat{a}(k) - \nu(\hat{y}(k) - y(k))\right].$$

If the number of sensors q is at least equal to the state dimension n, and the gain L is chosen as the Moore-Penrose pseudo-inverse of C, denoted $C^{\dagger}$, then the D-SSO coincides with the *Inertial Jacobi Alternating Minimization* (IJAM) algorithm.

- **Initialization:** $\tau > 0$, $\hat{x}(0) \in \mathbb{R}^n$, $\hat{a}(0) \in \mathbb{R}^q$
- Per $k = 0, \ldots, T_{\max}$:

$$\begin{aligned}
y(k) &= Cx(k) + a(k) \\
\hat{y}(k) &= C\hat{x}(k) + \hat{a}(k) \\
\hat{x}(k+1) &= A\hat{x}(k) - L(\hat{y}(k) - y(k)), \quad \text{where } \operatorname{eig}(A - LC) = 0 \\
\hat{a}(k+1) &= \mathcal{S}_{\nu\lambda}\left[\hat{a}(k) - \nu(\hat{y}(k) - y(k))\right] \\
x(k+1) &= Ax(k)
\end{aligned}$$

4.4.4 Summary and Remarks

The development presented throughout this section highlights a conceptual and methodological transition: we started from optimization algorithms to manage the sparsity of the attack vector and gradually evolved toward a dynamic observ formulation suitable for cyber-physical system under sensor attacks.

The Sparse Soft Observer represents the non-linear observer counterpart of the Iterative Shrinkage-Threshlding Algorithm (ISTA), where the SSO explicitly incorporates the system dynamic through the state transition matrix A. The same considerations can also be made comparing the D-SSO with the Inertial Jacobi Alternating Minimization algorithm (IJAM).

4.4.5 Limitation and Further Developments

Despite their effectiveness and concept, the Sparse Soft Observer (SSO) and its deadbeat variant (D-SSO) present certain theoretical limitations that remain open challenges in the current literature.

Specifically, there is currently no formal evidence of both observers' stability, despite the fact that they both show excellent empirical performance in estimating the system state and the attack vectors. This challenge results from the nonlinearity of the estimation process, due to the soft-thresholding operator, which creates discontinuities and makes it impossible to use traditional linear system analysis tools directly.

The analysis is further complicated in dynamic settings, where the interaction between the state evolution and the sparsity-enforcing dynamics introduces additional coupling effects that challenge existing theoretical frameworks.

These open issues motivate the development of a new class of *nonlinear observers* specifically designed to retain the robustness and sparsity-awareness of the SSO while enabling a rigorous stability analysis. Unlike the analysis carried out in this section, the upcoming model does not rely on the assumption of constant attacks; instead, it is formulated to handle **general time-varying attack signals**, thereby extending the applicability of the observer to a broader range of adversarial scenarios.

The new observer architecture, presented in the next chapter, not only offers a theoretical foundation for convergence guarantees but also deepens our understanding of the dynamic behavior of sparsity-based estimation methods in cyber-physical systems under adversarial conditions.

Chapter 5

New theoretical formulation

In this section, we introduce a novel theoretical framework for the estimation of states and detection of sparse attacks in cyber-physical systems (CPSs). The proposed formulation extends classical observer designs by explicitly incorporating a structured attack estimation mechanism based on a shrinkage operator inspired by the Elastic Net regularization. This new approach enables the simultaneous estimation of the system state and the attack vector, while guaranteeing desirable convergence properties under sparsity assumptions.

First, we describe the dynamics of the CPS under consideration and introduce the nonlinear dynamics of the new observer. We then present fundamental theorems regarding the stability of the observer, which guarantee correct estimation of the state and the attack. Complete proofs of the theorems described above are provided.

We focus on a class of discrete-time linear cyber-physical systems (CPSs) described by:

$$\begin{cases} x(k+1) = Ax(k), \\ y(k) = Cx(k) + a(k). \end{cases} \tag{5.1}$$

where $k \in \mathbb{N}$ is the sampling instant, the state is $x(k) \in \mathbb{R}^n$, the measurement vector is $y(k) \in \mathbb{R}^p$, and the attack vector is $a(k) \in \mathbb{R}^p$. The system matrices are $A \in \mathbb{R}^{n \times n}$ and $C \in \mathbb{R}^{p \times n}$.

Each sensor $i \in \{1, \ldots, p\}$ provides a measurement $y_i(k)$; we assume that sensor i is under attack at time k if and only if $a_i(k) \neq 0$. The attack vector $a(k)$ is assumed to be sparse, meaning that only a small number of sensors are compromised at any given time step.

5.1 Novel formulation

In this section, we provide a detailed presentation of the proposed estimation scheme. We consider the following dynamic estimator:

$$\begin{cases} \hat{x}(k+1) = A\hat{x}(k) + L_1\tilde{y}(k), \\ \hat{a}(k+1) = \Psi_q(\hat{a}(k) - \epsilon L_2\tilde{y}(k)), \\ \tilde{y}(k) = C(\hat{x}(k) - x(k)) + \hat{a}(k) - a(k). \end{cases} \tag{5.2}$$

where $\hat{x} \in \mathbb{R}^n$ denotes the estimated system states, and $\hat{a} \in \mathbb{R}^p$ represents the estimated attack vector, which shares the same dimension as the measured output $y \in \mathbb{R}^p$ and the estimated output $\hat{y} \in \mathbb{R}^p$. The output mismatch is defined as $\tilde{y} = \hat{y} - y \in \mathbb{R}^p$.

The matrix $L_1 \in \mathbb{R}^{n\times p}$ represents the gain matrix of the linear observer dynamics, while the matrix $L_2 \in \mathbb{R}^{p\times p}$ governs the nonlinear dynamics of the attack estimate. The nonlinear dynamics is related to the function Ψ_q, which is described below.

The estimator is inspired by the (strongly convex) Elastic Net model in 4.13, which augments the Lasso formulation with a Tikhonov regularization term:

$$(\hat{x}, \hat{a}) = \arg\min_{x\in\mathbb{R}^n,,a\in\mathbb{R}^{p\tau}} \frac{1}{2}\left\|y - Cx - a\right\|_2^2 + \lambda\left\|a\right\|_1 + \frac{\mu}{2}\left\|a\right\|_2^2, \tag{5.3}$$

where $\lambda > 0$ and $\mu > 0$ are regularization parameters. This formulation enforces sparsity through the ℓ_1-norm while maintaining strong convexity via the ℓ_2-term, thereby enhancing numerical stability and convergence.

The associated shrinkage operator is defined as

$$\Psi_q(z) = \frac{1}{1+\delta}\Psi_1(z) = q \cdot \Psi_1(z), \tag{5.4}$$

with $\delta > 0$, where

$$q := \frac{1}{1+\delta} \tag{5.5}$$

quantifies the contribution of the ℓ_2-regularization term. the nonlinear function $\Psi_1(\cdot)$ is the soft-thresholding operator presented in 4.12. A larger δ (equivalently, larger μ in (5.3)) leads to a stronger Elastic Net effect with increased shrinkage, while smaller δ values yield behavior closer to the Lasso case ($q \to 1$).

5.1.1 Assumptions

In order to move forward, the following assumptions are needed.

Assumption 1. *The couple (A, C) is observable.*

Assumption 2. *The following properties are satisfied by the function $\Psi_q : \mathbb{R}^n \to \mathbb{R}^p$:*

1. $\Psi_q(0) = 0$;
2. *there exists a symmetric positive definite matrix $Q \in \mathbb{R}^{n\times n}$ such that, for all $s \in \mathbb{R}^n$,*
$$\Psi_q(s)^\top Q \Psi_q(s) < s^\top Q s;$$
3. *Ψ_q is Lipschitz continuous, i.e., there exists a constant $l_\Psi > 0$ such that, for all $s, \hat{s} \in \mathbb{R}^n$,*
$$\|\Psi_q(s) - \Psi_q(s + \hat{s})\| \leq l_\Psi \|\hat{s}\|.$$

Assumption 3. *There exists a matrix L_2 such that the following inequality holds for all admissible $\hat{a}$:*

$$\Psi_q(\hat{a} - \varepsilon L_2(C\Pi(\hat{a}) + \hat{a}))^\top \, Q \, \Psi_q(\hat{a} - \varepsilon L_2(C\Pi(\hat{a}) + \hat{a})) \leq \rho_a'' \, \hat{a}^\top Q \hat{a},$$

where $0 < \rho_a'' < 1$.

5.1.2 Main results

Theorem 6. *Suppose Assumptions 1–3 hold and $a(k) = 0$ for all k. Then, there exist observer gains L_1 and L_2 such that the estimation errors*

$$e(k) = \hat{x}(k) - x(k), \quad \hat{a}(k)$$

are globally asymptotically stable *(GAS) at the origin:*

$$\lim_{k\to\infty} e(k) = 0, \quad \lim_{k\to\infty} \hat{a}(k) = 0.$$

Theorem 7 (ISS with Respect to Attack)**.** *Suppose Assumptions 1–3 hold. Then, for any bounded attack $a(k) \neq 0$, the estimation errors $(e(k), \hat{a}(k))$ satisfy a discrete-time* Input-to-State Stability (ISS) *property with respect to $a(k)$:*

$$\|e(k)\| + \|\hat{a}(k)\| \leq \beta(\|e(0)\| + \|\hat{a}(0)\|, k) + \gamma\Big(\sup_{0\leq j\leq k} \|a(j)\|\Big), \quad \forall k \geq 0,$$

for some $\beta \in \mathcal{KL}$ and $\gamma \in \mathcal{K}$.

5.1.3 Proof of the main results

To simplify the notation and improve readability, we adopt the following convention: all variables evaluated at time k will be denoted without the time index (e.g., $a(k) = a$), while variables at time $k+1$ will be indicated with a superscript "+" (e.g., $a(k+1) = a^+$).

Preliminaries

To analyze the convergence properties of the estimator, define the state estimation error

$$e := \hat{x} - x. \tag{5.6}$$

The error dynamics can be written as

$$\begin{aligned} e^+ &= \hat{x}^+ - x^+ \\ &= A\,\hat{x} + L_1\big(C(\hat{x} - x) + \hat{a}\big) + v_x - A\,x \\ &= (A + L_1 C)\,e + L_1\hat{a} - L_1 a + v_x. \end{aligned} \tag{5.7}$$

Introduce the change of variable

$$\eta = e - \Pi(\hat{a}), \tag{5.8}$$

where η represents the estimation error adjusted by a nonlinear mapping $\Pi(\cdot)$ of the estimated attack $\hat{a}$.

Lemma 1 (Existence and Lipschitz continuity of Π). There exists a function $\Pi : \mathbb{R}^p \to \mathbb{R}^n$ such that

$$(A + L_1 C)\Pi(\hat{a}) + L_1\hat{a} - \Pi\big(\Psi_q(\hat{a})\big) = 0, \tag{5.9}$$

and Π is Lipschitz continuous with constant

$$\ell_\Pi = \frac{\|L_1\|}{|\lambda_{\max}(A + L_1 C) - q|},$$

where $q = \ell_\Psi$ is the Lipschitz constant of Ψ_q.

Proof. Define $F := A + L_1 C$. Consider the series

$$\Pi(\hat{a}) = -\sum_{l=0}^{\infty} F^{-l-1} L_1 \Psi_q^l(\hat{a}). \tag{5.10}$$

Verification of the equation:

$$\begin{aligned}\Pi(\Psi_q(\hat{a})) &= -\sum_{l=0}^{\infty} F^{-l-1} L_1 \Psi_q^{l+1}(\hat{a}) \\ &= -\sum_{l=1}^{\infty} F^{-l} L_1 \Psi_q^{l}(\hat{a}) \\ &= F\Pi(\hat{a}) + L_1\hat{a}.\end{aligned}$$

Lipschitz continuity:

$$\begin{aligned}\Pi(\hat{a}) &= -\sum_{l=0}^{\infty} F^{-l-1} L_1 \Psi_q^{l}(\hat{a}) \\ \|\Pi(a_1) - \Pi(a_2)\| &= \left\| \sum_{l=0}^{\infty} F^{-l-1} L_1 \left(\Psi_q^l(a_2) - \Psi_q^l(a_1)\right) \right\| \\ &\leq \sum_{l=0}^{\infty} \left\|F^{-l-1}\right\| \|L_1\| \left\|\Psi_q^l(a_2) - \Psi_q^l(a_1)\right\| \\ &\leq \sum_{l=0}^{\infty} \left| \frac{\ell_\Psi^l}{(\lambda_{\max}(F))^{l+1}} \right| \|L_1\| \|a_1 - a_2\|\end{aligned}$$

Given that the Lipschitz constant of the operator Ψ_q is $\ell_\Psi = \frac{1}{1+\delta} = q$, the k-th iterate Ψ_q^l is Lipschitz continuous with constant ℓ_Ψ^k.

$$\begin{aligned}&\leq \left[\sum_{l=0}^{\infty} \left(\frac{q}{\lambda_{\max}(F)}\right)^l\right] \frac{\|L_1\|}{\lambda_{\max}(F)} \|a_1 - a_2\| \\ &\leq \left[\sum_{l=0}^{\infty} \left(\frac{q}{\lambda_{\max}(F)}\right)^l\right] \frac{\|L_1\|}{\lambda_{\max}(F)} ||a_1 - a_2||\end{aligned}$$

The geometric series converges when $\frac{q}{\lambda_{\max}(F)} < 1$, and in this case we obtain:

$$|\Pi(a_1) - \Pi(a_2)| \leq \left| \frac{1}{1 - \frac{q}{\lambda_{\max}(F)}} \right| \frac{\|L_1\|}{\lambda_{\max}(F)} ||a_1 - a_2||$$

The Lipschitz constant of the function Π can be computed as follows:

$$\begin{aligned}\ell_{\Pi} &= \left|\frac{1}{1-\frac{q}{\lambda_{\max}(F)}}\right| \frac{\|L_1\|}{\lambda_{\max}(F)} \\ &= \frac{||L_1||}{|\lambda_{\max}(F)-q|}\end{aligned} \tag{5.11}$$

□

5.1.4 Globally asymptotically stability when $a = 0$

Proof. Consider the equation (5.2), when a is set to zero, then by applying the change of coordinate the system is as follow:

$$\begin{aligned}\eta^+ &= (A + L_1C)\left[\eta + \Pi(\hat{a})\right] + L_1\hat{a} - \Pi\left(\Psi_q(\hat{a} - \varepsilon L_2\tilde{y})\right), \\ \hat{a}^+ &= \Psi_q\left(\hat{a} - \varepsilon L_2\tilde{y}\right), \\ \tilde{y} &= Ce + \hat{a} = C\left(\eta + \Pi(\hat{a})\right) + \hat{a}.\end{aligned} \tag{5.12}$$

We can observe that:

$$\varepsilon L_2\tilde{y} = \varepsilon L_2\left(C\eta + C\Pi(\hat{a}) + \hat{a}\right), \tag{5.13}$$

In this case, the system evolves as:

$$\begin{aligned}\eta^+ &= (A + L_1C)\eta + \Delta_\eta(\eta, \hat{a}), \\ \hat{a}^+ &= \Psi_q\left[\hat{a} - \varepsilon L_2(C\Pi(\hat{a}) + \hat{a})\right] + \Delta_a(\eta, \hat{a}),\end{aligned} \tag{5.14}$$

where the perturbation terms are defined as:

$$\begin{aligned}\Delta_\eta &= \Pi\left(\Psi_q(\hat{a})\right) - \Pi\left(\Psi_q\left(\hat{a} - \varepsilon L_2(C\eta + C\Pi(\hat{a}) + \hat{a})\right)\right), \\ \Delta_a &= \Psi_q\left[\hat{a} - \varepsilon L_2(C\eta + C\Pi(\hat{a}) + \hat{a})\right] - \Psi_q\left[\hat{a} - \varepsilon L_2(C\Pi(\hat{a}) + \hat{a})\right].\end{aligned} \tag{5.15}$$

Using the Lipschitz continuity of the functions involved, we have:

$$\begin{aligned}&\|\Delta_\eta\| = \|\Pi\left(\Psi_q\left(\hat{a}\right)\right) - \Pi\left(\Psi_q\left(b\right)\right)\| \leq \ell_\Pi\left\|\Psi_q\left(\hat{a}\right) - \Psi_q\left(b\right)\right\| \leq \ell_\Pi\ell_{\Psi_q}\left\|\hat{a} - b\right\| \\ &\|\hat{a} - b\| = \|\varepsilon L_2\left(C\eta + C\Pi\left(\hat{a}\right) + \hat{a}\right)\| \leq \varepsilon\ell_2\left(\ell_C\left\|\eta\right\| + (1 + \ell_c\ell_\Pi)\left\|\hat{a}\right\|\right) \\ &\|\Delta_\eta\| \leq \varepsilon\ell_\Pi\ell_{\Psi_q}\ell_2\left(\ell_C\left\|\eta\right\| + (1 + \ell_c\ell_\Pi)\left\|\hat{a}\right\|\right)\end{aligned} \tag{5.16}$$

From the definition of Δ_a, we write:

$$\Delta_a = \Psi_q \underbrace{[\hat{a} - \varepsilon L_2(C\eta + C\Pi(\hat{a}) + \hat{a})]}_{b_1} - \Psi_q \underbrace{[\hat{a} - \varepsilon L_2(C\Pi(\hat{a}) + \hat{a})]}_{b_2}$$

Again, using Lipschitz continuity:

$$\begin{aligned} &\|\Delta_a\| = \|\Psi_q(b_1) - \Psi_q(b_2)\| \leq \ell_{\Psi_q} \|b_1 - b_2\| \\ &\|b_1 - b_2\| = \|\varepsilon L_2 C\eta\| \leq \varepsilon \ell_2 \ell_C \|\eta\| \\ &\|\Delta_a\| \leq \varepsilon \ell_{\Psi_q} \ell_2 \ell_C \|\eta\| \end{aligned} \tag{5.17}$$

We consider the Lyapunov function:

$$W = \eta^\top P \eta + \mu \hat{a}^\top Q \hat{a}, \tag{5.18}$$

with $P > 0$, $Q > 0$, and $\mu > 0$.

At this point, we have to introduce an additional assumption:

We analyze its difference along the trajectories:

$$\begin{aligned} W^+ - W = &\underbrace{\eta^\top (A + L_1 C)^\top P (A + L_1 C)\eta - \eta^\top P \eta} \\ &+ 2\eta^\top (A + L_1 C)^\top P \Delta_\eta + \Delta_\eta^\top P \Delta_\eta \\ &+ \underbrace{\mu \Psi_q(\hat{a} - \varepsilon L_2(\Pi(\hat{a}) + \hat{a}))^\top Q \Psi_q(\hat{a} - \varepsilon L_2(\Pi(\hat{a}) + \hat{a})) - \mu \hat{a}^\top Q \hat{a}} \\ &+ 2\mu \Psi_q(\cdot)^\top Q \Delta_a + \mu \Delta_a^\top Q \Delta_a. \end{aligned} \tag{5.19}$$

$$\begin{aligned} W^+ - W = &(1 + \gamma_\eta)\eta^\top (A + L_1 C)^\top P (A + L_1 C)\eta - \eta^\top P \eta \\ &+ (1 + \frac{1}{\gamma_\eta}) \Delta_\eta^\top P \Delta_\eta \\ &+ (1 + \gamma_a)\mu \Psi_q(\hat{a} - \varepsilon L_2(\Pi(\hat{a}) + \hat{a}))^\top Q \Psi_q(\hat{a} - \varepsilon L_2(\Pi(\hat{a}) + \hat{a})) - \mu \hat{a}^\top Q \hat{a} \\ &+ (1 + \frac{1}{\gamma_a}) \mu \Delta_a^\top Q \Delta_a. \end{aligned}$$

Rewriting the Lyapunov function:

$$\Delta W \leq -\gamma \lambda_{\max}(P) \|\eta\|^2 + \varepsilon c_\eta \|\eta\|^2 + \varepsilon c_a \|\hat{a}\|^2 - \mu (1 - \rho_a'') \lambda_{max}(Q) \|\hat{a}\|^2 + \mu \varepsilon d_\eta \|\eta\|^2,$$

where:

$$
\begin{aligned}
c_\eta &= 2(1+\frac{1}{\gamma_\eta})\|P\|\ell_\Pi^2\ell_{\Psi_q}^2\ell_2^2\ell_C^2 \\
c_a &= 2(1+\frac{1}{\gamma_\eta})\|P\|\ell_\Pi^2\ell_{\Psi_q}^2\ell_2^2\left(1+\ell_C\ell_\Pi\right)^2 \\
d_\eta &= d_\eta = (1+\frac{1}{\gamma_a})\,\|Q\|\,\ell_{\Psi_q}^2\ell_2^2\ell_C^2 \\
\rho_a'' &= \frac{\rho_a}{(1+\gamma_a)} \\
\gamma &= \left(\gamma_\eta(\rho-1)+\rho\right).
\end{aligned}
$$

Thus, we ensure $W^+ - W < 0$ for all $\eta, \hat{a}$, provided:

$$
\varepsilon < \varepsilon^* = \min\left\{\frac{\gamma\lambda_{max}(P)}{c_\eta+\mu d_\eta}, \frac{\mu\left(1-\rho_a''\right)\lambda_{max}(Q)}{c_a}\right\}. \tag{5.20}
$$

Hence, for a sufficiently small gain parameter ε, the observer guarantees asymptotic convergence to the origin. □

5.1.5 Verification of assumption (3)

The goal is to determine for which values of L_2 assumption (3) holds.

Proof. This estimate exploits the following property of the soft-thresholding operator:

$$
\Psi_q\left(s\right)^\top Q\Psi_q\left(s\right) \leq q^2 s^\top Qs. \tag{5.21}
$$

The equation becomes:

$$
\left((I-\varepsilon L_2)\hat{a}-\varepsilon L_2 C\Pi(\hat{a})\right)^\top Q\left((I-\varepsilon L_2)\hat{a}-\varepsilon L_2 C\Pi(\hat{a})\right) \leq \frac{\rho''}{q^2}\hat{a}^\top Q\hat{a},
$$

We select L_2:

$$
L_2 = \min\left\{\frac{1}{2\ell_C\ell_\Pi}, 1\right\} I. \tag{5.22}
$$

$$
\left(\left(\frac{2|C|\ell_\Pi-\varepsilon}{2|C|\ell_\Pi}\right)\hat{a}-\varepsilon\frac{1}{2|C|\ell_\Pi}C\Pi(\hat{a})\right)^\top Q\left(\left(\frac{2|C|\ell_\Pi-\varepsilon}{2|C|\ell_\Pi}\right)\hat{a}-\varepsilon\frac{1}{2|C|\ell_\Pi}C\Pi(\hat{a})\right) \leq \frac{\rho''}{q^2}\hat{a}^\top Q\hat{a}.
$$

$$
\rho_2(\varepsilon) = \frac{2\ell_C\ell_\Pi-\varepsilon}{2\ell_C\ell_\Pi} = 1-\frac{\varepsilon}{2\ell_C\ell_\Pi} < 1,
$$

$$\left(\rho_2(\varepsilon)\hat{a} - \varepsilon\frac{1}{2\ell_C\ell_\Pi}C\Pi(\hat{a})\right)^\top Q\left(\rho_2(\varepsilon)\hat{a} - \varepsilon\frac{1}{2\ell_C\ell_\Pi}C\Pi(\hat{a})\right) \leq \frac{\rho''}{q^2}\hat{a}^\top Q\hat{a}.$$

$$\begin{aligned}
&\rho_2(\varepsilon)^2\hat{a}^\top Q\hat{a} + \frac{\varepsilon^2}{4\ell_C^2\ell_\Pi^2}\Pi(\hat{a})^\top C^\top QC\Pi(\hat{a}) - \frac{\varepsilon}{\ell_C\ell_\Pi}\rho_2(\varepsilon)\hat{a}^\top QC\Pi(\hat{a}) \\
\leq\ &\rho_2(\varepsilon)^2\hat{a}^\top Q\hat{a} + \frac{\varepsilon^2}{4\ell_C^2\ell_\Pi^2}\Pi(\hat{a})^\top C^\top QC\Pi(\hat{a}) + \frac{\varepsilon^2}{\zeta\ell_C^2\ell_\Pi^2}\Pi(\hat{a})^\top C^\top QC\Pi(\hat{a}) + \rho_2(\varepsilon)^2\zeta\hat{a}^\top Q\hat{a} \\
&\leq \left((1+\zeta)\rho_2(\varepsilon)^2\right)\hat{a}^\top Q\hat{a} \leq \frac{\rho''}{q^2}\hat{a}^\top Q\hat{a}.
\end{aligned} \tag{5.23}$$

Remark. The inequality above is satisfied because both multiplicative factors, $(1+\zeta)\rho_2(\varepsilon)^2$ and $\frac{\rho''}{q^2}$, are strictly less than one for appropriately chosen parameters. In particular, by selecting the observer gain $\varepsilon > 0$ sufficiently small, together with suitable choices of the constant q of the shrinkage operator and the weighting associated with the ℓ_2-norm, the left-hand side can be made smaller than the right-hand side while remaining positive definite.

Consequently, Assumption (3) is fully satisfied, ensuring the existence of the observer gain L_2 and the associated contraction property. This demonstrates that the suggested design guarantees the required convergence and stability characteristics.

□

5.1.6 ISS propriety when $a \neq 0$

The above ISS result provides a general discrete-time framework. In particular, it applies directly to our estimator dynamics (5.2) when the attack $a(k) \neq 0$. Specifically, by defining the state as $x = (\eta, \hat{a})$ and the input as $v = a$, and observing that the nominal system (with $a = 0$) is GAS, the Lipschitz continuity of the perturbation terms Δ_η and Δ_a ensures that all the conditions of the theorem are satisfied. Therefore, the estimation errors $(\eta(k), \hat{a}(k))$ satisfy a discrete-time ISS property with respect to the attack $a(k)$.

Theorem 8. *Let $f : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^n$ such that the following is verified:*

- *There exist $P = P^\top \succ 0$, $\rho \in (0,1)$ such that*

$$f(x,0)^\top P f(x,0) < \rho\, x^\top P x \qquad \forall x \in \mathbb{R}^n$$

- *there exists* $\ell > 0$ *such that*

$$|f(x,v) - f(x,0)| \leq \ell|v|$$
$$|f(x,v)| \leq \ell|x| + \ell|v|$$

Then, system $x^+ = f(x,v)$ *is ISS w.r.t* v*.*

Proof. Pick $V = x^\top P x$. Then

$$\begin{aligned}
V^+ - V &= f(x,v)^\top P f(x,v) - x^\top P x \\
&= f(x)^\top P f(x) - x^\top P x + \Big(f(x,v)^\top P f(x,v) - f(x)^\top P f(x)\Big) \\
&\leq -(1-\rho)x^\top P x + \Big(f(x,v)^\top P f(x,v) - f(x,v)^\top P f(x) \\
&\qquad + f(x,v)^\top P f(x) - f(x)^\top P f(x)\Big) \\
&\leq -(1-\rho)x^\top P x + f(x,v)^\top P\big(f(x,v)) - f(x)\big) \\
&\qquad + f(x)^\top P\big(f(x,v)) - f(x)\big) \\
&\leq -(1-\rho)x^\top P x + \ell(|x| + |v|)|P|\ell|v| + \ell^2|x||P||v| \\
&\leq -(1-\rho)x^\top P x + \zeta|x|^2 + \frac{1}{2\zeta}\ell^2|P|(2\ell^2|P| + 1)|v|^2 \\
&\leq -\left(1 - \rho - \frac{\zeta}{\lambda_{\min}(P)}\right) x^\top P x + \frac{1}{2\zeta}\gamma|v|^2
\end{aligned}$$

with $\gamma = \ell^2|P|(2\ell^2|P| + 1)$. Selecting $\zeta = \dfrac{1-\rho}{2}\lambda_{\min}(P)$ we obtain

$$V^+ - V < -\frac{1-\rho}{2}V + \bar{\gamma}|v|^2$$

with $\bar{\gamma} = \gamma\zeta$. □

Chapter 6

Numerical simulations

6.1 Dead-beat sparse soft observer

In this section, we present numerical simulations to assess the effectiveness of the Deadbeat Sparse Soft Observer (D-SSO) in estimating the state of a cyber-physical system (CPS) under sparse sensor attacks. The results demonstrate the observer's ability to accurately reconstruct both the system state and the attack signal.

6.1.1 Simulation Setup

We consider a discrete-time linear system of the form:

$$x(k+1) = Ax(k), \quad y(k) = Cx(k) + a(k),$$

where $x(k) \in \mathbb{R}^n$ is the state vector, $y(k) \in \mathbb{R}^q$ is the output, and $a(k) \in \mathbb{R}^q$ is the sparse attack vector.

- System dimension: $n = 15$, number of sensors: $q = 30$
- The matrices $A \in \mathbb{R}^{n \times n}$ and $C \in \mathbb{R}^{q \times n}$ are randomly generated with independent and identically distributed entries following a standard normal distribution, i.e., $\mathcal{N}(0, 1)$.

  To ensure that the cyber-physical system (CPS) is marginally stable, the matrix A has one eigenvalue equal to 1. This procedure allows us to analyze the performance of the observer in a scenario where the system is neither asymptotically stable nor unstable, which is particularly relevant when evaluating the impact of constant or sparse sensor attacks.

- The attack vector $a(k)$ is constant and sparse ($h = 3$), with non-zero entries at randomly selected sensor positions.
- The D-SSO is initialized with $\hat{x}(0) = 0$, $\hat{a}(0) = 0$.
- The observer gain L is selected such that $\text{eig}(A - LC) = 0$.
- Concerning the hyperparameters, we empirically set $\nu = 0.7$ and $\lambda = 10^{-1}$.

6.1.2 Algorithm

Algorithm 1: Dynamic Sparse Soft Observer (D-SSO)

Input: $\lambda > 0, \nu > 0, A, C, L_1$ such that all eigenvalues of $A - L_1 C$ are null

Output: $\begin{pmatrix} \hat{x}(k) \\ \hat{a}(k) \end{pmatrix}$ = estimate of $\begin{pmatrix} x(k) \\ a(k) \end{pmatrix}$

Initialization: $\hat{x}(0) = 0,\ \hat{a}(0) = 0$;

for $k = 0, 1, \ldots$ **do**

  Acquire new measurement: $y(k) = Cx(k) + a(k)$;

  Predicted measurement using current estimates: $\hat{y}(k) = C\hat{x}(k) + \hat{a}(k)$;

  State estimation update: $\hat{x}(k+1) = A\hat{x}(k) - L_1 \left(\hat{y}(k) - y(k)\right)$;

  Attack estimation update: $\hat{a}(k+1) = S_\lambda \left(\hat{a}\left(k\right) - \nu \left(\hat{y}\left(k\right) - y\left(k\right)\right)\right)$

end

6.1.3 Results

Figure 6.1 shows the mean corresponding error on the attack and state vectors over 50 simulations. For clarity, the state estimation error is presented on a logarithmic scale.

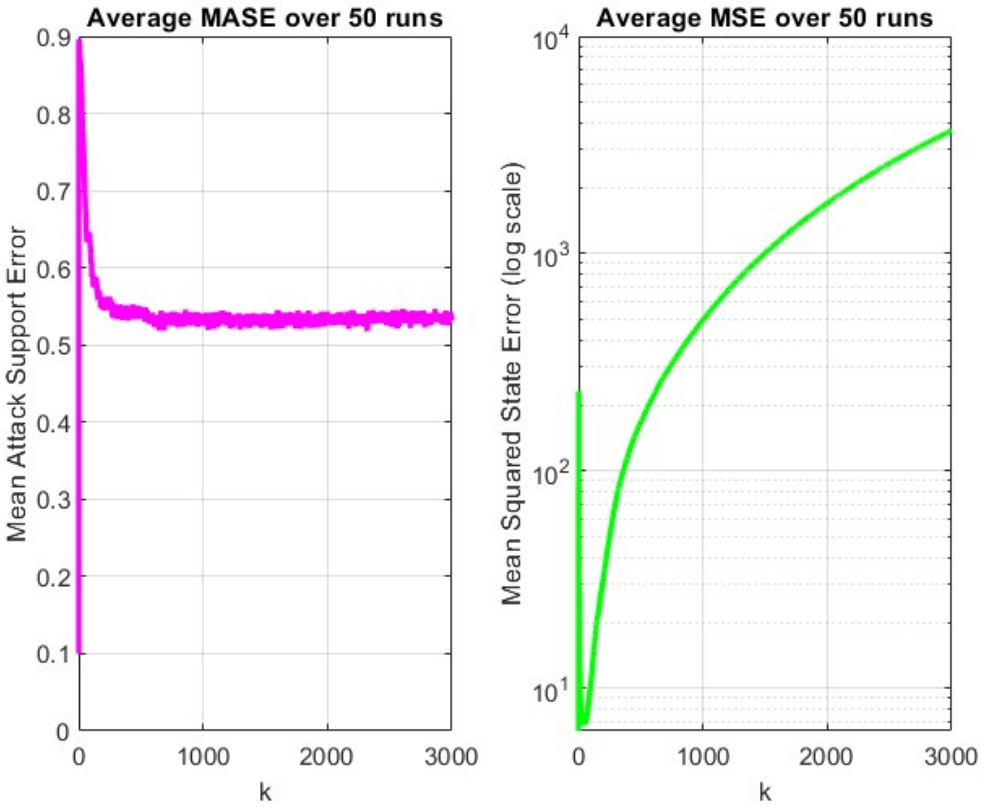

Figure 6.1: On the left the MASE, on the right the MSE for the DSSO algorithm in case of marginally stable case

6.1.4 Discussion

One of the main problems when using the Deadbeat Sparse Soft Observer (D-SSO) lies in the proper tuning of the parameters involved, namely the soft-thresholding parameter λ and the step size ν. If these parameters are not selected appropriately, the estimation of both the system state and the attack vector may become inaccurate.

In particular, the step size ν affects the convergence and stability of the observer. An improper choice may lead to slow convergence or divergence of the estimation process. Since the simulation plots the mean over 50 runs, it can happen that for some system scenarios the chosen parameter pair works quite well; however, there are cases in which the parameter pair is not suitable for estimation purposes. Unfortunately, in the literature, there is no proper theoretical framework for computing these parameters. Therefore, a proper calibration of these parameters is essential to ensure reliable performance of the observer.

6.2 New theoretical formulation

In order to verify and demonstrate the novel theoretical formulation presented in Chapter 5, we provide a number of numerical simulations in this section. The purpose of these simulations is to clearly compare the suggested method with the Dynamic Sparse State Observer (D-SSO) method and show how well it performs under different circumstances.

We consider a variety of scenarios that reflect realistic operating conditions, including:

- **No attack:** a baseline scenario in which the system operates without malicious interference, serving as a reference case for performance evaluation. In this scenario, the initial condition of the attack estimate is not null, but a random value, allowing us to investigate the observer's transient behavior and its convergence properties in the absence of attacks.
- **Constant attack:** a scenario where a persistent attack signal affects the system, representing a steady adversarial disturbance.
- **Time-varying attack support**:a scenario that mimics a more realistic adversarial strategy where the attack's support changes over time after a predetermined number of iterations.
- **Sinusoidal attack:** is a time-varying attack that mimics a dynamic and more difficult adversarial condition by having sinusoidal characteristics.

By analyzing the errors in state and attack estimation, we are able to evaluate the effectiveness of the observer presented in Chapter 5 in all four scenarios.

Algorithm 2: L2 based observer

Input: $\lambda > 0, A, C, L_1, L_2, q \in (0,1)$, Let $\varepsilon > 0$ be sufficiently small

Output: $\begin{pmatrix} \hat{x}(k) \\ \hat{a}(k) \end{pmatrix}$ = estimate of $\begin{pmatrix} x(k) \\ a(k) \end{pmatrix}$

Initialization: $\hat{x}(0) = 0,\ \hat{a}(0) = 0$ **for** $k = 0, 1, \ldots$ **do**

  Acquire new measurement: $y(k) = Cx(k) + a(k)$

  Predicted measurement using current estimates: $\hat{y}(k) = C\hat{x}(k) + \hat{a}(k)$

  State estimation update: $\hat{x}(k+1) = A\hat{x}(k) + L_1\left(\hat{y}(k) - y(k)\right)$

  Attack estimation update: $\hat{a}(k+1) = q \cdot S_\lambda\left(\hat{a}\left(k\right) - \varepsilon L_2\left(\hat{y}\left(k\right) - y\left(k\right)\right)\right)$

end

6.2.1 Numerical simulation

In this section, we present numerical simulations to assess the effectiveness of the observer involving the soft-thresholding operator in estimating the state of a cyber-physical system (CPS) under sparse sensor attacks. The results demonstrate the observer's ability to accurately reconstruct both the system state and the attack signal.

6.2.2 Common Simulation Setup

We consider a discrete-time linear system of the form:

$$x(k+1) = Ax(k), \quad y(k) = Cx(k) + a(k),$$

where $x(k) \in \mathbb{R}^n$ is the state vector, $y(k) \in \mathbb{R}^q$ is the output, and $a(k) \in \mathbb{R}^q$ is the sparse attack vector.

To ensure that the cyber-physical system (CPS) is marginally stable, the matrix A is constructed so that it has one eigenvalue equal to 1. This construction procedure allows us to analyze the performance of the observer in a scenario where the system is neither asymptotically stable nor unstable, which is particularly relevant when evaluating the impact of constant or sparse sensor attacks.

- A systematic exploration of the hyperparameter λ and of the gain L_2 was carried out with the objective of identifying the combination that yields the minimal estimation error. As regards the L_2 parameter, a research was carried out around the theoretical value. All simulations were compared with those performed with the DSSO, and to have the most reasonable comparison possible, a grid search was also performed for the DSSO parameters.

- To evaluate the error, we introduce a cost function defined as a convex combination of the state estimation error and the attack estimation error according to a parameter α:

$$\text{cost} = \alpha \cdot \text{MSE} + (1 - \alpha) \cdot \text{MASE} \tag{6.1}$$

  where:

  - **MSE**: Mean Squared Error (state estimation accuracy),
  - **MASE**: Mean Attack Support Error (attack support recovery).

By introducing the parameter $\alpha \in [0, 1]$, the relative importance of the attack support error (MASE) and the state estimation error (MSE) can be adjusted. Specifically, α values near 1 highlight the precision of reconstructing the system state, whereas α values near 0 give priority to accurately identifying the attacked sensors.

Remark on the computation of the errors The error on the state is quantified through the Mean Squared Error (MSE), computed as the average of the squared differences between the true state and its estimate, i.e., $\text{MSE} = \text{mean}\big((x - \hat{x})^2\big)$. For the attack vector, instead, the interest lies not in its exact numerical values but in identifying which sensors are effectively under attack. In other words, the goal is to correctly recover the support of the true attack vector. To this end, we define the Mean Attack Support Error (MASE) as a measure of the mismatch between the true and the estimated supports. A particular case occurs when all the entries of the estimated attack vector are zero: in this situation, the error is set to 1, since we know that the true attack vector has exactly three nonzero components ($\|a\|_0 = 3$). This is not true only in the case where no attacks corrupt the system. By construction, the value of MASE ranges between 0 and 1, where values close to zero indicate a good detection of the attacked sensors, while values close to one correspond to a complete failure in identifying them.

6.2.3 No attacks

In this scenario, the attack vector $a(k)$ is assumed to be zero, and to evaluate the ability of the observer to converge to zero, we initialize it with a random estimate of the attack support.

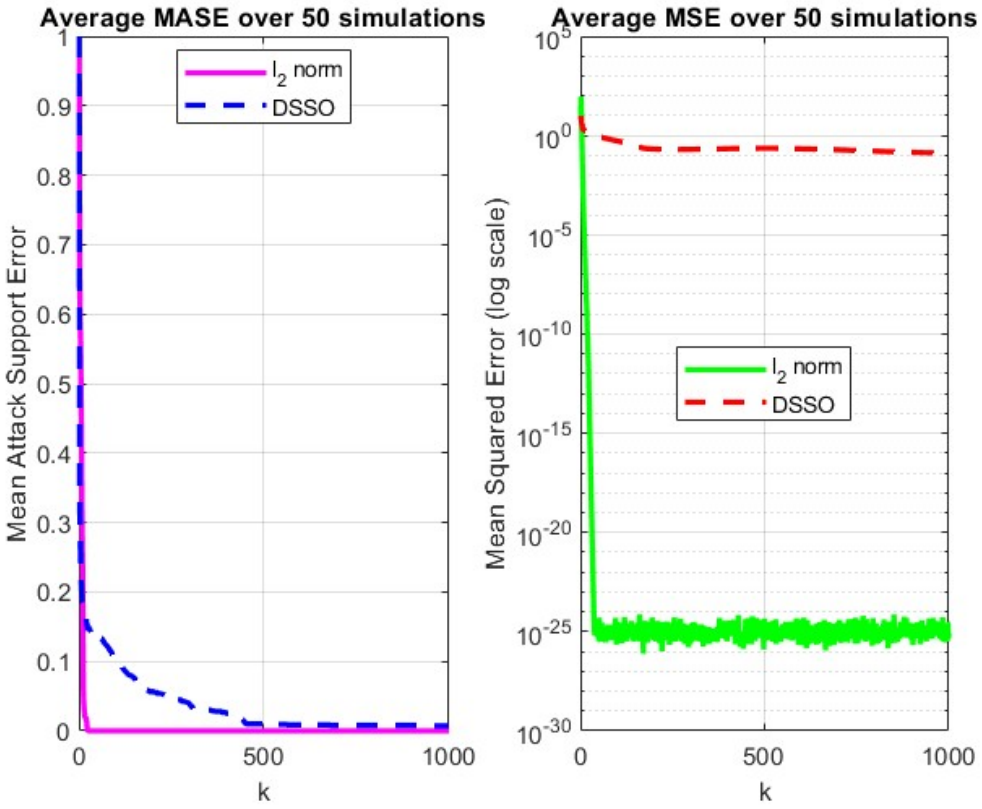

Figure 6.2: On the left, the comparison of the MASE between the D-SSO and the new theoretical approach is shown, while on the right, the corresponding comparison of the MSE on the estimated state is displayed, in the case without any attacks.

6.2.4 Constant attack

In this scenario, the attack vector $a(k)$ is assumed to be constant and sparse ($h = 3$), with non-zero entries at randomly selected sensor positions.

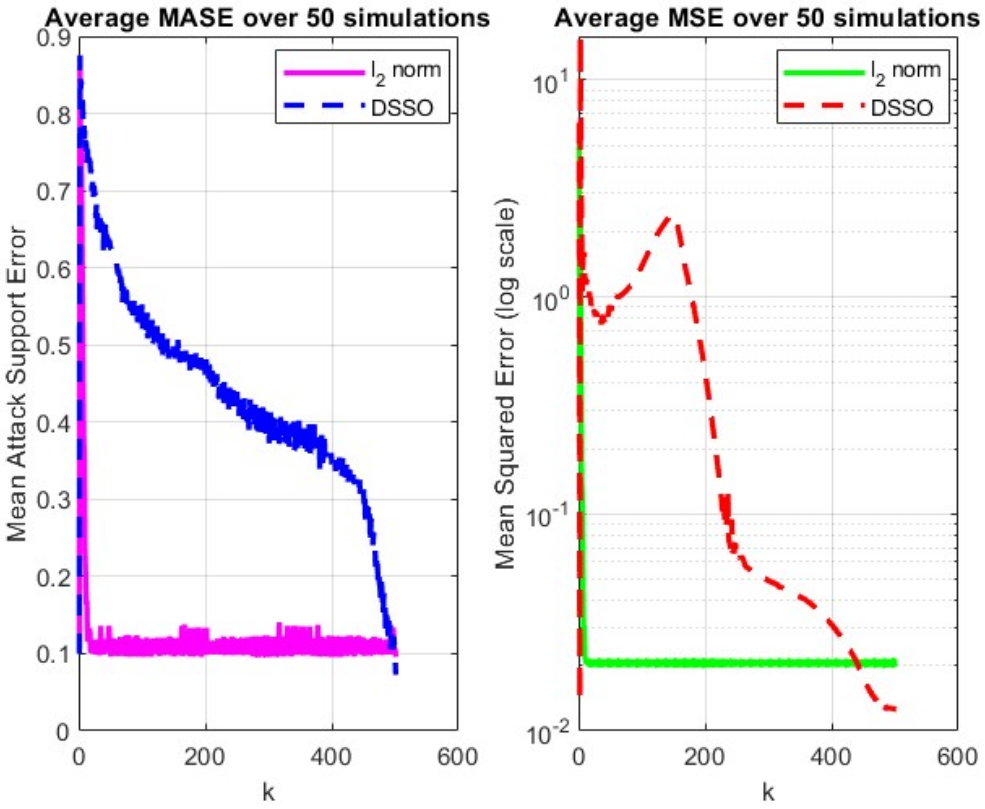

Figure 6.3: On the left, the comparison of the MASE between the D-SSO and the new theoretical approach is shown, while on the right, the corresponding comparison of the MSE on the estimated state is displayed, in the case of constant attack.

6.2.5 Time-varying attack support

In this scenario, the support of the attack vector $a(k)$ is assumed to change randomly every 400 iterations, while the attack vector itself remains piecewise constant and sparse ($h = 3$).

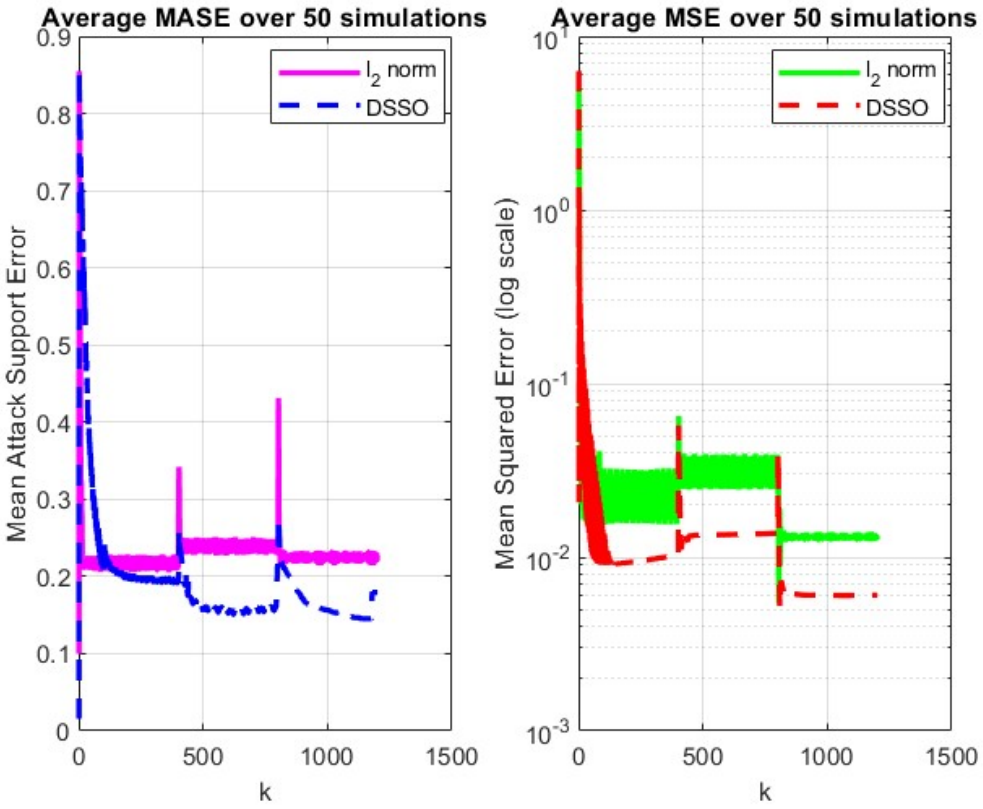

Figure 6.4: On the left, the comparison of the MASE between the D-SSO and the new theoretical approach is shown, while on the right, the corresponding comparison of the MSE on the estimated state is displayed, in the case where the attack support varies over time.

6.2.6 Sinusoidal attack

In this scenario, the attack vector $a(k)$ is characterized by a constant and sparse support ($h = 3$), whereas its nonzero entries evolve according to a sinusoidal law.

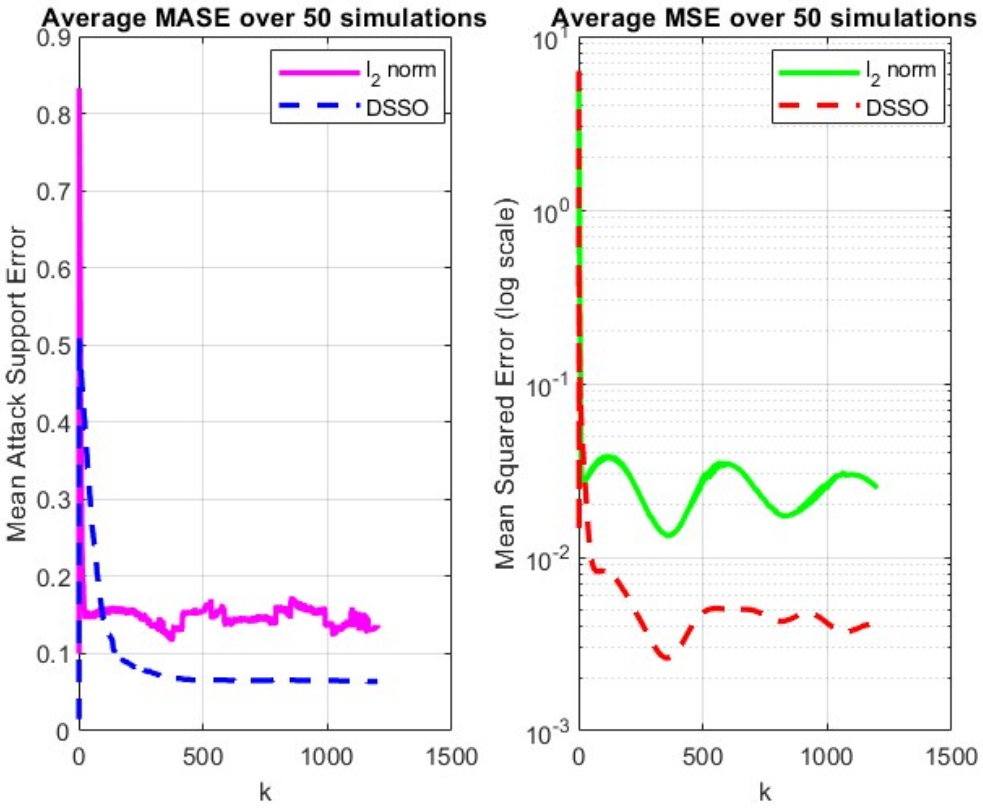

Figure 6.5: On the left, the comparison of the MASE between the D-SSO and the new theoretical approach is shown, while on the right, the corresponding comparison of the MSE on the estimated state is displayed, in the case where the attack follows a sinusoidal pattern.

6.2.7 Discussion

The proposed algorithm has shown promising results in the context of secure state estimation under sparse sensor attacks. The procedure for tuning the hyper-parameter λ and the gain matrix L_2 proved effective: through a systematic search it was possible to identify an optimal pair that balances the reconstruction of the state with the detection of the attacked sensors. The adopted cost function, combining the Mean Squared Error (MSE) and the Mean Attack Support Error (MASE), allowed us to evaluate the performance along two complementary directions: accuracy in tracking the system dynamics and reliability in identifying the malicious perturbations.

From the numerical simulations, it emerged that the observer is capable of accurately reconstructing the state trajectories, even in the presence of non-constant and sparse attacks on the sensors. In particular, the estimation error on the states converges to values close to zero, while the support of the attack vector is consistently

identified. The construction of the system matrix A, ensuring marginal stability, highlighted the robustness of the algorithm also in non-trivial dynamical scenarios, where the system is neither asymptotically stable nor unstable.

6.3 Localization problem

6.3.1 Introduction to Localization, Detection and Tracking

Localization refers to the process of estimating the position of a target using measurements collected from sensors or the environment.

However, our goal is not limited to solving localization problems. We aim to broaden our perspective by also addressing the following types of problems:

- **Detection** – determining whether a target is present in the environment;
- **Tracking** – estimating the trajectory of a moving target over time.

Localization, detection, and tracking tasks are fundamental in a wide range of Cyber-Physical Systems (CPS) because they require a complex system involving both physical and digital components. The localization problem is a current case study, as it is essential to enable cooperation between different agents, particularly in multi-agent systems and collaborative robotics. Even in the context of localization, security must be given great importance, as a well-designed external attack can lead to larger problems related to human safety.

Received signal strength

Localization fingerprinting refers to techniques that match the fingerprint of some characteristic of a signal that is location-dependent. In our development, we will consider the so-called RSS-fingerprint. RSS stands for Received signal strength, and it is defined as:

$$\mathrm{RSS}(d) = P_t - \overline{P}_l - 10\alpha \log_{10}\left(\frac{d}{d_0}\right) + \xi \tag{6.2}$$

where:

- P_t is the transmission power, expressed in dBm;
- d is the distance between the transmitter and the receiver;
- $\overline{P}_l$ denotes the average path loss in the environment;

- d_0 is a reference distance (typically $d_0 = 1$ meter);
- α is the attenuation coefficient that depends on the environment (e.g., walls, obstacles);
- ξ represents a random noise component, typically modeled as a Gaussian variable.

Localization via RSS Fingerprinting

Localization fingerprinting refers to a class of techniques that estimate the position of a target by matching signal characteristics,commonly the Received Signal Strength (RSS), that vary with location [28]. The fundamental idea is that the signal's "fingerprint" at each position is unique enough to serve as an identifier for that location.

General Fingerprinting Approach

The fingerprinting process is typically divided into two main phases:

1. **Training (offline) phase**: A set of fingerprints is collected throughout the environment by placing the target in known positions. At each of these positions, the system records signal characteristics such as RSS from multiple sensors.
2. **Runtime (online) phase**: The current RSS measurements collected by the sensors are compared with the pre-recorded fingerprints. The target's location is estimated by identifying the most similar fingerprint.

RSS-based Fingerprinting

In this thesis, we focus on RSS fingerprinting within a wireless sensor network (WSN). The following procedure shows the essential steps to properly perform the training phase and the runtime phase.

Training phase:

1. Divide the environment into a grid of cells.
2. For each cell, place the target at the center and let the sensors record the RSS values.

Runtime phase:

1. When a new target appears, each sensor measures the RSS.
2. Localization is then performed by determining in which cell the new RSS vector best matches the stored fingerprints.

Each grid cell represents a possible location where the target can be found, and the system predicts the most likely one based on current measurements by solving an optimization problem.

RSS-Fingerprinting: Training Phase

During the RSS-fingerprint training phase, a dictionary $D$is created that collects measurements from all sensors. This dictionary is essential for solving the optimization problem in the runtime phase. The structure of the dictionary is presented below.

The main steps are as follows:

- The environment is divided into a grid of n cells.
- The target device is placed sequentially in each cell (typically at the center).
- In each position, the target broadcasts a signal that is received by the sensors deployed in the area.
- Each sensor i measures the RSS and stores the result, building a signature that characterizes the signal strength for each cell.

Let us denote by $D_{i,j}$ the RSS measurement acquired by sensor i when the target is located in cell j. Then, each sensor i constructs a local dictionary as a column vector:

$$D_i = \begin{bmatrix} D_{i,1} \\ D_{i,2} \\ \vdots \\ D_{i,n} \end{bmatrix} \in \mathbb{R}^n$$

By collecting the measurements from all q sensors in the Wireless Sensor Network (WSN), we obtain the global dictionary:

$$D = \begin{bmatrix} D_1^\top \\ D_2^\top \\ \vdots \\ D_q^\top \end{bmatrix} \in \mathbb{R}^{q \times n}$$

This dictionary D will then be used during the online phase to estimate the position of the target by comparing new measurements with these reference signatures.

6.3.2 Mathematical Model and Localization Algorithms

After the training phase, in which the dictionary $D \in \mathbb{R}^{q \times n}$ is constructed from signal strength measurements in each cell of the environment, the runtime phase focuses on using new measurements to estimate the current location of the target.

Problem Setting

We consider a centralized scenario, where:

- Each of the q sensors acquires a measurement $y_i \in \mathbb{R}$ of the received signal strength (RSS) when the target is active.
- All the sensors send their measurements to a centralized fusion center (FC).
- The FC stores the dictionary D and processes the measurement vector $y = [y_1, \ldots, y_q]^\top \in \mathbb{R}^q$.

Ideal vs. Realistic Scenarios

In an idealized setting, we assume:

- Each fingerprint $D_{i,j}$ is unique across all cells j.
- The RSS measurements taken during the training and runtime phases are perfectly consistent.

Under these assumptions, each sensor could in principle localize the target independently by solving:

$$\text{Find } j \text{ such that } D_{i,j} = y_i.$$

However, in practice, this scenario is unrealistic due to the presence of noise and the similarity of measurements across neighboring cells. As a result:

- Runtime measurements deviate from training values.
- Different cells may yield similar RSS vectors.

The k-Nearest Neighbors (k-NN) Approach

Given the evident differences between the ideal case and the real one, it is necessary to find optimization algorithms suitable for minimizing the errors that arise in the real context. A common strategy to estimate the location is the k-nearest neighbors (k-NN) method. This algorithm works as follows:

- Assume that only one target is present.
- Given the measurement vector $y \in \mathbb{R}^q$, the fusion center searches for the column D_j of the dictionary D that is closest to y in the Euclidean norm:

$$\hat{j} = \arg \min_{j=1,\dots,n} \|D_j - y\|_2^2$$

  where $D_j = [D_{1,j}, \dots, D_{q,j}]^\top$ is the j-th column of D.

The algorithm just presented provides a simple approach for single-target localization, but becomes inefficient when multiple targets are present. This justifies its natural extension to multi-target scenarios, as described below.

Multi-Target Localization and Computational Complexity

The k-NN algorithm can be extended to localize multiple targets by exploiting the approximate additivity of RSS. For instance, if there are targets located in cells j_1 and j_2, the measurements received by sensor i can be approximated as:

$$y_i \approx D_{i,j_1} + D_{i,j_2} + \text{noise}$$

Thus, the optimization becomes:

$$(\hat{j}_1, \hat{j}_2) = \arg \min_{j_1,j_2=1,\dots,n} \|D_{j_1} + D_{j_2} - y\|_2^2$$

However, the number of possible combinations grows combinatorially with n and k:

$$\binom{n}{k} \quad \text{possible configurations}$$

This renders the problem computationally intractable for large values of n and k (NP-hard).

Localization as a Binary Linear Regression Problem

An alternative approach is to cast the localization task as a constrained optimization problem:

$$\begin{aligned} &\text{Find } x \in \{0,1\}^n \\ &\text{such that } Dx = y \\ &\sum_{j=1}^{n} x_j = k \end{aligned}$$

Here, the binary vector x encodes the positions of the k targets (with 1 indicating presence in a given cell), and $Dx = y$ models the additive RSS contribution of each active target. However, since this is a mixed-integer problem, it is again combinatorially hard to solve and belongs to the class of NP-hard problems.

Problem Relaxation and Regularized Optimization Methods

In the previous formulation, the localization problem was modeled as a binary linear regression problem with combinatorial constraints:

$$\begin{aligned} Dx &= y \\ x &\in \{0,1\}^n \\ \sum_{j=1}^{n} x_j &= k \end{aligned}$$

This formulation, although expressive, is computationally intractable in realistic scenarios due to its *NP-hard* nature.

Relaxation: Least-Squares

A first relaxation involves neglecting the binary and cardinality constraints, allowing x to take real values:

$$x \in \mathbb{R}^n$$

In the presence of noise, the relation $Dx = y$ is reformulated as a least-squares problem:

$$\hat{x} = \arg\min_{x \in \mathbb{R}^n} \|Dx - y\|_2^2$$

This formulation is effective when the number of sensors q is greater than or equal to the number of cells n ($q \geq n$), making the system overdetermined.

Relaxation: Lasso and Elastic Net Approach

However, in cases where $q < n$, i.e., when the number of sensors is smaller than the number of cells (which is typical in practical scenarios), the system becomes underdetermined. Moreover, in many situations, the number of targets is significantly smaller than the total number of cells, that is, $k \ll n$.

In such cases, it is useful to exploit the *sparse structure* of the vector x by using Lasso regression:

$$\hat{x} = \arg\min_{x \in \mathbb{R}^n} \|Dx - y\|_2^2 + \lambda \|x\|_1$$

The addition of the ℓ_1 regularization term promotes sparse solutions, in which only a small number of components of x are significantly different from zero. This approach is particularly suitable for multi-target localization problems.

As discussed in the previous chapters, an alternative regularization scheme is given by the *Elastic Net*, which combines the ℓ_1 and ℓ_2. IN the context of localization, its formulation is

$$\hat{x} = \arg\min_{x \in \mathbb{R}^n} \ \|Dx - y\|_2^2 + \lambda \|x\|_1 + \frac{\mu}{2}\|x\|_2^2,$$

where $\lambda, \mu \geq 0$ are regularization parameters.

Robust Localization under Sparse Sensor Attacks

In adversarial scenarios, some sensors may be compromised by attacks, leading to corrupted measurements. In this context, it is assumed that such attacks are *sparse*.

To model this situation, a vector $a \in \mathbb{R}^q$ is introduced to represent the effect of the attacks on the received measurements:

$$\hat{x}, \hat{a} = \arg\min_{x \in \mathbb{R}^n, a \in \mathbb{R}^q} \|Dx + a - y\|_2^2 + \lambda_x \|x\|_1 + \lambda_a \|a\|_1 + \frac{\mu_x}{2}\|x\|_2^2 + \frac{\mu_a}{2}\|a\|_2^2$$

This formulation is an extended Elastic Net problem, where both the state and the attack vector are regularized by introducing a combination of the ℓ_1 and ℓ_2 norms.

The algorithm to solve the previous optimization problem is illustrated in the next section.

Algorithm Setup

The proposed algorithm aims to estimate the position of a target and to detect possible sensor attacks in a wireless localization scenario, starting from noisy received

signal strength (RSS) measurements. The data used in the algorithm are loaded from a file, which contains the following variables:

- $D \in \mathbb{R}^{q \times n}$: the dictionary, whose columns represent the expected RSS fingerprints associated with each of the n candidate target positions, and whose rows correspond to the q sensors in the network.
- $y \in \mathbb{R}^{q \times 1}$: the vector of measured RSS values collected by the sensors at a given time.

The dictionary and the measurement vector are combined as:

$$G = [D \; I_q],$$

where $I_q \in \mathbb{R}^{q \times q}$ is the identity matrix. The augmented dictionary G is constructed in such a way that the first n columns correspond to the possible cells in which the target can be found taken from dictionary D, while the remaining q columns serve to model the possible additive signals of the attacks. Thus, the estimation problem can be written as

$$y = Gz = Dx + a,$$

where $x \in \mathbb{R}^n$ represents the (sparse) position vector of the target and $a \in \mathbb{R}^q$ represents the (sparse) attack vector.

The concatenated variable is defined as

$$\hat{z} = \begin{pmatrix} \hat{x} \\ \hat{a} \end{pmatrix} \in \mathbb{R}^{n+q},$$

which is set to zero at first, presuming that there isn't a target or an attack.

$N = 1000$ iterations are performed by the iterative algorithm presented below. The main parameters, which were selected based on the new theoretical formulation, are:

- L_2;
- λ_a, λ_x: soft-thresholding parameters that encourage sparsity on the attack estimate a and the target estimate x, respectively;
- q_x, q_a that are derived from Elastic Net regularization theory.

Whereas the nonzero entries in $\hat{a}$ identify the sensors that are under attack, the nonzero entries in $\hat{x}$ indicate the estimated target location.

Algorithm 3: Elastic net for localization problem

Input: N, $y \in \mathbb{R}^{q\times 1}$, $G \in \mathbb{R}^{q\times(n+q)}$, $\lambda_x > 0$, $\lambda_a > 0$, $q_x \in (0,1)$, $q_a \in (0,1)$, L_2, $\varepsilon > 0$

Output: $\begin{pmatrix}\hat{x}\\ \hat{a}\end{pmatrix}$ = final estimate of $\begin{pmatrix}x\\ a\end{pmatrix}$

Initialization: $\hat{z}(0) = \begin{pmatrix}0\\ 0\end{pmatrix}$

for $k = 1$ **to** N **do**

$$\gamma(k) = \hat{z}(k) - \varepsilon L_2\, G^\top \big(G\hat{z}(k) - y(k)\big)$$

Separate variables:

$$\gamma_x(k) = \gamma_{1:n}(k), \qquad \gamma_a(k) = \gamma_{n+1:\text{end}}(k)$$

Apply elastic-net soft-thresholding:

$$\hat{x}(k+1) = q_x \cdot S_{\lambda_x}\left(\gamma_x(k)\right)$$

$$\hat{a}(k+1) = q_a \cdot S_{\lambda_a}\left(\gamma_a(k)\right)$$

Recombine estimates:

$$\hat{z}(k+1) = \begin{pmatrix}\hat{x}(k+1)\\ \hat{a}(k+1)\end{pmatrix}$$

end

Results

Below there is a visual representation showing the functioning of the grid-based localization approach. The figure illustrates how the environment is divided into cells and how the algorithm estimates the target position based on sensor measurements. In particular, the system has 100 cells and 20 sensors, five of which are under attack.

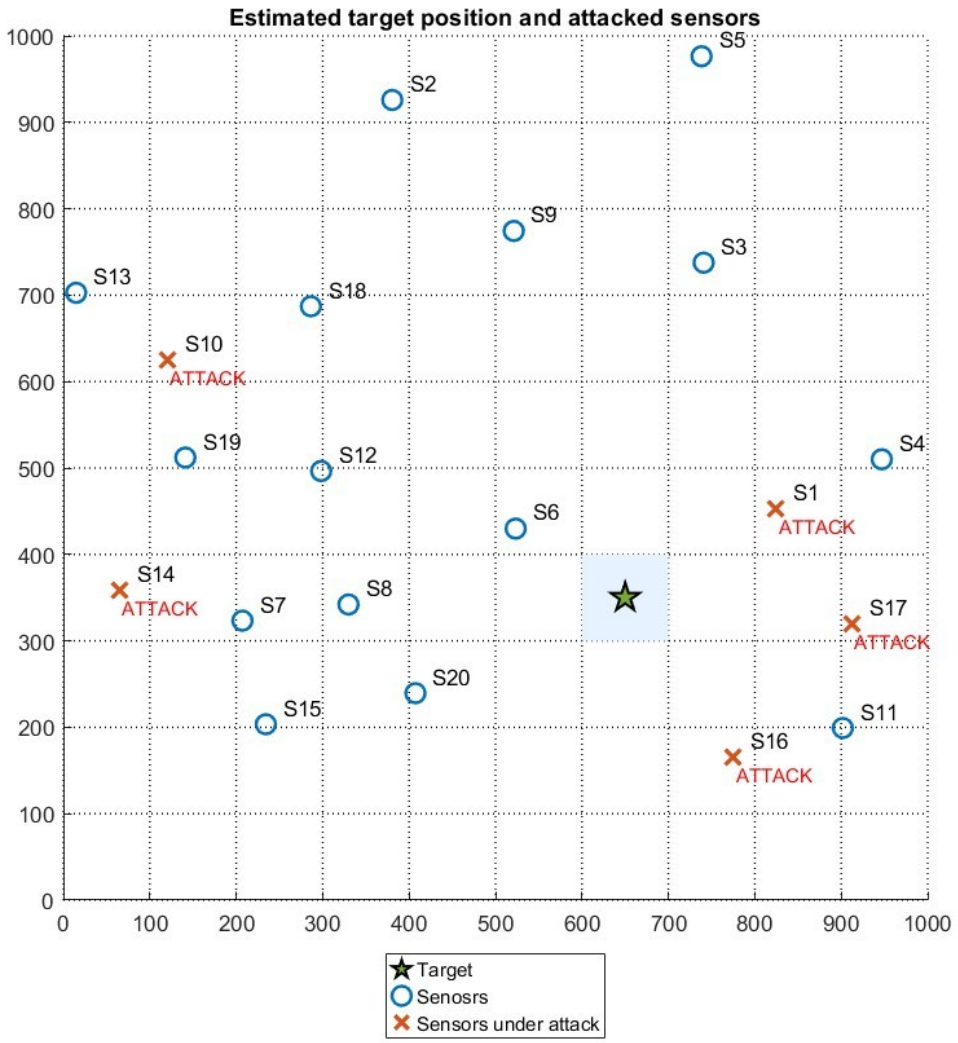

Figure 6.6: Target localization and sensor attack estimation results displayed on a grid.

As we can see from the figure 6.6, the observer was able to identify both the cell containing the target and which of the 20 sensors were attacked. In the figure 6.6, the cell containing the target is marked with a green star, the sensors under attack are marked with orange "X"s, while the remaining sensors were not attacked.

6.4 Real world case

It is very interesting to examine a real world case study inspired by an example presented in [29] where the system under study is an unmanned ground vehicle

(UGV), which is a ground vehicle that operates without an onboard human presence. This scenario is particularly relevant since the state transition matrix A has an eigenvalue equal to 1, a condition that, as previously discussed, directly affects the system's observability properties. The UGV is equipped with three sensors. The system dynamics and output equations are characterized by the following matrices:

$$A = \begin{bmatrix} 1 & 0.0099 \\ 0 & 0.9876 \end{bmatrix} \quad ; \quad C = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 1 \end{bmatrix}$$

For the simulation study, the malicious attack is acting on the second sensor. This scenario allows us to evaluate the observer's behavior under marginal stability conditions. The simulation results evaluating the errors in state and attack estimation are shown below.

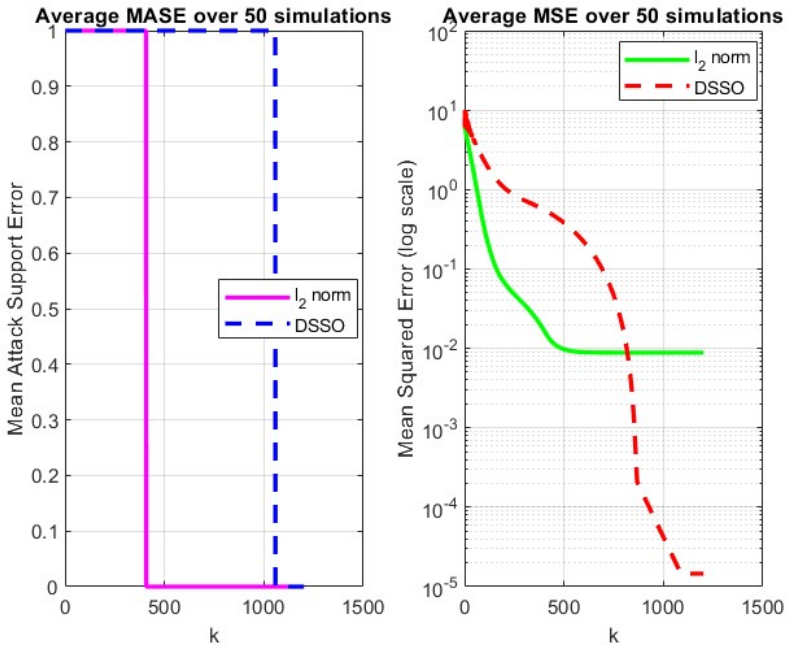

Figure 6.7: On the left, the comparison of the MASE between the D-SSO and the new theoretical approach is shown, while on the right, the corresponding comparison of the MSE on the estimated state is displayed, in the UGV scenario.

Interpretation of Results

The simulation shown in figure 6.7 shows that the observer is able to reconstruct the system state fairly accurately and, furthermore, is able to determine which of the three sensors is under attack. Specifically, the error in estimating the attack support is zero, confirming the observer's correct operation in this regard, while the

state estimation error converges to values close to zero. The scenario studied proved relevant for theoretical speed, given that matrix A has an eigenvalue of 1, placing us in a marginally stable case.

The results obtained encourage further investigation for real-world implementation.

Chapter 7

Conclusion and future prospects

This thesis investigated the problem of secure state estimation, starting from the theoretical foundations necessary to develop and enrich existing models in the literature, which present limitations when the system is subjected to malicious attacks. The main contribution of this thesis was to develop a new observer model, which is capable of estimating both the state and the attack. Under specified assumptions, the stability of the observer in the presence of malicious attacks is demonstrated via a Lyapunov-based approach. The proposed model did not require restrictive assumptions during its theoretical formulation, and this aspect does not limit its application to more real-world scenarios in contexts where the type of attack cannot be classified. The new observer architecture showed promising results in both synthetic and realistic scenarios, managing to estimate both the system state and the attack support. Overall, the results obtained in this thesis constitute a solid theoretical foundation in the field of SSE. Secondly, this work paves the way for potential future developments. Possible directions include real-world applications in the field of autonomous vehicles, industrial automation, medical equipment, and localization problems. Furthermore, a possible future development is to reformulate and extend the new theoretical approach developed in a decentralized context.

Bibliography

[1] E. Chang, M. Machizaud, and M. Dunn, "Advances in internet of things and cyber-physical systems and its adoption to smart ship," in *Int Conf Wirel Commun Network, Balt USA*, 2015.

[2] F. Liberati, E. Garone, and A. Di Giorgio, "Review of cyber-physical attacks in smart grids: A system-theoretic perspective," *Electronics*, vol. 10, no. 10, 2021.

[3] O. Vermesan and B. Roy, "Internet of things and cyber-physical systems sintef," 2016.

[4] N. Parikh and S. Boyd, "Proximal algorithms," *Foundations and Trends in Optimization*, vol. 1, no. 3, pp. 127–239, 2014.

[5] V. Cerone, S. M. Fosson, D. Regruto, and F. Ripa, "Lasso-based state estimation for cyber-physical systems under sensor attacks," in *IFAC-PapersOnLine*, vol. 58, pp. 163–168, Elsevier, 2024.

[6] N. Bof, R. Carli, and L. Schenato, "Lyapunov theory for discrete time systems," 2018.

[7] E. D. Sontag, "Remarks on stabilization and input-to-state stability," in *Proceedings of the IEEE Conference on Decision and Control*, vol. 2, pp. 1376–1378, IEEE, 1989.

[8] E. D. Sontag, "Comments on integral variants of iss," *Systems & Control Letters*, vol. 34, no. 1-2, pp. 93–100, 1998.

[9] Z.-P. Jiang and Y. Wang, "Input-to-state stability for discrete-time nonlinear systems," *Automatica*, vol. 36, no. 2, pp. 241–248, 2000.

[10] A. Isidori, *Nonlinear Control Systems II*. Springer, 2013.

[11] N. I. of Standards and T. (NIST), "Framework for cyber-physical systems: Volume 1, overview," 2017. Accessed: 2025-10-10.

[12] A. A. Alsulami, Q. A. Al-Haija, B. Alturki, *et al.*, "Security strategy for autonomous vehicle cyber-physical systems using transfer learning," *Journal of Cloud Computing*, vol. 12, no. 181, 2023.

[13] P. Leitão, A. Colombo, and S. Karnouskos, "Industrial automation based on cyber-physical systems technologies: Prototype implementations and challenges," *Computers in Industry*, vol. 81, 09 2015.

[14] M. Javaid, A. Haleem, R. P. Singh, and R. Suman, "An integrated outlook of cyber–physical systems for industry 4.0: Topical practices, architecture, and applications," *Green Technologies and Sustainability*, vol. 1, no. 1, p. 100001, 2023.

[15] S. Karnouskos, "Cyber-physical systems in the smartgrid," *IEEE International Conference on Industrial Informatics (INDIN)*, pp. 20 – 23, 08 2011.

[16] H. M. Khater, F. Sallabi, M. A. Serhani, E. Barka, K. Shuaib, A. Tariq, and M. Khayat, "Empowering healthcare with cyber-physical system—a systematic literature review," *IEEE Access*, vol. 12, pp. 83952–83993, 2024.

[17] S. Haque, S. Aziz, and M. Rahman, "Review of cyber-physical system in healthcare," *International Journal of Distributed Sensor Networks*, vol. 2014, p. 20, 04 2014.

[18] N. Dey, A. S. Ashour, F. Shi, *et al.*, "Medical cyber-physical systems: A survey," *Journal of Medical Systems*, vol. 42, no. 74, 2018.

[19] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: A survey," *Computer Networks*, vol. 38, no. 4, pp. 393–422, 2002.

[20] T. Rault, A. Bouabdallah, and Y. Challal, "Energy efficiency in wireless sensor networks: A top-down survey," *Computer Networks*, vol. 67, pp. 104–122, 2014.

[21] R. E. Kalman, "Contributions to the theory of optimal control," *Boletin de la Sociedad Matematica Mexicana*, vol. 5, pp. 102–119, 1960.

[22] 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, 2014.

[23] D. L. Donoho, "Compressed sensing," *IEEE Transactions on Information Theory*, vol. 52, no. 4, pp. 1289–1306, 2006.

[24] E. J. Candès and M. B. Wakin, "An introduction to compressive sampling," *IEEE Signal Processing Magazine*, vol. 25, no. 2, pp. 21–30, 2008.

[25] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 58, no. 1, pp. 267–288, 1996.

[26] G. C. Calafiore and L. El Ghaoui, *Optimization Models*. Cambridge University Press, 2014.

[27] E. Davison and S. Wang, "New results on the controllability and observability of general composite systems," *IEEE Transactions on Automatic Control*, vol. 20, no. 1, pp. 123–128, 1975.

[28] A. Bay, D. Carrera, S. M. Fosson, P. Fragneto, M. Grella, C. Ravazzi, and E. Magli, "Block-sparsity-based localization in wireless sensor networks," *EURASIP Journal on Wireless Communications and Networking*, vol. 2015, no. 1, p. 182, 2015.

[29] Y. Shoukry, P. Nuzzo, A. Puggelli, A. L. Sangiovanni-Vincentelli, S. A. Seshia, and P. Tabuada, "Secure state estimation for cyber-physical systems under sensor attacks: A satisfiability modulo theory approach," *IEEE Transactions on Automatic Control*, vol. 62, no. 10, pp. 4917–4932, 2017.