UNIVERSITY OF CALIFORNIA

Los Angeles

Security and Privacy in Dynamical Systems

A dissertation submitted in partial satisfaction
of the requirements for the degree
Doctor of Philosophy in Electrical and Computer Engineering

by

Mehrdad Showkatbakhsh

2019

ABSTRACT OF THE DISSERTATION

Security and Privacy in Dynamical Systems

by

Mehrdad Showkatbakhsh
Doctor of Philosophy in Electrical and Computer Engineering
University of California, Los Angeles, 2019
Professor Suhas N. Diggavi, Chair

Dynamical systems have found applications in many domains including control and optimization, which have risen to great prominence. Physical processes in nature can be classified as dynamical systems. Control theory tries to understand these systems, in order to design certain mechanisms and to obtain desired behaviors. On the other hand, optimization algorithms are inherently recursive and therefore can be modeled as dynamical systems. Such systems give rise to an abundance of applications, therefore, addressing their unreliability is important. In this dissertation, we focus on challenges arising from vulnerabilities of such systems against (active) attacks on physical components and (passive) attacks to infer about sensitive information. We take steps forward toward understanding these challenges and toward making progress in building robust systems.

Many control systems have a cyber-physical nature, meaning there is a tight interaction between cyber (computation and communication) and physical (sensing and actuation) components of the system. Cyber-Physical Systems (CPS) have enabled numerous applications in which decisions need to be taken depending on the environment and sensory information. However, addressing the unreliability that may stem from communication, software security, and physical vulnerabilities still remains a fundamental challenge. In the first part of this dissertation, we focus on the physical vulnerabilities of *sensing* and *actuation* modules, in which an adversary manipulates these components. Particularly, two problems of "state estimation" and "system identification" are analyzed in an adversarial environment. In order

to make the system robust against such attacks, we propose several schemes to mitigate the adversarial agents impact.

In recent years, personal data from health care, finance, and etc are becoming available that enables learning high complexity models for applications ranging from medical diagnosis and financial portfolio strategies among others. The common paradigm to learn such models is to optimize a cost function involving the model parameters and the data. Acquiring data from individuals and publishing models based on them compromises the privacy of users against a passive adversary observing the training procedure. Addressing this vulnerability is crucial in this increasingly common scenario where we build models based on sensitive data. For instance, the privacy concern is a major roadblock in large scale use of sensitive personal data in health care. In the second part of this dissertation, we investigate two problems in this area: "private linear-regression" and "private distributed optimization". These methods develop and analyze private learning mechanisms which guarantee utility while ensuring a given privacy level.

The dissertation of Mehrdad Showkatbakhsh is approved.

Lieven Vandenberghe

Arash Ali Amini

Paulo Tabuada

Suhas N. Diggavi, Committee Chair

University of California, Los Angeles

2019

To my family,

To my parents, Rahman and Nooshin,

To my brother, Milad ...

Table of Contents

List of Figures

List of Tables

Acknowledgments

The past fives years, over which I have been working on my graduate studies have had its fair share of ups and downs. I would like to express my sincere gratitude to my Advisor, Professor Suhas Diggavi for his support, patience, and guidance throughout these years, without which I would not have been able to make it to the end. This thesis would have been impossible without your help and invaluable feedback.

I also would like to thank Professor Arash Ali Amini and Lieven Vandenberghe for serving on my doctoral committee and whose feedback was imperative to building this thesis. I am also grateful to Professor Paulo Tabuada, who in addition to serving on my doctoral committee, lead me through my first steps in my doctoral research.

I would like to thank my lab mates and colleagues; Nikhil, Shaunak, Jad, Can, Joyson, Yasser, Joris, Sina, Yair, Ayan, Debraj, Deepesh, and Navjot. Specially, I would like to express my gratitude to Can Karakus for being both a great friend during these challenging yet rewarding years and for providing invaluable feedback in the second part of this thesis.

Last but not least, I would like to thank my family: my parents, Rahman and Nooshin and my brother, Milad for immeasurable love and support throughout this challenging period of my life, for giving me comfort regardless of their own difficulties. This thesis is dedicated to you.

Vita

2013-2019	Research Assistant, Electrical and Computer Engineering University of California, Los Angeles
2017-2018	Teaching Assistant, Electrical and Computer Engineering University of California, Los Angeles
Summer 2018	Data Scientist Morgan Stanley, New York, NY
Summer 2017	Strategist Goldman Sachs, New York, NY
Summer 2016	Quantitative Analyst Jefferies, New York, NY
June 2015	M.S., Electrical Engineering University of California, Los Angeles
June 2013	B.S., Electrical Engineering Sharif University of Technology, Iran

Publications

M. Showkatbakhsh, C. Karakus, S. Diggavi, "Differentially private consensus-based distributed optimization", *arXiv.org*, 2019.

M. Showkatbakhsh, Y. Shoukry, S. Diggavi, P. Tabuada "Securing state estimation under sensor and actuator attacks: theory and design", *arXiv.org*, 2019.

M. Showkatbakhsh, C. Karakus, S. Diggavi, "Privacy-utility trade-off of linear regression under random projections and additive noise", *IEEE International Symposium on Information Theory (ISIT)*, 2018.

M. Showkatbakhsh, Y. Shoukry, R. H. Chen, S. Diggavi, P. Tabuada, "An SMT-based approach to secure state estimation under sensor and actuator attacks", *IEEE Conference on Decision and Control (CDC)*, 2017.

M. Showkatbakhsh, P. Tabuada, S. Diggavi, "Secure system identification", *IEEE Annual Allerton Conference on Communication, Control, and Computing*, 2016.

M. Showkatbakhsh, P. Tabuada, S. Diggavi, "System identification in the presence of adversarial outputs", *IEEE Conference on Decision and Control (CDC)*, 2016.

CHAPTER 1

Introduction

1.1 Dynamical Systems

Dynamical systems have found applications in many domains including control and optimization, which have risen to great prominence. Physical processes are dynamical systems in essence and control theory tries to understand them. Optimization algorithms are inherently recursive and therefore can be classified as dynamical systems. Over the past few decades, these systems have found numerous and growing applications in a variety of fields. Consequently, there has been an increasing number of incidents targeting the integrity and security of such systems. In this dissertation, we take steps forward toward addressing the vulnerabilities of these systems against adversaries.

Many control systems have a cyber-physical nature, meaning there is a tight interaction between cyber (computation and communication) and physical (sensing and actuation) components. In the first part of this dissertation, we focus on the vulnerability of Cyber-Physical Systems (CPS) against an active adversary disrupting the control loop. In the second part, we turn to optimization systems. We focus on scenarios in which we optimize a cost function to build an inference model based on sensitive data. We address privacy concerns in these systems and build robust mechanism against passive adversaries overhearing the training procedure.

1.1.1 Cyber Physical Systems

CPS are characterized by a tight interconnection of its cyber and physical components, where at its core, a dynamical system lies along with a network of sensing and control modules. These systems find applications in many domains ranging from electricity grids and power plants to modern cars. CPS are not only prone to actuator and sensor failures, but also to adversarial attacks on the control and sensing modules. Therefore, the security of CPS is no longer restricted to the cyber domain. Moreover, publicized incidents [Gre15, Kel16] motivated the recent interest in the security of CPS, specially from the control community (see for example, [CAS08, SPH+10, ASH13, MKB+12] and references therein). In this dissertation, we study among others, the physical vulnerabilities, where a malicious agent can corrupt sensing and actuation modules and thereby causing damages. Particularly we touch on two problems in this domain: "secure state estimation" and "secure system identification".

Erroneous state estimation could result in a serious disruption of the performance of the CPS and may cause damages to the underlying infrastructure. We study the state estimation when some of the sensing and actuation modules are manipulated by an adversarial agent under the topic of secure state estimation. We formalize the redundancy we need despite attacks on both sensors and actuators by introducing the notion of sparse-strong observability. We further propose an estimator to harness the complexity of this intrinsically combinatorial problem, by leveraging satisfiability modulo theory solving paradigm.

Identifying the underlying model of systems is of great importance in control theory and is crucial for designing controllers. The second problem in the CPS domain that we study is system identification in an environment where an adversarial agent alters some of the sensor measurements. We show that we can still construct a model that is useful for stabilization, and closely related to the correct model.

1.1.2 Optimization

In recent years, personal data from health care, finance, and etc are becoming more and more available, which by itself enables learning high complexity models for applications ranging from medical diagnosis and financial portfolio strategies. The common paradigm to learn such models is to optimize a cost function involving the model parameters and the data. Acquiring data from individuals and publishing models based on them compromises the privacy of users against a passive adversary observing the training procedure. Addressing this vulnerability is crucial in this increasingly common scenario where we build models based on sensitive data. For instance, the privacy concern is a major roadblock in large scale use of sensitive personal data in health care.

Differential privacy is perhaps the most well-known notion for privacy [DR$^+$14], and has been applied to a variety of domains (see, for example, [SC13] and [DR$^+$14] and references therein). It assumes a strong adversary which has access to all data samples except one, thereby ensuring robustness of the privacy guarantee to adversaries with side-information about the database. By taking differential privacy as the rigorous and quantifiable privacy metric, we take steps toward understanding the fundamental trade-off between the privacy and utility for two problems: "private linear-regression" and "private distributed optimization". These methods develop and analyze private learning mechanisms which ensure utility while giving privacy guarantees.

One possible way of fulfilling the learning task while preserving user privacy is to train the model on a transformed, noisy version of the data, which does not reveal the data itself directly to the training procedure. We analyze the privacy-utility trade-off of two such schemes for the problem of linear regression: additive noise, and random projections.

In the second problem, we turn to the distributed setup which consists of a set of computational nodes, arranged in a graph, each having a local objective that depends on their sensitive data. Our proposed method is a modified version of the distributed gradient descent algorithm [NO09, YLY16], in which nodes perturb their messages. We show that by

properly injecting noise at different steps, the algorithm converges to a neighborhood around the optimal point while ensuring differential privacy.

1.2 Thesis Outline

This dissertation is composed of four research topics. The summary of the results is as follows:

Chapter 2 addresses the problem of state estimation when some of sensors and actuators are under attack. Our attack model is quite general and we impose no constraint on the magnitude, statistical properties, or temporal characteristics of the signal injected. We introduce the notion of sparse strong observability thereby characterizing systems for which state estimation is possible despite attacks. In the second half of this work, we propose an estimator to harness the complexity of this intrinsically combinatorial problem, by leveraging satisfiability modulo theory solving paradigm.

In Chapter 3, we study system identification of linear time-invariant systems in the presence of an adversarial agent attacking sensors. The attacker is omniscient and we impose no restrictions (statistical or otherwise) on how the adversary alters the sensor measurements. Given a bound on the number of attacked sensors, and under a certain observability condition, we show that we can still construct a model that is useful for stabilization. Furthermore, we show that this model is closely related to the original system through similarity modulo outputs relation.

In Chapter 4, we turn to the privacy problem. One possible way of fulfilling the machine learning task while preserving user privacy is to train the model on a transformed, noisy version of the data, which does not reveal the data itself directly to the training procedure. In this chapter, we analyze the privacy-utility trade-off of two such schemes for the problem of linear regression: additive noise, and random projections. We observe that the random projections scheme yields a substantially improved utility for a given privacy level, comparing to the additive noise scheme.

In Chapter 5, we focus on a distributed setup in which sensitive data is stored across different nodes. We study the consensus-based distributed optimization algorithm which consists of a set of computational nodes, arranged in a graph, each having a local objective that depends on their local data. In each step, nodes take a new gradient step and a linear combination of their neighbors' messages. Since the algorithm requires exchanging messages that depend on local data, private information gets leaked at every step. Our proposed method is a modified version of the distributed gradient descent algorithm, in which nodes perturb their messages by adding noise. We show that by properly injecting noise at different steps, the algorithm converges to a neighborhood around the optimal point.

Chapter 6 concludes the dissertation with conclusion and future direction.

We point out that most of the material in this thesis has been published, or submitted for publication as of this date. The preliminary result of Chapter 2 was partly published in 56th IEEE conference on Decision and Control [SSC+17], and partly submitted for publication in [SSDT18], those in Chapter 3 were published in 55th IEEE conference on Decision and Control [STD16b] and 54th Annual Allerton Conference on Communication, Control, and Computing [STD16a]. Chapter 4 was published in 2018 IEEE International Symposium on Information Theory [SKD18]. Contents of Chapter 5 is submitted for publication [SKD19].

CHAPTER 2

Secure State Estimation

2.1 Introduction

Cyber-Physical Systems (CPS) are characterized by the tight interconnection of cyber and physical components. CPS are not only prone to actuator and sensor failures but also to adversarial attacks on the control and sensing modules. Security of CPS is no longer restricted to the cyber domain, and recent incidents such as the StuxNet malware [Lan11] and the security flaws reported on modern cars [Gre15, Kel16] motivated the recent interest in security of CPS, (see for example, [CAS08, SPH+10, ASH13, MKB+12] and references therein). During the last decade, a number of security problems have been tackled by the control community, *e.g.,* denial-of-service [ZM14, DPT15, STDP16, GLB10], replay attacks [MS09], man-in-the-middle attacks [Smi15], false data injection [MGCS10], etc.

This chapter addresses the problem of state estimation when several sensors *and* actuators are under attack. We broadly refer to state estimation in the adversarial environment as secure state estimation. Our attack model is quite general and we impose no constraints on the magnitude, statistical properties, or temporal characteristics of the signals manipulated by the adversary.

Secure state estimation has gained the attention of the control community over the past decade [GUC+18]. In one line of work, the problem of state estimation and control under sensor attacks is investigated and the authors derived necessary and sufficient conditions under which estimation and stabilization are possible [FTD14a]. Shoukry et. al. [ST16b] further refined this condition and called it sparse observability. Chong et. al. [CWH15a]

found an equivalent condition for continuous-time systems and called it observability under attack. Nakahira et. a. [NM15] investigated a similar problem while considering the asymptotic correctness of state estimation. The authors relaxed the sparse observability condition to sparse detectability and showed it is a necessary and sufficient condition for asymptotic correctness. The noisy version of this problem has been investigated in the literature [BPG17, BGP17, MCS14, MS16, MSK$^+$17]. Mishra et. al. [MSK$^+$17] derived the optimal solution for Gaussian noise. In this work, we solve the more general problem of *actuator and sensor* attacks that includes, as a special case, sensors attacks.

Under the sparse attack model in which an adversary can only target a bounded number of actuators and sensors, state estimation is intrinsically a combinatorial problem. Shoukry et. al. [SNP$^+$17] proposed a novel secure state estimator using the Satisfiability Modulo Theory (SMT) paradigm, called Imhotep-SMT. The authors only considered attacks on sensors. In this work we address the more general problem of sensor *and* actuator attacks and build an SMT-based estimator that can correctly reconstruct the state under both types of attacks.

In another line of work, the problem of secure state estimation has been studied when the exact model of the system is not available [YFF16, PWB$^+$14a]. Tiwari et. al. [TDJ$^+$14] proposed an online learning method by building so-called safety envelopes as it receives attack-free data to detect abnormality in the data when the system is prone to attacks. In [STD16b, STD16a] the authors considered system identification under sensors attacks. In all of these works, the adversarial agent is restricted to only attacking sensors.

Pasqualetti et. al. [PDB13] investigated the problem of attack detection and identification. The authors related the undetectable and unidentifiable attacks to the zero-dynamics of the underlying system. The proposed attack identification mechanism consists of a number of fault-monitor filters that provide formal guarantees for the existence of the attack. The number of filters, however, grows exponentially with the number of attacked sensors/actuators, and therefore hinders scalability. In another work [ST16a], the authors investigated detectibility and identifiability of attacks in the presence of disturbances and the concept of

security index is generalized to dynamical systems. The proposed method is inherently combinatorial and does not scale well with the number of attacked sensors and actuators. In this work, by leveraging the SMT paradigm, we design a state estimator that scales well with the number of sensors and actuators.

Fault isolation and fault detection filters are classical control topics closely related to secure state estimation. The traditional fault tolerant filters can detect faults on actuators and sensors, however, they are not adequate for the purpose of security. Some of these filters assume a priori knowledge (statistical or temporal) of the fault signals [BKL$^+$06], an assumption that does not hold in the security framework. The classical fault detection filters [Jon73] do not guarantee identification of all possible adversarial signals and zero-dynamics attacks remain stealthy. As an alternative approach, robustification has been used in order to estimate the state despite sparse attacks by either deploying Kalman filters or principle component analysis [MB10, FGA11]. The main drawback of these methods is the absence of formal guarantees for the correctness of the state. In contrast, the method proposed in this work is guaranteed to construct the state correctly in spite of attacks on sensors *and/or* actuators if the number of attacked components is below a specified threshold that depends on the system. In a recent work [HO16], Harirchi et. al. proposed a novel fault detection approach using techniques from model invalidation. The authors pursued a worst-case scenario approach and therefore their framework is suitable for security. However, necessary and sufficient conditions for state estimation in a general adversarial setting were not investigated in [HO16]. In this work, we precisely characterize the class of systems, by providing necessary and sufficient conditions, for which state reconstruction is possible despite sensor and/or actuator attacks.

The contributions of this work can be summarized as follows:

- We introduce the notion of sparse strong observability by drawing inspiration from sparse observability [FTD14a, ST16b] and the classical notion of strong observability [Hau83]. We show this is the relevant property when the adversarial agent not only compromises sensor measurements but can also attack inputs.

- We develop an observer by leveraging the SMT approach to harness the exponential complexity of the problem. Our observer consists of two blocks interacting iteratively until the true state is found (see Section 2.4 for the detailed explanation of the observer's architecture).

- We propose two methods to further decrease the running time of the proposed algorithm by reducing the number of iterations of the observer. The first method exploits heuristics that can be efficiently computed at each iteration (see Section 2.4.3). The second method is inspired by the QuickXplain algorithm [Jun01] that efficiently finds an irreducibly inconsistent set (see Section 2.4.4). We demonstrate the scalability of our proposed observer by several numerical simulations.

This chapter is organized as follows. Section 2.2 gives the attack model and the precise problem formulation after establishing the notation. In Section 2.3, we introduce the notion of sparse strong observability and relate this notion to the problem of state reconstruction when some of the inputs and outputs are under adversarial attacks. This section concludes with the main theoretical contribution of this work that is Theorem 2.1. Section 2.4 is devoted to designing an observer by exploiting the SMT paradigm. Section 2.5 provides the simulation results followed by Section 2.6 that concludes the chapter.

2.2 Problem Definition

Notation. We denote the sets of real, natural and binary numbers by $\mathbb{R}$, $\mathbb{N}$ and $\mathbb{B}$. We represent vectors and real numbers by lowercase letters, such as u, x, y, and matrices with capital letters, such as A. Given a vector $x \in \mathbb{R}^n$ and a set $O \subseteq \{1, \ldots, n\}$, we use $x|_O$ to denote the vector obtained from x by removing all elements except those indexed by the set O. Similarly, for a matrix $C \in \mathbb{R}^{n_1 \times n_2}$ we use $C|_{(O_1,O_2)}$ to denote the matrix obtained from C by eliminating all rows and columns except the ones indexed by O_1 and O_2, respectively, where $O_i \subseteq \{1, \ldots, n_i\}$ with $n_i \in \mathbb{N}$ for $i \in \{1, 2\}$. In order to simplify the notation, we use $C|_{(.,O_2)} := C|_{(\{1,\ldots,n_1\},O_2)}$ and $C|_{(O_1,.)} := C|_{(O_1,\{1,\ldots,n_2\})}$. We denote the com-

plement of O by $\overline{O} := \{1, \ldots, n\} \setminus O$. We use the notation $\{x(t)\}_{t=0}^{T-1}$ to denote the sequence $x(0), \ldots, x(T-1)$, and we drop the sub(super)scripts whenever it is clear from the context.

A Linear Time Invariant (LTI) system is described by the following equations:

$$\begin{aligned} x(t+1) &= Ax(t) + Bu(t), \\ y(t) &= Cx(t) + Du(t), \end{aligned} \tag{2.1}$$

where $u(t) \in \mathbb{R}^m$, $x(t) \in \mathbb{R}^n$ and $y(t) \in \mathbb{R}^p$ are the input, state and output variables, respectively, $t \in \mathbb{N} \cup \{0\}$ denotes time, and A, B, C and D are system matrices with appropriate dimensions. We use (A, B, C, D) to denote the system described by (2.1). The order of an LTI system is defined as the dimension of its state space. A trajectory of a system is defined as its input sequence along with the output sequence. For an LTI system,

$$\mathcal{O}_{(A,C)} := \begin{bmatrix} C^T & A^T C^T & \ldots & (A^T)^{n-1} C^T \end{bmatrix}^T, \tag{2.2}$$

$$\mathcal{N}_{(A,B,C,D)} := \begin{bmatrix} D & 0 & \ldots & 0 \\ CB & D & \ldots & 0 \\ \vdots & & \ddots & \\ CA^{n-2}B & CA^{n-3}B & \ldots & D \end{bmatrix}, \tag{2.3}$$

are the *observability* and *invertibility* matrices, respectively, where n is the order of the system. In this paper, we often work with subsets of inputs and outputs. For a subset of outputs $\Gamma_y \subseteq \{1, \ldots, p\}$, we use the notation $\mathcal{O}_{\Gamma_y} := \mathcal{O}_{(A, C|_{(\Gamma_y, .)})}$ to denote the observability matrix of outputs in the set Γ_y. For a set of inputs $\Gamma_u \subseteq \{1, \ldots, m\}$, we use the notation $\mathcal{N}_{\Gamma_u \to \Gamma_y}$ to denote $\mathcal{N}_{(A, B_{(.,\Gamma_u)}, C_{(\Gamma_y,.)}, D_{(\Gamma_y,\Gamma_u)})}$. For $x \in \mathbb{R}^n$, we define its support set as the set of indices of its non-zero components, denoted by $\mathrm{supp}(x)$. Similarly we define the support of the sequence $\{x(t)\}$ as $\mathrm{supp}(\{x(t)\}) := \cap_t \mathrm{supp}(x(t))$. The observer proposed in this paper uses batches of inputs and outputs in order to reconstruct the state. We reserve capital bold letters to denote these batches,

$$\mathbf{Y}^\tau(t) := \begin{bmatrix} y(t-\tau+1)^T & \ldots & y(t)^T \end{bmatrix}^T, \tag{2.4}$$

$$\mathbf{U}^\tau(t) := \begin{bmatrix} u(t-\tau+1)^T & \ldots & u(t)^T \end{bmatrix}^T, \tag{2.5}$$

where $\tau \leq n$. Whenever τ is the order of the underlying system, we may drop the superscript for ease of notation. For a subset of outputs (inputs), denoted by $\Gamma_y \subseteq \{1, \ldots, p\}$ ($\Gamma_u \subseteq \{1, \ldots, m\}$), we use the notation $\mathbf{Y}^\tau|_{\Gamma_y}(t)$ ($\mathbf{U}^\tau|_{\Gamma_u}(t)$) for the batches of length τ that only consists of outputs (inputs) in the set Γ_y (Γ_u). For a vector $x \in \mathbb{R}^n$, we denote a generic norm, l_2-norm and l_1-norm of x by $\|x\|$, $\|x\|_2$ and $\|x\|_1$.

2.2.1 System and Attack Model

This work is concerned with the problem of state reconstruction of LTI systems. We consider the scenario in which sensors and actuators are both prone to adversarial attacks. The ultimate goal is to reconstruct the state despite these attacks. In this part, we define the attack model and conclude this section with the precise problem statement.

System S, is described by the following equations:

$$\begin{aligned} x(t+1) &= Ax(t) + Bu_S(t), \\ y_S(t) &= Cx(t) + Du_S(t). \end{aligned} \tag{2.6}$$

Without loss of generality we assume $\begin{bmatrix} B^T & D^T \end{bmatrix}^T$ to be of full column rank.

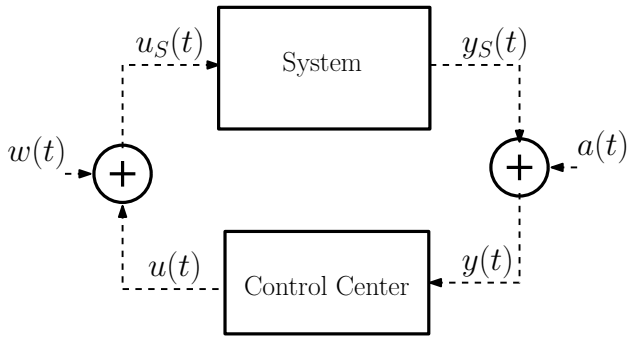


Figure 2.1: The generic attack model considered in this paper.

Each actuator (sensor) corresponds to one input (output) and we use input/output terminology instead of actuator/sensor in the rest of this paper. In this set up the adversary can attack both inputs and outputs. We model these attacks by additive terms and by imposing

a sparsity constraint on them,

$$\begin{cases} u_S(t) &= u(t) + w(t), \\ y(t) &= y_S(t) + a(t), \end{cases} \tag{2.7}$$

where $u(t) \in \mathbb{R}^m$ and $y(t) \in \mathbb{R}^p$ are the controller-designed input and the observed output, respectively, and $w(t) \in \mathbb{R}^m$ and $a(t) \in \mathbb{R}^p$ are signals injected by the malicious agent. In the rest of this paper, we refer to these signals $(w\,(t)\,, a(t))$ as the attack of the adversarial agent. We use the subscript S for signals that directly come from/to the system. The controller can only observe $y(t)$ and compute the input $u(t)$. This generic attack model is depicted in Figure 2.1.

When the adversary attacks an input (output) it can change its value to any arbitrary number without explicitly revealing its presence. The only limitation that we impose on the power of the malicious agent is the maximal number of inputs and outputs that can be attacked.

Assumption 2.1 (Bound on the number of attacks)**.** *The number of inputs and outputs under attack are bounded by r and s, respectively.*

Therefore, the malicious agent can attack a subset of inputs and outputs denoted by $\Gamma_u \subseteq \{1, \ldots, m\}$ and $\overline{\Gamma}_y \subseteq \{1, \ldots, p\}$,[1] respectively, with $|\Gamma_u| \leq r$ and $|\overline{\Gamma}_y| \leq s$, such that $\text{supp}(\{w(t)\}) \subseteq \Gamma_u$ and $\text{supp}(\{a(t)\}) \subseteq \overline{\Gamma}_y$. Note that these sets are not known to the controller and only upper bounds on their cardinality are given. Once the adversary chooses these sets, inputs and outputs outside these sets remain attack-free. This assumption is realistic when the time it takes for the adversarial agent to attack new inputs and outputs is large compared to the time scale of the system.

We now precisely define the main problem we tackle in this paper.

[1]For ease of exposition, we use Γ_u to denote under-attack inputs while Γ_y is reserved for the set of attack-free outputs, therefore, the set of under-attack outputs is represented by $\overline{\Gamma}_y := |\{1, \ldots, p\} \setminus \Gamma_y$ in this paper.

Problem 2.1 (Secure state estimation)**.** *For the linear system defined by* (2.6) *under the attack model defined by* (2.7)*, what are necessary and sufficient conditions under which the state of the compromised system* (2.6) *can be reconstructed with bounded delay?*

It is well-known that the secure state estimation problem, when only outputs are under adversarial attacks, is combinatorial and belongs to the class of *NP-hard* problems [SNP+17, PDB13]. Therefore we are motivated to design an observer that harness the complexity of this problem.

Problem 2.2 (Secure observer design)**.** *Assumming conditions in Problem 2.1 are satsified, how can we design an observer that reconstructs the state of the compromised system?*

2.3 Condition for Secure State Estimation

In this section, we solve Problem 2.1, i.e., we provide conditions on the system described by (2.6) under which state reconstruction (with bounded delay) is possible. We first develop the notion of sparse strong observability. This section concludes with Theorem 2.1 that relates this notion to the solution of Problem 2.1.

In the absence of attacks, the problem of estimating the state of a system while some of the inputs are unknown has been studied and the notion of strong observability was introduced in the literature [Hau83]. For strongly observable systems, it is possible to estimate the state of the system without the knowledge of inputs. The following definition formalizes this concept.

Definition 2.1 (Strong observability)**.** *An LTI system is called strongly observable if for any initial state* $x(0) \in \mathbb{R}^n$ *and any input sequence* $\{u(t) \in \mathbb{R}^m\}_{t=0}^{\infty}$ *there exists an integer* $\tau \in \mathbb{N} \cup \{0\}$ *such that* $x(0)$ *can be uniquely recovered from* $\{y(t)\}_{t=0}^{\tau}$*.*

Note that τ is always upper-bounded by the order of the system. Linearity implies the following lemma.

Lemma 2.1. *An LTI system is strongly observable if and only if* $y(t) = 0, \forall t \in \mathbb{N} \cup \{0\}$ *implies that* $x(0) = 0$.

Proof. See Appendix A. □

It is straightforward to conclude the following corollary.

Corollary 2.1. *An LTI system is not strongly observable if and only if there exist a non-zero intial state and an input sequence such that* $y(t) = 0$ *for* $t \in \mathbb{N} \cup \{0\}$.

Proof. Follows directly from Lemma 2.1. □

It is well-understood that when the adversary is restricted to attacking outputs, state reconstruction is possible only if there is enough redundancy in the outputs of the system. This redundancy can be stated in terms of observability of the system while removing a number of outputs. This property has been formalized in [FTD14a] and is called sparse observability [ST16b]. By analogy with sparse observability, we define the notion of (r, s)-sparse strong observability as follows:

Definition 2.2 ((r, s)-sparse strong observability)**.** *An LTI system* (A, B, C, D) *with* m *inputs and* p *outputs is* (r, s)*-sparse strongly observable if for any* $\Gamma_u \subseteq \{1, \ldots, m\}$ *and* $\Gamma_y \subseteq \{1, \ldots, p\}$ *with* $|\Gamma_u| \leq r$ *and* $|\Gamma_y| \geq p - s$*, the system* $(A, B_{(.,\Gamma_u)}, C_{(\Gamma_y,.)}, D_{(\Gamma_y,\Gamma_u)})$ *is strongly observable.*

Note that in Definition 2.2, the value of r and s are upper bounded by the number of inputs and outputs, respectively. This modified notion of strong observability is the key for formalizing redundancy across inputs and outputs. We show that a necessary and sufficient condition for secure state estimation can be stated using this property. Note that $(0, s)$-sparse strong observability is equivalent to the notion of s-sparse observability that was introduced before in the literature [FTD14a, ST16b, MSK+17]. The following theorem is the main theoretical result in this paper.

Theorem 2.1. *Let the number of attacked inputs and outputs be bounded by r and s, respectively. Under the attack model (2.7), the state can be reconstructed (possibly with delay) if and only if the underlying system is $(2r, 2s)$-sparse strongly observable.*

Remark 2.1. *It is worth mentioning that the maximum number of attacked outputs, s, cannot be greater than $\lfloor \frac{p}{2} \rfloor$, which is an inherent limitation of LTI systems with p outputs [FTD14a]. However the maximum number of attacked inputs is not inherently restricted by $\lfloor \frac{m}{2} \rfloor$ and can take values up to m, depending on the specific system under the consideration.*

Remark 2.2. *Pasqualetti et. al. [PDB13] addressed the problem of attack detection and identification in the presence of adversarial inputs and outputs for continuous-time LTI systems. They showed that attack identification is possible if and if for any $\Gamma_u \subseteq \{1, \ldots, m\}$ and $\Gamma_y \subseteq \{1, \ldots, p\}$ with $|\Gamma_u| \leq 2r$ and $|\Gamma_y| \geq p - 2s$, the system $(A, B_{(.,\Gamma_u)}, C_{(\Gamma_y,.)}, D_{(\Gamma_y,\Gamma_u)})$ does not have any invariant zeros.*

It is clear that from the state and the dynamics of the system, the attack can be identified, therefore the attack identification comes free with the solution to the secure estimation problem. Strongly observable LTI systems do not have any invariant zeros (see, for example Theorem 1.8 in [Hau83]). Therefore this theorem shows that under this sparse-attack model, the conditions for identifying the attack also enable one to reconstruct the state, i.e., characterizations of attack identifiability and secure state estimation are equivalent for LTI systems. Putting these together, secure state estimation also comes with the solution to the attack identification problem. However, we provide a direct proof that does not require this machinery.

Proof. First we show that $(2r, 2s)$-sparse strong observability is a sufficient condition for correctly estimating the state. For the sake of the contradiction, assume that the state cannot be reconstructed, i.e., there exist two different (initial) states, denoted by $x^{(1)}$ and $x^{(2)}$, that cannot be distinguished under this attack model. More precisely, there exist two attack strategies that will lead to the same exact (observed) trajectories. We reserve superscripts $.^{(1)}$ and $.^{(2)}$ for variables across those scenarios. Let us denote the adversarial additive

terms by $\{w^{(1)}(t)\}, \{a^{(1)}(t)\}$ and $\{w^{(2)}(t)\}, \{a^{(2)}(t)\}$. We represent the corresponding inputs and outputs of the system by $\{u_S^{(1)}(t)\}, \{y_S^{(1)}(t)\}$ and $\{u_S^{(2)}(t)\}, \{y_S^{(2)}(t)\}$, and the common (corrupted) measured output and the controller input sequences are denoted by $\{y(t)\}$ and $\{u(t)\}$, respectively.

The attack model (2.7) implies that there exist $\Gamma_u^{(i)}, \overline{\Gamma}_y^{(i)}$ for $i \in \{1, 2\}$ with bounded cardinality such that

$$\text{supp}(\{w^{(i)}(t)\}) \subseteq \Gamma_u^{(i)}, \quad \text{supp}(\{a^{(i)}(t)\}) \subseteq \overline{\Gamma}_y^{(i)}, \tag{2.8}$$

Recall from the attack model,

$$\begin{cases} u_S^{(1)}(t) = u(t) + w^{(1)}(t) \\ u_S^{(2)}(t) = u(t) + w^{(2)}(t) \end{cases}, \tag{2.9}$$

where $u(t)$ is the controller designed input. Putting (2.8) and (2.9) together,

$$\begin{aligned} \text{supp}(\{u_S^{(1)}(t) - u_S^{(2)}(t)\}) &= \text{supp}(\{w^{(1)}(t) - w^{(2)}(t)\}) \\ &\subseteq \Gamma_u^{(1)} \cup \Gamma_u^{(2)}. \end{aligned} \tag{2.10}$$

Similarly, it is straightforward to conclude that $\text{supp}(\{y_S^{(1)}(t) - y_S^{(2)}(t)\}) \subseteq \overline{\Gamma}_y^{(1)} \cup \overline{\Gamma}_y^{(2)}$. We are ready to reach the contradiction. The underlying system is LTI, thus the input sequence $\{u_S^{(1)}(t) - u_S^{(2)}(t)\}$ with the initial state $x^{(1)} - x^{(2)}$ generates the output sequence $\{y_S^{(1)}(t) - y_S^{(2)}(t)\}$. The underlying system is $(2r, 2s)$-sparse strongly observable so the sub-system $(A, B_{(.,\Gamma_u)}, C_{(\Gamma_y,.)}, D_{(\Gamma_y,\Gamma_u)})$ is strongly observable for any $|\Gamma_u| = 2r$ and $|\Gamma_y| = p - 2s$. Let us choose Γ_u and Γ_y as any set of $2r$ inputs and $p - 2s$ outputs such that,

$$\Gamma_u^{(1)} \cup \Gamma_u^{(2)} \subseteq \Gamma_u, \quad \Gamma_y \subseteq \Gamma_y^{(1)} \cap \Gamma_y^{(2)}. \tag{2.11}$$

Note that $\{y_S^{(1)}(t)|_{\Gamma_y} - y_S^{(2)}(t)|_{\Gamma_y}\}$ is a zero sequence, together with Lemma 2.1 we conclude that the corresponding initial state $(x^{(1)} - x^{(2)})$ is zero, which contradicts the assumption of $x^{(1)} \neq x^{(2)}$ and therefore the proof is complete.

Now we prove that $(2r, 2s)$-sparse strongly observability is a necessary condition. For the sake of contradiction, suppose that the system described by (2.6) is not $(2r, 2s)$-sparse

strongly observable, however, reconstructing the state (possibly with delays) is still possible. We construct two system trajectories with different (initial) states that have exactly the same input and output sequences under suitable attack strategies (additive terms). This implies that estimating the correct state is indeed impossible thereby establishing the desired contradiction.

By the assumption of the contradiction, the underlying system is not $(2r, 2s)$-sparse strongly observable, so there exist subsets of inputs and outputs denoted by Γ_u with $|\Gamma_u| = 2r$ and Γ_y with $|\Gamma_y| = p - 2s$, respectively, such that $(A, B_{(.,\Gamma_u)}, C_{(\Gamma_y,.)}, D_{(\Gamma_y,\Gamma_u)})$ is not strongly observable. Corollary 2.1 implies that there exist an initial condition Δx and an input sequence $\{\Delta u(t)\}$ (with its support lying inside Γ_u) that generates an output sequence $\{\Delta y(t)\}$ with $\text{supp}(\{\Delta y(t)\}) \subseteq \overline{\Gamma}_y$. One can rewrite $\Delta u(t)$ and $\Delta y(t)$ as sum of two sparse signals, more precisely:

$$\Delta u(t) = \Delta u^{(1)}(t) + \Delta u^{(2)}(t), \tag{2.12}$$

$$\Delta y(t) = \Delta y^{(1)}(t) + \Delta y^{(2)}(t), \tag{2.13}$$

where cardinality of $\text{supp}(\{\Delta u^{(i)}(t)\})$ and $\text{supp}(\{\Delta y^{(i)}(t)\})$ are upper-bounded by r and s for $i \in \{1, 2\}$, respectively. For example, we can rewrite $\overline{\Gamma}_y = \overline{\Gamma}_y^{(1)} \cup \overline{\Gamma}_y^{(2)}$ where $|\overline{\Gamma}_y^{(i)}| \leq s$ for $i \in \{1, 2\}$. Then we define

$$\begin{cases} \Delta y^{(i)}(t)|_{\overline{\Gamma}_y^{(i)}} & := \Delta y(t)|_{\overline{\Gamma}^{(i)}} \\ \Delta y^{(i)}(t)|_{\Gamma_y^{(i)}} & := 0 \end{cases}, \quad \text{for} \quad i \in \{1, 2\}.$$

Now consider the following two different trajectories of the system

$$\begin{cases} u_S^{(1)}(t) & = \Delta u(t) \\ y_S^{(1)}(t) & = \Delta y(t) \end{cases}, \quad \begin{cases} u_S^{(2)}(t) & = 0 \\ y_S^{(2)}(t) & = 0 \end{cases} \tag{2.14}$$

with their initial states

$$\begin{cases} x^{(1)}(0) & = \Delta x \\ x^{(2)}(0) & = 0 \end{cases}, \tag{2.15}$$

and their corresponding attack strategies,

$$\begin{cases} w^{(1)}(t) &= \Delta u^{(1)}(t) \\ a^{(1)}(t) &= -\Delta y^{(1)}(t) \end{cases}, \quad \begin{cases} w^{(2)}(t) &= -\Delta u^{(2)}(t) \\ a^{(2)}(t) &= \Delta y^{(2)}(t) \end{cases}. \tag{2.16}$$

It is straightforward to verify that $\{y^{(1)}(t)\} = \{y^{(2)}(t)\}$ and $\{u^{(1)}(t)\} = \{u^{(2)}(t)\}$, *i.e.*, under the attack model (2.7) the controlled inputs and the observed outputs are exactly the same for both trajectories while having different (initial) states. We reached the contradiction and the proof is complete. □

2.4 Secure Observer Design

In this section, we seek solutions to Problem 2.2. In the first part, we explain the intuition behind the proposed algorithm that estimates the state despite attacks on inputs and outputs. We give formal guarantees that the algorithm reconstructs the state correctly. In the second part, we introduce the observer by leveraging the SMT paradigm followed by two methods that enhance the run time of state estimation.

Based on the attack model (2.7), the input to the system is decomposed into two additive terms, the controller-designed input $u(t)$ and the adversarial input $w(t)$. The underlying system (2.6) is linear and therefore we can easily exclude the effect of the controller-designed input from the output by subtracting its effect. Hence, without loss of generality we assume that the true $u(t)$ is zero.

The proposed algorithm is based on the following proposition.

Proposition 2.1. *Suppose the underlying system is $(2r, 2s)$-sparse strongly observable, and the number of attacked inputs and outputs are bounded by r and s, respectively. Given any subset of inputs and outputs denoted by Γ_u and Γ_y with $|\Gamma_u| \leq r$ and $|\Gamma_y| \geq p - s$, the first statement below implies the second:*

1. *There exist $\hat{\mathbf{U}} \in \mathbb{R}^{n|T|}$ and $\hat{x} \in \mathbb{R}^n$ such that*

$$\mathbf{Y}|_{\Gamma_y}(t) = \mathcal{O}_{\Gamma_y}\hat{x} + \mathcal{N}_{\Gamma_u \to \Gamma_y}\hat{\mathbf{U}}. \tag{2.17}$$

2. *The estimated state $\hat{x}$, is equal to the actual state of the system at time $t-n+1$, $x(t-n+1)$, where n is the order of the underlying system.*

Remark 2.3. *The underlying system is $(2r, 2s)$-sparse strongly observable and therefore $(A, B_{(.,\Gamma_u)}, C_{(\Gamma_y,.)}, D_{(\Gamma_y,\Gamma_u)})$ is strongly observable. If (2.17) has a solution, then $\hat{x}$ would be the unique solution for x (see Section III-B of [YB75]).*

Proof. Let us denote the set of attack-free outputs and under-attack inputs by Γ_y^* and Γ_u^*. At most s outputs are under attack, therefore $|\Gamma_y \cap \Gamma_y^*| \geq p-2s$. Note that $\mathbf{Y}|_{\Gamma_y \cap \Gamma_y^*}$ can be written as follows:

$$\mathbf{Y}|_{\Gamma_y \cap \Gamma_y^*} = O_{\Gamma_y \cap \Gamma_y^*} x(t-n+1) + \mathcal{N}_{\Gamma_u \to \Gamma_y \cap \Gamma_y^*} \mathbf{W}|_{\Gamma_u} + \mathcal{N}_{\Gamma_u^* \setminus \Gamma_u \to \Gamma_y \cap \Gamma_y^*} \mathbf{W}|_{\Gamma_u^* \setminus \Gamma_u}. \tag{2.18}$$

On the other hand, we can rewrite (2.17) by taking only outputs in $\Gamma_y \cap \Gamma_y^*$,

$$\mathbf{Y}|_{\Gamma_y \cap \Gamma_y^*} = O_{\Gamma_y \cap \Gamma_y^*} \hat{x} + \mathcal{N}_{\Gamma_u \to \Gamma_y \cap \Gamma_y^*} \hat{\mathbf{U}} + \mathcal{N}_{\Gamma_u^* \setminus \Gamma_u \to \Gamma_y \cap \Gamma_y^*} \mathbf{0}, \tag{2.19}$$

where $\mathbf{0}$ is a zero vector with appropriate dimensions. The underlying system is $(2r, 2s)$-sparse strongly observable, therefore we conclude that the sub-system consisting of inputs $\Gamma_u \cup \Gamma_u^*$ and outputs $\Gamma_y \cap \Gamma_y^*$ denoted by $\hat{S} := (A, B_{(.,\Gamma_u \cup \Gamma_u^*)}, C_{(\Gamma_y \cap \Gamma_y^*,.)}, D_{(\Gamma_y \cap \Gamma_y^*, \Gamma_u \cup \Gamma_u^*)})$ is strongly observable. One can reinterpret both equations as two (possibly different) valid trajectories of the system $\hat{S}$ that share the same output sequence. Strong observability of $\hat{S}$ implies that $\hat{x} = x(t-n+1)$ which completes the proof. □

The main algorithm in this paper builds upon this proposition. We search for a set of inputs and outputs that satisfies equality (2.17), *i.e.*, we check if there exist $\hat{\mathbf{U}}$ and $\hat{x}$ that make equality (2.17) hold. Based on Proposition 2.1, we define a consistency check as follows,

Test 2.1 (Consistency Check)**.** *Given subsets of inputs and outputs denoted by Γ_u and Γ_y,* TEST(Γ_u, Γ_y) *returns true if*

$$\min_{\hat{\mathbf{U}}, \hat{x}} \|\mathbf{Y}|_{\Gamma_y} - \mathcal{O}_{\Gamma_y} \hat{x} - \mathcal{N}_{\Gamma_u \to \Gamma_y} \hat{\mathbf{U}}\| \leq \epsilon, \tag{2.20}$$

where $\epsilon > 0$ is the solver tolerance, due to numerical errors. However, for the sake of clarity, we focus in this paper on the case when ϵ is negligible[2].

Finding the right subset of inputs and outputs that satisfies this test is a combinatorial problem in nature and requires exhaustive search. It is well-known that secure state estimation under this attack model is in general *NP-hard* [SNP+17, PDB13]. This test is depicted in Algorithm 2.

In the rest of this section, we introduce an architecture for our observer followed by methods to improve its computational performance. For each input (output), we assign a binary variable $\mathbf{b}_i \in \mathbb{B}$ ($\mathbf{c}_i \in \mathbb{B}$) that indicates if the corresponding input (output) is under attack or not, *i.e.*, $\mathbf{b}_i = 1$ ($\mathbf{c}_i = 1$) means that the i^{th} input (output) is under attack. In the rest of this paper, we use the bold letters ($\mathbf{b}$ and $\mathbf{c}$) to denote these Boolean variables and we reserve non-bold type face (b and c) as instances of them. Finding the right assignment of these Boolean variables is combinatorial in nature; to efficiently decide which set of inputs and outputs satisfies the TEST in (2.20), we design an observer using the lazy SMT paradigm [BSST09].

2.4.1 Overall Architecture

The observer consists of two blocks that interact with each other, a propositional satisfiability (SAT) solver and a theory solver. The former reasons about the combination of Boolean and pseudo-Boolean constraints and produces a feasible instance of $\mathbf{b} \in \mathbb{B}^m$ and $\mathbf{c} \in \mathbb{B}^p$ based on its current state. The theory solver checks the consistency of Boolean variables using the consistency test, and when the test fails, it encodes the inconsistency as a pseudo-Boolean constraint and returns it to the SAT solver. The general architecture is depicted in Figure 2.2.

The initial pseudo-Boolean constraint only bounds the number of attacked inputs and

[2] *Note that the minimum always exists for* (2.20) *as the cost function is a semi-definite quadratic function.*

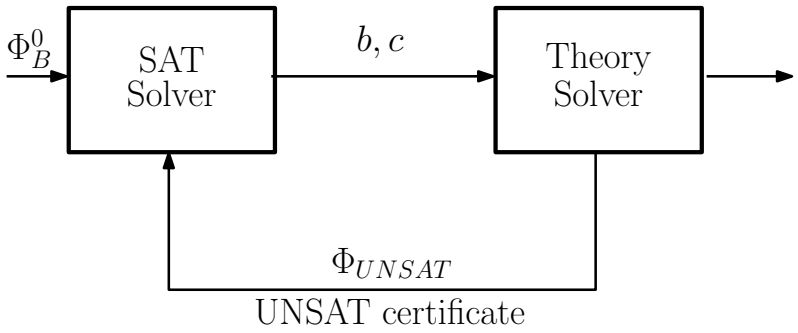


Figure 2.2: The lazy SMT paradigm architecture.

outputs,

$$\Phi_B := (\sum_{i=1}^{m} \mathbf{b}_i \leq r) \bigwedge (\sum_{j=1}^{p} \mathbf{c}_i \leq s). \tag{2.21}$$

The SAT solver generates instances of $\mathbf{b}$ and $\mathbf{c}$ that satisfy Φ_B. The theory solver checks whether $\Gamma_u := \text{supp}(b)$ and $\Gamma_y := \overline{\text{supp}(c)}$ satisfies the consistency check. If the test is satisfied, then the algorithm terminates and returns the (delayed) estimate of the state. Otherwise, the theory solver outputs UNSAT and generates a reason for the conflict, a certificate, or a counterexample that is denoted by Φ_{cert}. This counterexample encodes the inconsistency among the chosen inputs and outputs. The following always constitutes a naive certificate.

$$\Phi_{\text{naive-cert}} := \sum_{i \in \overline{\text{supp}(b)}} \mathbf{b}_i + \sum_{j \in \overline{\text{supp}(c)}} \mathbf{c}_j \geq 1. \tag{2.22}$$

On the next iteration, the SAT solver updates the constraint by conjoining Φ_{cert} to Φ_B, and generates another feasible assignment for $\mathbf{b}$ and $\mathbf{c}$. This procedure is repeated until the theory solver returns SAT as illustrated in Algorithm 1.

Note that Proposition 2.1 implies that the SAT solver eventually produces an assignment that satisfies the consistency test and therefore Algorithm 1 always terminates. The size of the certificate plays an important role in the overall execution time of the algorithm [SNP+17]. The attack model considered in [SNP+17] is restricted to outputs, and the major contribution of our work is to handle both input and output attacks. In the next section, we focus on constructing shorter counterexamples to improve the run time.

Algorithm 1: Secure state estimator

input : A, B, C, D (system), Y (output), r, s (bounds);

status $\leftarrow$ UNSAT ;

$\Phi_B \leftarrow (\sum_{i\in\{1,\dots,m\}} \mathbf{b}_i \leq r) \bigwedge (\sum_{i\in\{1,\dots,p\}} \mathbf{c}_i \leq s)$;

while status == UNSAT **do**

  $(b, c) \leftarrow$ SAT-solver(Φ_B) ;

  (status, x) $\leftarrow$ T-solver.check(supp(b), $\overline{\text{supp}(c)}$);

  $\Phi_{\text{cert}} \leftarrow$ T-solver.Certificate(supp(b), $\overline{\text{supp}(c)}$);

  $\Phi_B \leftarrow \Phi_B \bigwedge \Phi_{\text{cert}}$;

end

return (x, b, c);

Algorithm 2: T-solver.check

input : Γ_u, Γ_y;

Solve: $(\hat{x}, \hat{\mathbf{U}}) = \text{argmin}_{x,\mathbf{U}} \|\mathbf{Y}|_{\Gamma_y} - \mathcal{O}_{\Gamma_y} x - \mathcal{N}_{\Gamma_u \to \Gamma_y} \mathbf{U}\|$;

if $\|\mathbf{Y}|_{\Gamma_y} - \mathcal{O}_{\Gamma_y} \hat{x} - \mathcal{N}_{\Gamma_u \to \Gamma_y} \hat{\mathbf{U}}\| \leq \epsilon$ **then**

  status $\leftarrow$ SAT ;

else

  status $\leftarrow$ UNSAT ;

end

return (status, $\hat{x}$)

2.4.2 SAT Certificate

In this part, we improve the efficiency of Algorithm 1 by constructing a shorter certificate (counter-example or conflicts). As it was discussed before, the naive certificate only excludes the current assignment of **b** and **c** from the search space of the SAT solver, however, by exploiting the structure of the underlying system, we show that we can further decrease the size of the certificate and therefore prune the search space more efficiently.

One of the main results of this work is to show that we can always find a smaller conflicting subset of inputs and outputs. We propose two methods for generating shorter certificates. The first method reduces the size of the counterexample by at least $s-1$, we explain this method in Lemma 2.2 and give a formal proof of the existence of such shorter certificate. In practice, however we observe the reduction in the length of conflicts is much larger than this theoretical bound. The second method is inspired by the QuickXplain algorithm. This method generates counter-examples that are irreducible, meaning that we cannot reduce the size of the counter-example by removing some of its entries. We also note that by generating multiple certificates at each iteration we can further enhance the execution time. At the end of this section Lemma 2.3 states that for a generic LTI system the size of the certificate cannot be smaller than $m+1$.

Let us assume that the SAT solver hypothesized $\Gamma_u^{\text{SAT}} := \text{supp}(b)$ and $\Gamma_y^{\text{SAT}} := \overline{\text{supp}(c)}$ as the set of compromised inputs and safe outputs, respectively. Recall that the certificate consists of inputs in $\overline{\Gamma}_u^{\text{cert}}$ and outputs in Γ_y^{cert}. The main intuition behind both methods is to look for $\Gamma_u^{\text{cert}} \supseteq \Gamma_u^{\text{SAT}}$ and $\Gamma_y^{\text{cert}} \subseteq \Gamma_y^{\text{SAT}}$ that would not satisfy the consistency test.

2.4.3 Method I: based on heuristics

Method I reduces the size of the certificate by increasing the size of (supposedly under attack) inputs (Γ_u^{cert}) followed by decreasing the size of (supposedly safe) outputs (Γ_u^{cert}). The summary of the above procedure of shortening certificates is illustrated in Algorithm 3. We begin by adding inputs to Γ_u^{SAT} while making sure TEST still returns false and the

number of inputs is bounded by $2r$. Let us denote this new set of inputs by Γ_u^{cert}.

At the second step, we shrink the set of conflicting outputs in order to further shorten the size of the counterexample. Let us denote a subset of Γ_y^{SAT} of size $p-2s$ by Γ_y^{temp}. The following lemma shows we can reduce the size of conflicting outputs at least by $s-1$.

Algorithm 3: T-solver.Certificate 1

input : $\Gamma_u^{\text{SAT}}, \Gamma_y^{\text{SAT}}$;

`/* step 1: Conduct a linear search in the input set */`

Sort $\overline{\Gamma}_u^{\text{SAT}}$;

status $\leftarrow$ UNSAT, $j \leftarrow \emptyset, \Gamma_u^{\text{cert}} \leftarrow \Gamma_u^{\text{SAT}}$;

while status == UNSAT ***and*** $|\Gamma_u^{cert}| < 2r$ **do**

  $\Gamma_u^{\text{cert}} \leftarrow \Gamma_u^{\text{cert}} \cup \{j\}$;

  Pick another input $j \in \overline{\Gamma}_u^{\text{SAT}}$;

  (status, x) $\leftarrow$ T-Solver.check($\Gamma_u^{\text{cert}} \cup \{j\}, \Gamma_y^{\text{SAT}}$) ;

end

`/* step 2: Conduct a linear search in the output set */`

Sort Γ_y^{SAT} ;

Pick a subset of size $p-2s$: $\Gamma_y^{\text{temp}} \subseteq \Gamma_y^{\text{SAT}}$;

status $\leftarrow$ SAT, $i \leftarrow \emptyset$;

while status == SAT **do**

  $\Gamma_y^{\text{cert}} \leftarrow \Gamma_y^{\text{temp}} \cup \{i\}$;

  (status, x) $\leftarrow$ T-Solver.check($\Gamma_u^{\text{cert}}, \Gamma_y^{\text{cert}}$) ;

  Pick another output $i \in \Gamma_y^{\text{SAT}} \setminus \Gamma_y^{\text{temp}}$;

end

$\Phi_{\text{cert}}^1 \leftarrow \sum_{j \in \overline{\Gamma}_u^{\text{cert}}} \mathbf{b}_j + \sum_{i \in \Gamma_y^{\text{cert}}} \mathbf{c}_i \geq 1$;

return Φ_{cert}^1 ;

Lemma 2.2. *Assume that System S is $(2r, 2s)$-sparse strongly observable, and the number of*

attacked inputs and outputs are bounded by r and s, respectively. Pick any subset of inputs and outputs denoted by Γ_u^{cert} and Γ_y^{SAT} with $|\Gamma_u^{cert}| \leq 2r$ and $|\Gamma_y^{SAT}| \geq p - s$, that do not satisfy the consistency check (2.20)*. Given any subset of at most $p - 2s$ outputs denoted by $\Gamma_y^{temp} \subseteq \Gamma_y^{SAT}$, one of the following is true:*

1. $\text{TEST}(\Gamma_u^{cert}, \Gamma_y^{temp})$ *returns false.*
2. *There exists an output $i \in \Gamma_y^{SAT} \setminus \Gamma_y^{temp}$ such that* $\text{TEST}(\Gamma_u^{cert}, \Gamma_y^{temp} \cup \{i\})$ *returns false.*

Proof. See Appendix A. □

We denote this smaller set of conflicting outputs Γ_y^{temp}(if $\text{TEST}(\Gamma_u^{\text{cert}}, \Gamma_y^{\text{temp}})$ returns false, otherwise $\Gamma_y^{\text{temp}} \cup \{i\}$) by Γ_y^{cert}. Lemma 2.2 gives formal guarantee of the existence of shorter certificates that holds no matter how the subsets of inputs and outputs (Γ_u^{temp} and Γ_y^{temp}) are chosen. This lemma shows that Method I reduces the size of the certificate by at least $s - 1$.

In practice, we choose these subsets based on heuristics that have for objective a decrease in the overall running time. We assign slack variables to inputs and outputs similarly to [SNP+17] and [SSC+17], and sort them based on the structure of the system. Recall that Algorithm 3 shortens the certificate by reducing the number of inputs followed by the reduction in the number of outputs, *i.e.*, we *simultaneously* reducing both inputs and outputs in the certificate. We observe that by generating two counterexamples, we can prune the search space of the SAT solver more efficiently. Similarly to Algorithm 5, we can find two counterexamples by reducing the number of inputs following a reduction in the number of outputs and vice-verse.

Sorting $\overline{\Gamma}_u^{\mathbf{SAT}}$ and $\Gamma_y^{\mathbf{SAT}}$. Assuming $\text{TEST}(\Gamma_u^{\text{SAT}}, \Gamma_y^{\text{SAT}})$ returns false, we assign slack variables to inputs in $\overline{\Gamma}_u^{\text{SAT}}$ and outputs in Γ_y^{SAT}, denoted by $\text{slack}_u(j)$ and $\text{slack}_y(i)$, respectively. Let us denote a solution to the optimization (2.20) inside $\text{TEST}(\Gamma_u^{\text{SAT}}, \Gamma_y^{\text{SAT}})$ by $\hat{x}$ and $\hat{\mathbf{U}}$.

We define $\text{slack}_u(j)$ for $j \in \overline{\Gamma}_u^{\text{SAT}}$ as the norm of the projection of $\mathbf{Y}|_{\Gamma_y^{\text{SAT}}} - \mathcal{O}_{\Gamma_y^{\text{SAT}}}\hat{x} - \mathcal{N}_{\Gamma_u^{\text{SAT}} \to \Gamma_y^{\text{SAT}}}\hat{\mathbf{U}}$ onto the column space of $\mathcal{N}_{j \to \Gamma_y^{\text{SAT}}}$,

$$\text{slack}_u(j) := \|\mathcal{N}_{j \to \Gamma_y^{\text{SAT}}}\mathcal{N}_{j \to \Gamma_y^{\text{SAT}}}^{\dagger}\left(\mathbf{Y}|_{\Gamma_y^{\text{SAT}}} - \mathcal{O}_{\Gamma_y^{\text{SAT}}}\hat{x} - \mathcal{N}_{\Gamma_u^{\text{SAT}} \to \Gamma_y^{\text{SAT}}}\hat{\mathbf{U}}\right)\|.$$

This slack variable measures how much of the residual can be justified by considering j in addition to Γ_u^{SAT}. Recall that we want to append inputs to Γ_u^{SAT} while having a false TEST. We proceed by normalizing these slack variables by the norm of the corresponding invertibility matrix, and $\overline{\Gamma}_u^{\text{SAT}}$ is obtained by sorting slack variables in *ascending* order.

We define $\text{slack}_y(i)$ as the residual of each output:

$$\text{slack}_y(i) := \|\mathbf{Y}|_i - \mathcal{O}_i\hat{x} - \mathcal{N}_{\Gamma_u^{\text{SAT}} \to \{i\}}\mathbf{U}\|, \quad i \in \Gamma_y^{\text{SAT}}. \tag{2.23}$$

Note that,

$$\sum_{i \in \Gamma_u^{\text{SAT}}} \text{slack}_y(i) = \min_{\hat{\mathbf{U}}, \hat{x}} \|\mathbf{Y}|_{\Gamma_y^{\text{SAT}}} - \mathcal{O}_{\Gamma_y^{\text{SAT}}}\hat{x} - \mathcal{N}_{\Gamma_u^{\text{SAT}} \to \Gamma_y^{\text{SAT}}}\hat{\mathbf{U}}\|. \tag{2.24}$$

We normalize each slack variable by the norm of the corresponding observability matrix. Recall that we aim to find a smaller subset of Γ_u^{SAT} while ensuring TEST returns false. We pick the output with the highest slack variable as the first element of Γ_u^{SAT}. We sort the rest based on the dimension of the kernel of each observability matrix, following the intuition provided in [SNP+17].

2.4.4 Method II: based on QuickXplain

The second method (Algorithm 5) is inspired by QuickXplain and generates a counter-example by pruning the naive-certificate (2.22) to make it irreducible. We formally define this property as follows,

Definition 2.3 (Irreducible certificate)**.** *A certificate consisting of inputs $\overline{\Gamma}_u$ and outputs Γ_y is irreducible, if no other subset of it can generate a conflict, i.e., for all subsets denoted by $\overline{\Gamma}'_u \subseteq \overline{\Gamma}_u$ and $\Gamma'_y \subseteq \Gamma_y$ the following are equivalent for an irreducible certificate:*

1. $\overline{\Gamma}'_u$ *and* Γ'_y *generate a conflict.*

2. $\overline{\Gamma}'_u = \overline{\Gamma}_u$ *and* $\Gamma'_y = \Gamma_y$.

One cannot prune irreducible certificates and each element is necessary for the set to remain a counter-example. Let Δ^{SAT} be the elements (consisting of inputs $\overline{\Gamma}_u^{\text{SAT}}$ and outputs Γ_y^{SAT}) of the naive certificate. For ease of exposition we slightly abuse notation to denote $\text{TEST}(\Gamma_u^{\text{SAT}}, \Gamma_y^{\text{SAT}})$ by $\text{TEST}(\Delta^{\text{SAT}})$. We denote the output of this algorithm by Δ_{cert} which consists of inputs $\overline{\Gamma}_u^{\text{cert}}$ and outputs Γ_y^{cert}.

This method consists of an exploration phase in which it finds an element (input or output) that belongs to an irreducible certificate. Let us denote an enumeration of Δ^{SAT} by $e_1, \cdots, e_k$, and the internal state of the algorithm by $\Delta_{\text{temp}} \leftarrow \emptyset$. This method begins by adding step-by-step elements of Δ^{SAT} to Δ_{temp}. The first element ($e_i \in \Delta^{\text{SAT}}$) that fails $\text{TEST}(\Delta_{\text{temp}})$ is part of an irreducible certificate, and therefore is added to Δ_{cert}.

In order to find further elements of this certificate, we keep e_i in the background and repeat the procedure by adding elements one-by-one. The first element that fails the consistency check is added to Δ_{cert}. We continue until $\text{TEST}(\Delta_{\text{cert}})$ returns false. It is clear that Δ_{cert} is an irreducible certificate based on the construction. This repeated process can be implemented efficiently by using the divide and conquer paradigm as depicted in Algorithm 4. When an element e_i of Δ^{SAT} is detected we divide the the remaining elements into two disjoint subsets $\Delta^1 := \{e_1, \cdots, e_j\}$ and $\Delta^2 := \{e_{j+1}, \cdots, e_{i-1}\}$. We can now recursively apply the algorithm to find a conflict Δ^2_{cert} among Δ^2 by keeping the set Δ^1 in the background and a conflict Δ^1_{cert} among Δ^1 by keeping the set Δ^2_{cert} in the background. This method of finding an irreducible subset is depicted in Algorithm 4.

Note that the resulting counter-example depends on the initial enumeration of elements in Δ^{SAT}. If the all the inputs (outputs) are ahead of outputs (inputs), then the resulting counter-example mostly consists of inputs (outputs). In order to have the maximal reduction in the search space of the SAT solver at each iteration, we produce three certificate using this method, putting inputs first, outputs first and mixing both inputs and outputs.

In the last part of this section, we look at the certificate size for a generic LTI system. We observe that the certificate size cannot be smaller that the number of inputs which is stated formally in the following lemma.

Algorithm 4: T-solver.QuickXplain

input : $\Delta^0_{\text{cert}}, \Delta^0$;

if $\text{T}-\text{solver.check}(\Delta^0_{cert}) == \text{UNSAT}$ ***or*** $\Delta^0 == \emptyset$ **then**

  return $\emptyset$;

end

Let $e_1, \cdots, e_k$ be an enumeration of Δ^0 ;

$i \leftarrow 0$, $\Delta_{\text{temp}} \leftarrow \Delta^0_{\text{cert}}$, ;

while $\text{T}-\text{solver.check}(\Delta_{temp}) == \text{SAT}$ *and* $i \leq k$ **do**

  $i \leftarrow i + 1$;

  $\Delta_{\text{temp}} \leftarrow \Delta_{\text{temp}} \cup e_i$;

end

$\Delta_{\text{cert}} \leftarrow e_i$, $j \leftarrow \lfloor \frac{i}{2} \rfloor$;

$\Delta^1 \leftarrow \{e_1, \cdots, e_j\}$;

$\Delta^2 \leftarrow \{e_{j+1}, \cdots, e_{i-1}\}$;

$\Delta_{\text{cert}} \leftarrow \Delta_{\text{cert}} \cup$ T-solver.QuickXplain$(\Delta^1 \cup \Delta_{\text{cert}}, \Delta^2)$;

$\Delta_{\text{cert}} \leftarrow \Delta_{\text{cert}} \cup$ T-solver.QuickXplain$(\Delta_{\text{cert}}, \Delta^1)$;

return Δ_{cert} ;

Lemma 2.3. *For a generic LTI system the size of the certificate is always lower bounded by* $m + 1$*, where* m *is the number of inputs.*

Proof. See Appendix A. □

Algorithm 5: T-solver.Certificate 2

input : $\Gamma_u^{\mathrm{SAT}}, \Gamma_y^{\mathrm{SAT}}$;

$\Delta_{\mathrm{cert}} \leftarrow$ T-solver.QuickXplain$(\emptyset, \overline{\Gamma}_u^{\mathrm{SAT}} \cup \Gamma_y^{\mathrm{SAT}})$;

Divide Δ_{cert} to inputs $\overline{\Gamma}_u^{\mathrm{cert}}$ and outputs Γ_y^{cert} ;

$\Phi^2_{\mathrm{cert}} \leftarrow \sum_{j \in \overline{\Gamma}_u^{\mathrm{cert}}} \mathbf{b}_j + \sum_{i \in \Gamma_y^{\mathrm{cert}}} \mathbf{c}_i \geq 1$;

return Φ^2_{cert} ;

2.5 Simulation Results

We implemented our SMT-based estimator in MATLAB while interfacing with the SAT solver SAT4J [LBP10] and assessed its performance in two case studies, randomly generated LTI systems and a chemical plant. We report the overall running time by using the two proposed methods, Algorithm 3 and Algorithm 5.

2.5.1 Random Systems

We randomly generate systems with a fixed state dimension ($n = 40$) and increase the number of inputs and outputs. Each system is generated by drawing entries of (A, B, C, D) according to uniform distribution, when necessary we scale A to ensure that the spectral radius is close to one. In each experiment, twenty percent of inputs and outputs are under adversarial attacks, and we generate the support set for the adversarial signals uniformly at random. Attack signals and the initial states are drawn according to independent and normally distributed random variables with zero mean and unit variance. All the systems under experiment satisfy a suitable sparse strong observability condition as described in Section 2.3.

Figures 2.3 and 2.4 report the results of the simulations, each point represents the average of 20 experiments. All the experiments run on an Intel Core i5 2.7GHz processor with 16GB of RAM. We verify the run-time improvement resulting from using the shorter certificates,

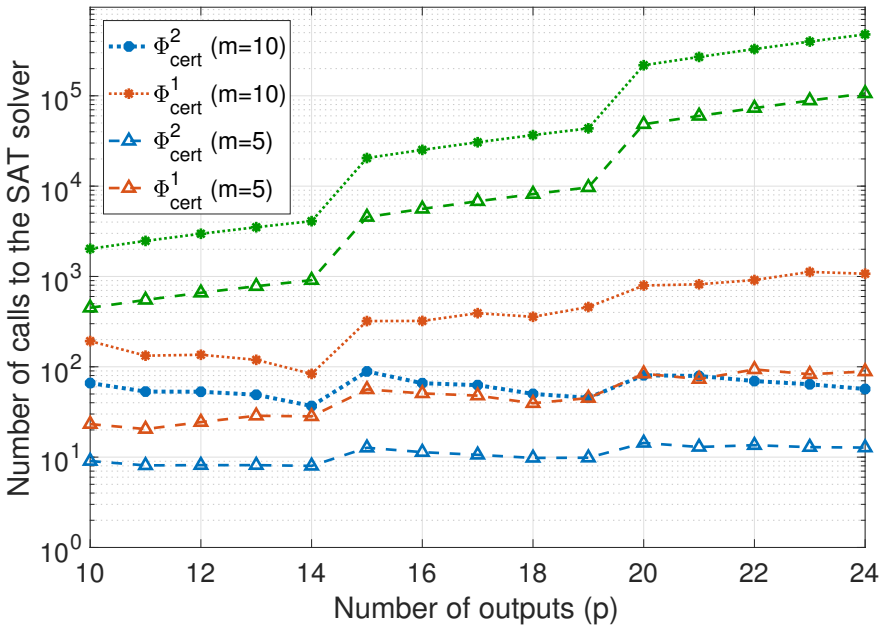


Figure 2.3: Number of calls to the SAT solver in Algorithm 1 using Φ^1_{cert}, Φ^2_{cert} vs. the number of outputs (p) for a fixed number of inputs (m). Green dotted and green dashed lines represent upper-bounds for the number of the SAT solver calls when using the naive certificate for $m = 5$ and $m = 10$, respectively.

Φ^1_{cert} and Φ^2_{cert}, compared to the theoretical upper-bound of the brute-force approach in Figure 2.3. For instance, consider the scenario with $p = 24$ and $m = 10$ in Figures 2.3 and 2.4. In the brute-force approach, we require to check all $\binom{24}{4} \times \binom{10}{2} \approx 4.8 * 10^5$ different combinations of inputs and outputs, however, by exploiting either Φ^1_{cert} or Φ^2_{cert} we observe a substantial improvement. We observe that although Φ^2_{cert} gives a worse run time for systems with smaller number of outputs, it scales better compared to Φ^1_{cert} when the number of inputs and outputs grow.

2.5.2 Chemical Plant

In this part, we use the proposed observer to detect attacks on inputs and outputs of a simplified version of the Tennessee Eastman control challenge problem [DV93]. Ricker [Ric93] derived a continuous time LTI model of the plant interaction in its steady state. This system

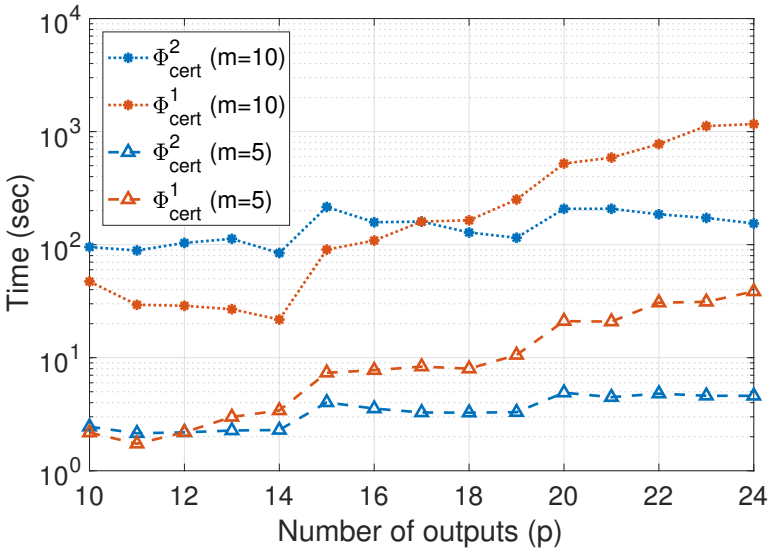


Figure 2.4: Execution time of Algorithm 1 using Φ^1_{cert}, Φ^2_{cert} vs. the number of outputs (p) for a fixed number of inputs (m).

consists of 4 control inputs and 10 measured outputs and the linearized model has 8 state variables. The structure of the continuous-time dynamics is reported below.

$$\frac{dx}{dt} = \begin{bmatrix} * & * & * & * & * & * & * & 0 \\ * & * & * & * & * & 0 & * & 0 \\ * & * & * & * & * & 0 & * & 0 \\ * & * & * & * & 0 & 0 & 0 & * \\ 0 & 0 & 0 & 0 & * & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & * & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & * & 0 \\ 0 & 0 & 0 & * & 0 & 0 & 0 & * \end{bmatrix} x + \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ * & 0 & 0 & 0 \\ 0 & * & 0 & 0 \\ 0 & 0 & * & 0 \\ 0 & 0 & 0 & * \end{bmatrix} u,$$

$$y = \begin{bmatrix} 0 & 0 & 0 & 0 & * & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & * & 0 & 0 \\ * & * & * & * & 0 & 0 & * & 0 \\ * & * & * & * & 0 & 0 & 0 & * \\ * & * & * & * & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & * & 0 & 0 & 0 & 0 \\ * & * & * & 0 & 0 & 0 & 0 & 0 \\ * & * & * & 0 & 0 & 0 & 0 & 0 \\ * & * & * & 0 & 0 & 0 & 0 & 0 \\ * & * & * & 0 & 0 & 0 & * & * \end{bmatrix} x,$$

where $*$ represents a non-zero entry[3], and $x \in \mathbb{R}^8$, $u \in \mathbb{R}^4$ and $y \in \mathbb{R}^{10}$ are state, input and output variables, respectively. The only known limitation of this LTI model is the system should operate close to its steady-state. We obtain a discrete-time model by discretizing the

[3]For the exact dynamics of the LTI model, see [Ric93].

Table 2.1: Average performance of the proposed observers.

	Overall execution time	Number of calls to the SAT solver
Φ^1_{cert}	$0.22s$	20.05
Φ^2_{cert}	$0.21s$	7.95

continuous-time model assuming a zero-order hold for the input u, with a time-step of $5s$. The attacker can read all the inputs and outputs and manipulate one control input and two measured outputs. The linearized system is $(2, 4)$-sparse strongly observable, therefore our observer can correctly reconstruct the state under this attack model.

We randomly generate attack signals and the initial state according to independent and normally distributed random variables. The support set of attacks are drawn uniformly at random, and in each experiment one input and two outputs are under adversarial attacks. The proposed observer in this paper can correctly reconstruct the (delayed) state after 8 samples, and the average performance of 20 experiments, by using Φ^1_{cert} and Φ^2_{cert} is reported in Table 2.1. The overall execution time is the run time of the observer after receiving all the required samples from the plant, and it does not take the sampling time of the plant into account. We observe that the execution time of the observer to reconstruct the state and to detect attacks is much smaller compared to the sampling time of the plant.

2.6 Conclusion

In this chapter, we considered the problem of secure state estimation when inputs and/or outputs are under adversarial attacks. In this set-up, there is no restriction on how the adversary manipulates inputs and outputs. By introducing the notion of sparse strong observability, we derived necessary and sufficient conditions under which state estimation is possible given bounds on the number of attacked outputs and inputs. Furthermore, we proposed an estimator to harness the complexity of this intrinsically combinatorial problem,

by leveraging satisfiability modulo theory solving. We demonstrated the scalability and effectiveness of the proposed estimator with numerical simulations.

CHAPTER 3

Secure System Identification

3.1 Introduction

The recent spate of publicized attacks on cyber-physical systems ranging from cars [Gre15] to infrastructure [Lan11] has led to significant research on security of cyber-physical systems, (see for example, [CAS08, SPH+10, ASH13, MKB+12] and references therein). One mechanism advocated in several recent works is to use the properties of the system dynamics to defend against sensor attacks (see for example[FTD14b, MGCS10, TSSJ15, MHS14, PDB13] and references therein). The basic assumption in these works is that the system model is accurately known to all parties.

In this chapter we ask the question of whether one can identify the system despite attacks on sensor measurements. Clearly this can be an ill-posed problem. For instance, consider the system in Figure 3.1 labeled "attack free" and its attacked version labeled "under attack". The attack consists of changing the output of the p^{th} sensor from $c_p x$ to $c'_p x$. Since the resulting system is still LTI, we cannot expect to distinguish the attacked system from the un-attacked system in the bottom of Figure 3.1 solely based on the (corrupted) measured data. Therefore, we seek to characterize the class of systems that cannot be distinguished in the presence of attacks. Moreover, we want to demonstrate a meaningful use of such an identification for a control task, *e.g.,* stabilization.

The main result in this chapter is a characterization of this *equivalence* class, for given bounds on the number of attacked sensors as well as a certain observability condition on the system (see Sections 3.2 and 3.3 for more details). We also demonstrate that identification

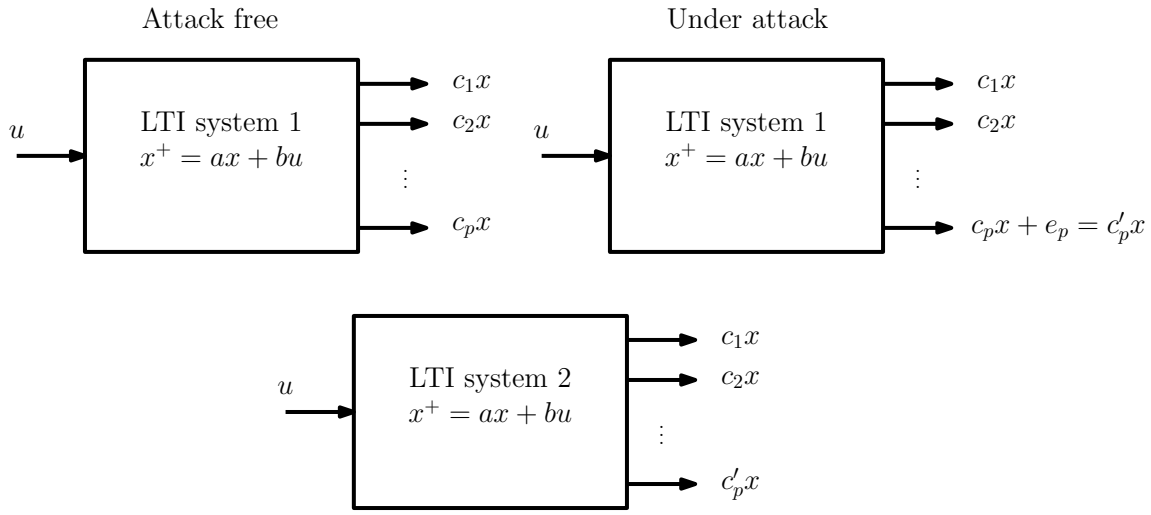


Figure 3.1: An example that illustrates the impossibility of exact system identification under adversarial attacks. Consider the system labeled "attack free" and its attacked version labeled "under attack". The attack consists in changing the output of the p^{th} sensor from c_px to c'_px. Since the resulting system is still LTI, it is impossible to distinguish under-attacked LTI system 1 from un-attacked LTI system 2 solely based on the (corrupted) measured data.

up to this class is indeed useful, as we can use it to stabilize the underlying system. These results generalize the classical results in (attack-free) system identification, where there is a characterization of such an equivalence class (of "similar state-space representations"), see for example [AM06].

Related work. Among the several different security problems reported in the literature, *e.g.,* denial-of-service [ZM14, DPT15, STDP16, GLB10], man-in-the-middle [Smi15], etc, our results are closest to the line of research on the secure state estimation problem, [FTD14b, TSSJ15, CWH15b, SCW+15, PWB+14b, MS15] and [ST16b]. The problem of secure and resilient state estimation in the presence of malicious agents has recently gained attention, [MGCS10, MHS14, PDB13, PWB+14b] and [YZF15]. Fawzi et. al. [FTD14b] considered the problem of control and estimation of LTI systems under adversarial attacks. The authors exploit the dynamics of the system for the identification of attacks. As mentioned, in this work we study the problem of identifying attacks when the plant is not known. Using

a coding theoretic approach, Fawzi [FTD14b] investigated conditions under which attack detection is possible and showed this problem to be closely related to observability under the absence of several sensors. This notion was further refined by Shoukry et. al. [ST16b] and called sparse-observability. Independently, Chong et. al. [CWH15b] investigated same problem for continuous-time LTI systems and introduced the notion of observability under attacks. Mishra et. al. [MSK$^+$17] analyzed the noisy version of this problem and identified its optimal solution for Gaussian noise. Secure state estimation for a class of non-linear plants has been explored recently *e.g.*, [TSSJ15, SCW$^+$15, YZF15] and [SNB$^+$15].

In another line of work, Tiwari et. al. [TDJ$^+$14] considered the problem of determining sensor spoofing attacks. Their proposed two-step method does not rely on the dynamics of the system. In the first step, they construct a safety envelope to be used for attack detection. This method relies on the attack-free stream of data for the first step. Our method can be applied directly to the corrupted data and does not rely on the existence of attack-free data.

This chapter is organized as follows. In Section 3.2 we give the precise formulation after establishing the notation and a brief review of the behavioural approach to system theory. We introduce the notion of "similarity modulo outputs" in this section. Section 3.3 gives the main result. The proof outlines are given in Section 3.4. In Section 3.5 we develop a computational method for our result. Section 3.6 studies the problem in the noisy scenario where in addition to the attacks, the sensor measurements are affected by additive noise. This chapter concludes with a discussion in Section 3.7.

3.2 Preliminaries and Problem Definition

Notation. We represent vectors and real numbers by lower case letters, such as u, x, y, and matrices with capital letters, such as A. For a vector $x \in \mathbb{R}^n$ and $O \subseteq \{1, \ldots, n\}$, we denote the vector obtained from x by removing all the elements except those indexed by O by $x|_O$. We denote the size of O by $|O|$. For a given vector space $\mathbb{Y} \subseteq \mathbb{R}^n$, we use the notation $\mathbb{Y}|_O = \cup_{y \in \mathbb{Y}} \{y|_O\}$. A time series is a map $\mathbf{w} : \{0, \ldots, T-1\} \to \mathbb{R}^d$ where T is the length

of time series and d is the dimension of the signal space. We represent time series by lower case bold letters, such as $\mathbf{w}$, and the restriction of $\mathbf{w}$ to the i-th component with $\mathbf{w}_i$. For a set $O \subseteq \{1, \ldots, d\}$, we define $\mathbf{w}|_O(t) := \mathbf{w}(t)|_O$. We use the terms "sequence", "time series" and "trajectory" interchangeably. We may represent time series $\mathbf{w}$ as $\{w(t)\}_{t=0}^{T-1}$. For times series $\mathbf{u}$ and $\mathbf{y}$, we represent their Cartesian product by $(\mathbf{u}, \mathbf{y})$. We denote the Hankel matrix of time series $\mathbf{u}$ by

$$\mathcal{H}_{i,j}(\mathbf{u}) := \begin{bmatrix} u(0) & u(1) & \ldots & u(j-1) \\ u(1) & u(2) & \vdots & u(j) \\ \vdots & \vdots & \vdots & \vdots \\ u(i-1) & u(i) & \ldots & u(i+j-2) \end{bmatrix}, \tag{3.1}$$

where i and j are the number of rows and columns of the Hankel matrix, respectively. We also denote a Hankel matrix by $\mathcal{H}_i(\mathbf{u})$ whenever j takes the maximal possible value $j = T - i + 1$.

3.2.1 Behavioural System Theory

In this work we use many ideas from the behavioral approach to system theory introduced by Willems, see, *e.g.*, [WP13] and [MWVHDM06]. Since all we have access to is data generated by a system, *i.e.*, behaviors, the behavioral framework provides a natural setting to investigate what can be inferred from data even in the presence of attacks. In the rest of this part, we briefly review concepts and terminology in the behavioural approach.

Definition 3.1 (Discrete-time dynamical system)**.** *A discrete-time dynamical system S is defined as a 3-tuple $S = (\mathbb{T}, \mathbb{W}, \mathcal{B})$, with $\mathbb{T} \subseteq \mathbb{N}_0$ the time axis, $\mathbb{W}$ the signal space, and $\mathcal{B} \subseteq \mathbb{W}^{\mathbb{T}}$ collection of time series called its behavior. In the context of control systems the signal space is often decomposed into input and output spaces, i.e., $\mathbb{W} := \mathbb{U} \times \mathbb{Y}$, where $\mathbb{U}$ and $\mathbb{Y}$ denote the input and the output spaces, respectively.*

In the remainder of this chapter, we refer to discrete-time dynamical systems simply as systems. We say that a system $S = (\mathbb{T}, \mathbb{W}, \mathcal{B})$ explains a time series $\mathbf{w}$ when $\mathbf{w} \in \mathcal{B}|_{[0,T]}$, where $\mathcal{B}|_{[0,T-1]}$ is the restriction of the behavior to $[0, T-1]$, i.e., $\mathcal{B}|_{[0,T-1]} := \{\Pi_{[0,T-1]}\mathbf{w} | \mathbf{w} \in \mathcal{B}\}$,

and $\Pi_{[0,T-1]} : \mathbb{W}^{\mathbb{N}_0} \to \mathbb{W}^T$ is the natural projection mapping onto the first T components. The notion of system in this definition is quite general. In this prospectus we focus on systems that are linear and time-invariant.

Definition 3.2 (LTI system)**.** *A system $S = (\mathbb{T}, \mathbb{W}, \mathcal{B})$ is linear when the signal space $\mathbb{W}$ is a vector space and $\mathcal{B}$ is a linear subspace of $\mathbb{W}^{\mathbb{T}}$. System S is time invariant if $\mathcal{B} \subseteq \sigma\mathcal{B}$, where σ is the backward shift operator on the time series $(\sigma\mathbf{w})(t) := \mathbf{w}(t+1)$ and $\sigma\mathcal{B} := \{\sigma\mathbf{w} | \mathbf{w} \in \mathcal{B}\}$. We say S is Linear Time-Invariant (LTI) if it is both linear and time-invariant.*

Consider the difference equation

$$R_0 w(t) + R_1 w(t+1) + \ldots + R_l w(t+l) = 0, \tag{3.2}$$

where $R_\tau \in \mathbb{R}^{g \times d}, \tau \in \{0, \cdots, l\}$. This difference equation (3.2) induces a dynamical system via the representation

$$\mathcal{B} = \{\mathbf{w} \in \left(\mathbb{R}^d\right)^{\mathbb{N}_0} | \quad (3.2) \quad \text{holds}\}. \tag{3.3}$$

We call (3.2) a kernel representation of the behavior (3.3). For a given behavior the kernel representation exists[1] but it is not unique, however they are all related by an equivalence relation. We define the lag of a behavior, $\mathcal{B}$ as the maximum lag of its shortest lag kernel representation[2] and denote it by $l(\mathcal{B})$. We represent the state dimension of its minimal state-space realization by $n(\mathcal{B})$.

In order to address resiliency of a system to adversarial attacks, we need to define a notion that formalizes redundancy in the outputs. Observability in the behavioral framework formalizes this concept.

Definition 3.3 (Observability)**.** *Let $(\mathbb{T}, \mathbb{W}_1 \times \mathbb{W}_2, \mathcal{B})$ be a time-invariant dynamical system. Trajectories in $\mathcal{B}$ are partitioned as $(\mathbf{w}_1, \mathbf{w}_2)$. We say $\mathbf{w}_2$ is observable from $\mathbf{w}_1$ if $(\mathbf{w}_1, \mathbf{w}_2), (\mathbf{w}_1, \mathbf{w}_2') \in \mathcal{B}$ implies $\mathbf{w}_2 = \mathbf{w}_2'$.*

[1] Under the assumption of completeness of the behaviour, see Chapter 7 in [MWVHDM06]

[2] See chapter 7 in [MWVHDM06] for the detailed explanation of characterization of the shortest lag kernel representation.

State-space observability (see Chapter 3 in [AM06]) and observability in the behavioral framework are different notions. However, in the special case when we have a minimal realization of the system $S = (\mathbb{T}, \mathbb{U} \times \mathbb{Y}_1 \times \mathbb{Y}_2, \mathcal{B})$ denoted by $(A, B, \left[C_1^T, C_2^T\right]^T, D)$, observability of y_2 from (u, y_1) essentially means that (A, C_1) is an observable pair.

We often work with different subsets of sensors and use the traditional identification algorithms for each subsystem to defend against potential attacks. We formally define this notion as a quotient system.

Definition 3.4 (Quotient system)**.** *We say $S_Q = (\mathbb{T}, \mathbb{U} \times \mathbb{Y}_Q, \mathcal{B}_Q)$ is a quotient of $S = (\mathbb{T}, \mathbb{U} \times \mathbb{Y}, \mathcal{B})$ if both have the same input space and there exists a linear projection denoted by $\Pi : \mathbb{U} \times \mathbb{Y} \to \mathbb{U} \times \mathbb{Y}_Q$ such that $\mathcal{B}_Q = \Pi\mathcal{B} := \{\mathbf{w} | \exists \mathbf{w}_0 \in \mathcal{B}$ s.t. $w(t) = \Pi w_0(t), \forall t \in \mathbb{T}\}$.*

When $\mathbb{Y} \subseteq \mathbb{R}^p$, $O \subseteq \{1, \ldots, p\}$, and Π is the natural projection mapping from $\mathbb{U} \times \mathbb{Y}$ to $\mathbb{U} \times \mathbb{Y}|_O$, we represent this specific quotient subsystem $(\mathbb{T}, \mathbb{U} \times \mathbb{Y}|_O, \Pi\mathcal{B})$ by $S|_O$. We say a map $\Pi : \mathbb{M} \to \mathbb{N}$ is a linear projection if it is linear and surjective.

3.2.2 Preliminaries

In the rest of this section, we develop the machinery to introduce the notion of "similarity modulo outputs" as well as "s-sparse observability" and "Hamming distance". These notions are required to state and understand the main results.

We define the notion of s-sparse observability in the behavioral setting by adapting the state-space notion introduced in [ST16b].

Definition 3.5 (s-sparse observability)**.** *System S is s-sparse observable if any s outputs are observable from the input and the remaining outputs.*

Given any minimal state-space realization of the system, this definition is equivalent to Definition 3.1 in [ST16b].

Proposition 3.1. *System S is s-sparse observable if for any minimal realization (A, B, C, D),*

$(A, B, C|_O, D|_O)$ *is observable for any subset* O *of indices with* $|O| = p - s$*, where* p *is the number of outputs.*

Proof. By definition of the observability. □

Following concept will be used to characterize identifiability subject to attacks.

Definition 3.6 (Parallel composition)**.** *Consider systems* $S_i = (\mathbb{T}, \mathbb{U} \times \mathbb{Y}_i, \mathcal{B}_i)$ *for* $i \in \{1, 2\}$ *with the same input space. The parallel composition of* S_1 *with* S_2 *is the system* $(\mathbb{T}, \mathbb{U} \times \mathbb{Y}_1 \times \mathbb{Y}_2, \mathcal{B})$*, where* $\mathcal{B}$ *is defined by*

$$\{(\mathbf{u}, \mathbf{y}_1, \mathbf{y}_2) \in \left(\mathbb{U} \times \mathbb{Y}_1 \times \mathbb{Y}_2\right)^{\mathbb{T}} | (\mathbf{u}, \mathbf{y}_1) \in \mathcal{B}_1, (\mathbf{u}, \mathbf{y}_2) \in \mathcal{B}_2\} \tag{3.4}$$

Now we are ready to introduce the notion of similarity modulo outputs which is a core concept in secure system identification.

Definition 3.7 (Similar modulo outputs)**.** *Two LTI systems,* S_1 *and* S_2*, with the same input spaces are called similar modulo outputs, if both of them have the same order* n*, and there exists an* n*-dimensional subspace which is invariant under the dynamics of the parallel composition of* S_1 *with* S_2*. Throughout this chapter, we denote this relation by* $\sim$*.*

Proposition 3.2. *Similarity modulo outputs is an equivalence relation and divides* $\mathcal{L}_m^{m+p,n}$ *into equivalence classes.*

Proof. See Appendix B. □

We now define the Hamming distance between two time series similarly to classical coding theory. We can think of each time series, such as $\mathbf{w}$, as a code and its components $\mathbf{w}_i$ for $i \in \{1, \ldots, d\}$ as symbols.

Definition 3.8 (Hamming distance)**.** *For two time series* $\mathbf{y}$ *and* $\mathbf{z}$ *the Hamming distance between* $\mathbf{y}$ *and* $\mathbf{z}$ *is the maximum number of indices,* i*, such that* $\mathbf{y}_i \neq \mathbf{z}_i$*.*

Note that $\mathbf{y}_i$ is a time series. Hence, the equality $\mathbf{y}_i = \mathbf{z}_i$ is to be understood as $y_i(t) = z_i(t)$ for all $t \in \{0, \ldots, T-1\}$.

3.2.3 Problem Definition

We consider the problem of system identification of LTI systems when sensors measurements are subject to adversarial attacks. The adversary is omniscient and can arbitrarily alter sensor measurements. We impose no assumptions on the signals injected by the adversary. However, we assume that an upper bound on the number of attacked sensors is known.

Assumption 3.1 (Bound on the number of attacked sensors)**.** *We assume that an upper bound s on the number of attacked sensors is given.*

The following assumption is required, otherwise even the identification of the system in the absence of attacks becomes an ill-posed problem.

Assumption 3.2 (Identifiablity given the input sequence)**.** *The behavior of the underlying system is identifiable from the input sequence*[3]*.*

We are ready to precisely state the problem we study. The underlying LTI system is denoted by $S = (\mathbb{N}_0, \mathbb{U} \times \mathbb{Y}, \mathcal{B})$ with $\mathbb{U} = \mathbb{R}^m$ and $\mathbb{Y} = \mathbb{R}^p$. System S is identifiable given the input sequence and upper bounds on the lag and order of its behavior are given by $l_{\max}$ and $n_{\max}$, respectively. The available data is the time series $(\mathbf{u}, \mathbf{y})$, where $\mathbf{u}$ is the input sequence and $\mathbf{y}$ is the corrupted output sequence.

The sensor measurements are given by, $\mathbf{y} = \mathbf{y}_S + \mathbf{y}_{\text{attack}}$, where $\mathbf{y}_{\text{attack}}$ is the signals injected by the adversary, which can attack a set $K \subseteq \{1, \ldots, p\}$ with $|K| \leq s$, *i.e.*, $\mathbf{y}_{\text{attack}}|_{\{1,\cdots,p\}\setminus K} = 0$. We do not impose any further restrictions on $\mathbf{y}_{\text{attack}}$, and $\mathbf{y}_{\text{attack}}|_K$ can be any arbitrary sequence.

Problem statement: Given the sequence $(\mathbf{u}, \mathbf{y})$, we seek answers to the following problems:

1. Identify a model that explains the input-output behavior of the unattacked sensor measurements $(\mathbf{u}, \mathbf{y}|_{\{1,\ldots,p\}\setminus K})$. Note that such a model is not unique.

[3]See Appendix B Proposition B.1 and the explanation followed for such conditions

2. Characterize the equivalence class of models that can explain this sequence.

3. Stabilize the true underlying system using the identified model.

3.3 Main Result

In this section, we present our main theoretical results followed by implications and explanations. In Section 3.4 we prove our results using tools we developed.

Theorem 3.1. *The Hamming distance between output trajectories of two* $2s$*-sparse observable systems is at least* $2s + 1$*, provided that*

1. *Systems are not similar modulo outputs.*

2. *The same input sequence excites both systems.*

3. *The input sequence is sufficiently rich to identify the systems.*

Remark 3.1. *Theorem 3.1 essentially states that the output trajectories of* $2s$*-sparse observable systems that are not similar modulo outputs are distinguishable despite attacks on* s *sensors. In the introduction, we argued that one can only identify the system up to an equivalence class, using the corrupted data. Theorem 3.1 states that under a* $2s$*-sparse observability assumption, it is possible to find a model which is closely related to the underlying system via similarity modulo outputs relation.*

Now we are ready to present our main contribution, we show that under a $2s$-sparse observability assumption, it is possible to construct a model that can be used for stabilization of system S.

Theorem 3.2. *Let us denote the system that explains* $(\mathbf{u}, \mathbf{y}|_O)$ *by* S'*, where* O *is any subset of at least* $p - s$ *sensors that* $(\mathbf{u}, \mathbf{y}|_O)$ *can be explained by an* s*-sparse observable system. System* S *is similar modulo outputs to* S' *and any controller that stabilizes* S' *also stabilizes* S*, provided that* S *is an* $2s$*-sparse observable system and Assumptions 3.1 and 3.2 hold.*

Remark 3.2. *Note that this set, O may contain some of the under-attack sensors, however, they are* ineffective *in misleading us and we still have the same guarantee. We can further consider other subsets that result in an s-sparse observable system and take the union of all such subsets for identification. Clearly one subset corresponds to $p - s$ attack-free sensors which satisfies the test,* i.e.*, our method captures all the attack-free sensors.*

3.4 Proof Outlines

The Following lemmas have been used for proving our main theorems. Lemma 3.1 gives an equivalent condition for similarity modulo output based on the specific realization of the system. Intuitively speaking, two systems are similar modulo outputs if they share the same internal dynamical structure. Lemma 3.2 characterizes the relation between a system and its quotient systems for observable LTI systems.

Lemma 3.1. *Two systems S_1 and S_2 are similar modulo outputs if and only if for any minimal realizations of S_1 and S_2, denoted by (A, B, C, D) and (A', B', C', D'), respectively, there exists a linear change of coordinates, P, such that*

$$\begin{aligned} A' &= PAP^{-1}, \\ B' &= PB. \end{aligned} \tag{3.5}$$

Proof. See Appendix B. □

Lemma 3.2. *The following are equivalent for any system $S = (\mathbb{N}_0, \mathbb{U} \times \mathbb{Y}_1 \times \mathbb{Y}_2, \mathcal{B})$:*

1. *y_2 is observable from (u, y_1).*
2. *S and $S_Q = (\mathbb{N}_0, \mathbb{U} \times \mathbb{Y}_1, \mathcal{B}_Q)$ are similar modulo outputs, where $\mathcal{B}_Q = \Pi\mathcal{B}$ and Π is the natural projection mapping from $\mathbb{U} \times \mathbb{Y}_1 \times \mathbb{Y}_2$ to $\mathbb{U} \times \mathbb{Y}_1$.*

Proof. See Appendix B. □

3.4.1 Theorem 3.1

Suppose not the Hamming distance between output trajectories of S_1 and S_2 is at least $2s+1$. We show that S_1 and S_2 are similar modulo outputs and the claim follows by contradiction. There exist at least $p-2s$ sensors in each system identical output sequences. Now we show these $p-2s$ sensors to be enough to conclude that both systems are similar modulo outputs. Without loss of generality we assume these $p-2s$ sensors to be indexed from 1 to $p-2s$ in both systems, we denote the restriction of the outputs to these indices by $\mathbb{Y}_1$. Therefore the output space can be decomposed as $\mathbb{Y}_1 \times \mathbb{Y}_2$, where $\mathbb{Y}_2$ represents the space of the remaining $2s$ outputs. Consider systems $S_i = (\mathbb{T}, \mathbb{U} \times \mathbb{Y}_1 \times \mathbb{Y}_2, \mathcal{B}_i)$ for $i \in \{1,2\}$ and their corresponding quotient systems $Q_i = (\mathbb{T}, \mathbb{U} \times \mathbb{Y}_1, \mathcal{B}_i^Q)$ for $i \in \{1,2\}$, where $\mathcal{B}_i^Q = \Pi_1 \mathcal{B}_i$ and Π_1 is the projection map onto the first $p-2s$ coordinates. Lemma 3.2 implies that $S_i \sim Q_i$ for $i \in \{1,2\}$. Since the input is sufficiently rich for identification, so the quotient systems should have the same behavior, i.e., $Q_1 = Q_2$ and therefore $Q_1 \sim Q_2$. Similarity modulo outputs is an equivalence relation and it divides the set of all LTI systems into equivalence classes, therefore we conclude that $S_1 \sim Q_1 \sim Q_2 \sim S_2$.

3.4.2 Theorem 3.2

Such a set exists since at most s sensors are under attacks. Furthermore, there exists a subset of O, denoted by O_{clean} that corresponds to $p-2s$ attack-free sensors. Note that assumption 3.2 implies that $S'|_{O_{\text{clean}}} = S|_{O_{\text{clean}}}$. Clearly S' and S are both s-observable therefore $y|_{\{1,\ldots,p-s\}\setminus O_{\text{clean}}}$ are observable from $(u, y|_{O_{\text{clean}}})$ for both systems. Lemma 3.2 implies that $S' \sim S'|_{O_{\text{clean}}}$ and $S|_{O_{\text{clean}}} \sim S$, therefore $S \sim S'$.

Pick any arbitrary minimal state-space realizations of S and S' denoted by (A, B, C, D) and (A', B', C', D'), respectively. Lemma 3.1 implies that there exists a linear change of coordinates, P, such that $A' = PAP^{-1}$ and $B = PB'$. Let us denote the state sequences corresponding to these realizations by $\mathbf{x}$ and $\mathbf{x}'$, respectively. According to the definition of similarity modulo outputs and given the fact that both systems have same input se-

quences, we know that $x'(t) = Px(t)$. The controller makes S' asymptotically stable, i.e., $\lim_{t\to\infty} \|x'(t)\| = 0$. Note that $\|x(t)\| \leq \|P^{-1}\|\|x'(t)\|$, so $\lim_{t\to\infty} \|x(t)\| = 0$. We conclude that the same control input makes S asymptotically stable.

3.5 Implementation

In this section we analyze the computational part of our method. In the first part of this section, we develop an algorithm for (deterministic) LTI systems and we illustrate the effectiveness of it by simulation results.

Recall from Section 3.3 that the main idea is to find a subset of sensors O such that $(\mathbf{u}, \mathbf{y}|_O)$ can be explained by an s-sparse observable system. An straightforward approach is to construct a realization of this model, such as the state-space realization, and check whether it is an s-sparse observable system. This approach is summarized in Algorithm 6. Instead, we proceed by directly checking the behavioral definition of observability. Let us consider an LTI system $(\mathbb{N}_0, \mathbb{U} \times \mathbb{Y}_1 \times \mathbb{Y}_2, \mathcal{B})$, we aim to check if $\mathbf{y}_2$ is observable from $(\mathbf{u}, \mathbf{y}_1)$. According to Definition 3.3, we need to check the existence of a map from $(\mathbb{U} \times \mathbb{Y}_1)^{\mathbb{N}_0}$ to $\mathbb{Y}_2^{\mathbb{N}_0}$ that maps $(\mathbf{u}, \mathbf{y}_1)$ to $\mathbf{y}_2$. Since the underlying system is LTI, this map should be linear and causal. We know the lag of the underlying behavior is upper bounded by $l_{\max}$, therefore we need to check the existence of linear mappings $\mathcal{L}_u : \mathbb{U}^{l_{\max}+1} \to \mathbb{Y}_2$ and $\mathcal{L}_{y_1} : \mathbb{Y}_1^{l_{\max}+1} \to \mathbb{Y}_2$ such that:

$$y_2(t) = \mathcal{L}_u(u(t), \ldots, u(t - l_{\max})) + \mathcal{L}_{y_1}(y_2(t), \ldots, y_2(t - l_{\max})), \quad \forall t \in \{l_{\max}, \ldots, T-1\}, \tag{3.6}$$

where T is the length of time series.

Lemma 3.3. *Linear mappings $\mathcal{L}_u : \mathbb{U}^{l_{\max}+1} \to \mathbb{Y}_2$ and $\mathcal{L}_{y_1} : \mathbb{Y}_1^{l_{\max}+1} :\to \mathbb{Y}_2$ satisfying (3.6) exist if and only if $[y_2(l_{\max}), \ldots, y_2(T-1)]$ lies in the row space of $\mathcal{H}_{l_{\max}+1}(\mathbf{u}, \mathbf{y}_1)$.*

Proof. See Appendix B. □

Algorithm 7 summarizes the proposed method, in which for each subset we check the condition in Lemma 3.3. This algorithm returns a subset of outputs, O that we can use to identify a system using any off-the-shelf identification algorithms.

Algorithm 6: The secure identification algorithm.

```
/* Let {O_i, i = 1, ..., i_max} be an enumeration of subsets of size p − s of
   {1, ..., p}                                                              */
```

input : $\{u(t)\}_{t=0}^{T-1}$, $\{y(t)\}_{t=0}^{T-1}$, Identify() (a system identification algorithm);

output: S';

$S' \leftarrow \emptyset$;

for $i = 1, \ldots, i_{\max}$ **do**

  $S_{\text{temp}} \leftarrow \text{Identify}\Big(\{u(t)\}_{t=0}^{T-1}, \{y(t)|_{O_i}\}_{t=0}^{T-1}\Big)$;

  if S_{temp} *is s-sparse observable* **then**

    $S' \leftarrow S_{\text{temp}}$;

    break;

  end

  $i \leftarrow i + 1$;

end

return S'

3.5.1 Simulation

In this section, we illustrate the effectiveness of our algorithm with a numerically simulated example. We consider a simple physical model of a locomotive that pulls a car, the connection is modeled by a spring in parallel with a damper. The dynamics of this system can be described by the following differential equations,

$$\frac{d}{dt}\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 0 & 1 & -1 \\ -\frac{k}{m_1} & -\frac{b+c}{m_1} & \frac{b}{m_1} \\ \frac{k}{m_2} & \frac{b}{m_2} & -\frac{b}{m_2} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{m_1} \\ 0 \end{bmatrix} v, \tag{3.7}$$

Algorithm 7: The secure system identification algorithm for *deterministic* LTI systems.

/* Let $\{O_i, i = 1, \ldots, i_{\max}\}$ be an enumeration of subsets of size $p - s$ of $\{1, \ldots, p\}$ */

/* Let $\{O_{i,j}, j = 1, \ldots, j_{\max}\}$ be an enumeration of subsets of size $p - s$ of O_i */

input : $\{u(t)\}_{t=0}^{T-1}$, $\{y(t)\}_{t=0}^{T-1}$, $n_{\max}$, $l_{\max}$, s ;

output: $O \subseteq \{1, 2, \ldots, p\}$;

$O \leftarrow \emptyset$;
for $i \leftarrow 1$ ***to*** $i_{\max}$ **do**
    flag $\leftarrow \operatorname{rank}\Big(\mathcal{H}_{l_{\max}+1}\big((\mathbf{u}, \mathbf{y}|_{O_i})\big)\Big)$;
    j $\leftarrow 1$;
    while flag == $\operatorname{rank}\Big(\mathcal{H}_{l_{\max}+1}\big((\mathbf{u}, \mathbf{y}|_{O'_{i,j}})\big)\Big)$ **do**
        if $j = j_{\max}$ **then**
            $O \leftarrow O_i \cup O$;
            break;
        end
        increment j;
    end
end
return O

Table 3.1: Rank of Hankel matrices corresponding to different subsets of outputs.

Subset of Outputs	$\{1\}$	$\{2\}$	$\{3\}$	$\{1,2\}$	$\{1,3\}$	$\{2,3\}$	$\{1,2,3\}$
Rank of Hankel matrix	12	9	9	15	15	9	15

where x_1 is the distance between the locomotive and the car, x_2 and x_3 are the velocities of the locomotive and the car, respectively, $c = 160Ns/m$ represents the aerodynamic friction coefficient. Damping and restitution coefficients of the spring are represented by $b = 1100Ns/m$ and $k = 7874N/m$. The parameters $m_1 = 176580Kg$ and $m_2 = 100698Kg$ denote the mass of the locomotive and the mass of the car, respectively. The input to the system is the force applied by the locomotive and is denoted by v.

The car is equipped with a radar that reports its distance from the locomotive, and two encoders that measure the velocities of the car and the locomotive. Therefore the output can be written as:

$$y = \begin{bmatrix} y_{\text{GPS}} \\ y_{\text{ENC1}} \\ y_{\text{ENC2}} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}. \tag{3.8}$$

We obtain a discrete-time model by discretizing the continuous-time model assuming zero-order hold for the input v, with the time-step $0.1s$.

For testing the proposed method, the input sequence has been generated randomly, it satisfies Assumption 3.2 (persistency of excitation), and the sensors are assumed to be noiseless. It is straightforward to verify the system is 2-sparse observable. In this example, the adversary injects signals into the first sensor, *i.e.*, $y_1' = y_1 + e_1$. We run Algorithm 7 for this corrupted time series with $l_{\max} = 6$ and $n_{\max} = 10$. Rank of Hankel matrices for different groups are given in Table 3.1. We observe that $\{2,3\}$ is the only subset that satisfies the test in algorithm 7 while for other subsets the the rank of the Hankel matrices are different.

3.6 Extension to Noisy Measurements

In this section, we study the noisy scenario in which the sensor measurements, in addition to adversarial attacks, are affected by additive noise. It is worth mentioning that Algorithm 6 can resist against attacks under the Assumptions 3.1 and 3.2 and $2s$-sparse observability for any class of systems, not only LTI systems. However, Assumption 3.2 and the notion of s-sparse observability are only well-understood for exact deterministic LTI systems. We would like to emphasize the "deterministic LTI system" terminology to differentiate it from the case where the sensor measurements are not exact due to *e.g.*, the measurement noise. In this scenario, the output of any off-the-shelf system identification tools is probabilistic and we need to develop a machinery to address challenges arising from that.

Assume System S can be described by the following equations:

$$\begin{aligned} x(t+1) &= Ax(t) + Bu(t), \\ y_S(t) &= Cx(t) + Du(t) + \epsilon(t), \end{aligned} \tag{3.9}$$

the random variable $\epsilon(t)$ represents additive noise, which is assumed to be Independent and Identically Distributed (i.i.d.) with zero mean. When the sensor measurements are corrupted by additive noise, (attack-free) identification task becomes more challenging and requires further consideration in order to remove the effect of noise. We would like to emphasize that Assumption 3.2 cannot be satisfied for this scenario and instead we consider the following assumption.

Assumption 3.3 (Identifiablity given the input sequence)**.** *The underlying system is asymptotically identifiable from the input sequence. Formally speaking, the identified model converges to the true model with probability approaching one as the number of measurements tends to infinity.*

In the literature of system identification, there exist several methods that guarantee exact identification when the length of the training sequence tends to infinity (see for example [VODM12, Lju87]). Subspace identification algorithms are one of the most prominent such

methods. It is known that under mild conditions on the input sequence and an upper-bound on the order of system, it is still possible to identify the system asymptotically, see Appendix C and Theorem C.1 therein. Our approach does not rely on a specific identification algorithm, instead we transform the secure system identification problem to a problem that can be solved with any of the off-the-shelf identification algorithms. In the rest of this section, we develop a machinery to tackle the probabilistic setup. We show that even for the noisy scenario, Algorithm 6 could identify a useful model.

As we elaborated in the introduction, two LTI systems that are similar to each other will have the exact same input-output characteristics, and in the context of system identification they are equivalent. We use the notation $\mathcal{L}$ to denote the set of equivalence classes of this relation.[4] In the rest of this section, with a slight abuse of notation we use S as its corresponding equivalence class of similarity transformation. We reserve $[S]$ for the equivalence class of similarity-modulo outputs. In order to formalize our convergence argument, we define the notion of ϵ-similarity modulo outputs.

Definition 3.9 (ϵ-similarity modulo outputs)**.** *Two LTI systems S_1 and S_2 are ϵ-similar modulo outputs if $d([S_1],[S_2]) < \epsilon$, where d is the metric on the space of equivalence classes of similarity modulo outputs that makes the quotient map $(\mathcal{L}, d_0) \to (\mathcal{L}/\sim, d)$ continuous, and d_0 is the metric[5] on the space of equivalence classes of similarity.*[6]

Theorem 3.3. *Let us denote the output of Algorithm 6 by S'. For any $\epsilon > 0$ the probability that S' is not ϵ-similar modulo outputs to the underlying LTI system S, approaches zero when the number of data points tends to infinity, provided that S is $2s$-sparse observable and*

[4]Similar systems are similar modulo outputs, i.e., similarity modulo outputs induces an equivalence relation on $\mathcal{L}$, with a slight abuse of notation we also denote this relation by $\sim$.

[5]*Note that d_0 should make the quotient map $(\mathbb{L}, d_S) \to (\mathcal{L}, d_0)$ continuous, where $\mathbb{L}$ is the set of LTI systems (matrices (A, B, C, D)) equipped with the metric d_S.*

[6]*These metrics exist. One can always endow the quotient space of a metric space with a (pseudo)metric. The only difference between pseudometrics and metrics is topological, and for T_0 topological spaces, a pseudometric is also a metric [How12]. In this case, with a properly defined distance d_S on $\mathbb{L}$, quotient spaces ($\mathcal{L}$ and $\mathcal{L}/\sim$) are T_0, therefore d and d_0 can be constructed.*

Assumptions 3.1 and 3.3 hold. Furthermore any feedback controller that stabilizes S' also stabilizes S with probability approaching one.

Proof. First we argue that the probability of $S' = \emptyset$ converges to zero as the number of measurements, T goes to infinity. Recall that there always exists a set $O \subseteq \{1, \dots, p\}$ with $|O| = p - s$ that $\{(u(t), y(t)|_O)\}_{t=0}^{T-1}$ can be explained by an s-sparse observable system since at most s sensors are under attack and S is $2s$-sparse observable. By Assumption 3.3 the identify() subroutine guarantees the exact model when the number of measurements tends to infinity. Note that s-sparse observability is a robust property, *i.e.*, an s-sparse observable system remains s-sparse observable by a small enough perturbation of the system dynamics. Therefore, with probability approaching one the identified model for this subset O is s-sparse observable. We conclude that Algorithm 6 does not return $\emptyset$ with high probability. However, such a subset and model are not unique.

Let us assume that the output of Algorithm 6 is not $\emptyset$, and we denote the corresponding subset by O'. We prove S' is ϵ-similar modulo outputs to S with high probability. Note that S' is not necessarily the true model of the corresponding subset of sensors, s sensors are under attack therefore there exists a subset of O denoted by O_{clean} that corresponds to $p - 2s$ attack-free sensors. Assumption 3.3 implies that for any $\epsilon' > 0$ we have

$$\lim_{T \to \infty} \Pr(d_0(S'|_{O_{\text{clean}}}, S|_{O_{\text{clean}}}) < \epsilon') = 1, \tag{3.10}$$

where d_0 is the metric on the space of equivalence classes of similarity transformation.

The rest of the argument consists of two steps. The quotient map $S \mapsto [S]$ is continuous, and (3.10) holds for any $\epsilon' > 0$. Putting this together with 3.10,

$$\lim_{T \to \infty} \Pr(d([S'|_{O_{\text{clean}}}], [S|_{O_{\text{clean}}}]) < \epsilon) = 1, \tag{3.11}$$

where d is the metric on the space of equivalence classes of similarity modulo outputs that makes the map $S \mapsto [S]$ continuous.

Following the same lines of the proof of Theorem 3.2, $S \sim S|_{O_{\text{clean}}}$ and $S' \sim S'|_{O_{\text{clean}}}$. Similarity modulo outputs is an equivalence relation, hence $S' \sim S$. Together with (3.11) we

conclude

$$\lim_{T\to\infty} \Pr(d([S'],[S]) < \epsilon) = 1, \tag{3.12}$$

or equivalently S' is ϵ-similar modulo outputs to S with probability approaching one.

We showed that with high probability the identified model is ϵ-similar modulo outputs to S. Now we are ready to prove the stability result. It is a standard result in control theory that a small enough perturbation on system dynamics does not affect the stabilization for feedback controllers. Therefore any feedback controller that stabilizes S' also stabilizes $S|_{\mathrm{O}'}$ with probability approaching one. Following the argument in the proof of Theorem 3.2 we conclude that the same feedback controller makes S asymptotically stable with high probability. □

3.7 Conclusion

We considered the problem of system identification of LTI systems under adversarial attacks. We imposed no restriction on the sensors attacked by the adversary. Given a bound on the number of attacked sensors, and under a suitable sparse-observability assumption, we showed that it is possible to construct a meaningful model that enables stabilization of the unknown system. Although such a model is not unique we provided a precise characterization of which models can be distinguished by sensor measurements under attack. We defined the notion of similarity modulo outputs and showed that all of such models are similar modulo outputs. This generalizes the ideas of equivalent systems in classical linear systems theory to the case when there are sensor attacks. Moreover, we showed that the secure identification algorithm produces a useful model even in a noisy scenario in which sensor measurements are corrupted by additive noise in addition to the adversarial attack.

CHAPTER 4

Private Linear Regression

4.1 Introduction

High-complexity models are needed to solve modern learning problems, which require large amounts of data to achieve low generalization error. However, acquiring such data from users directly compromises the user privacy. Training useful machine learning models without compromising user privacy is an important and challenging research problem. One natural way to tackle this problem is to keep the data itself private, and reveal only a processed, noisy version of the data to the training procedure. Ideally, such processing would completely hide the content of the data samples, while still providing useful information to the training objective. In this work, we analyze the privacy-utility trade-off of two such schemes for the linear regression problem: additive noise, where training is performed on the data samples with additive Gaussian noise; and random projections, where each data sample is randomly projected to a lower-dimensional subspace through Johnson-Lindenstrauss Transform (JLT) [Vem05] before adding Gaussian noise. We explore guarantees for a model that is trained on such transformed data for a given privacy constraint.

Differential Privacy (DP) is perhaps the most well-known notion for privacy [DR+14], and has been applied to a variety of domains (we refer reader to [SC13] and [DR+14] and references therein). It assumes a strong adversary which has access to all data samples except one, thereby ensuring robustness of the privacy guarantee to adversaries with side-information about the database. Moreover differential privacy makes no distributional assumption on the data.

In this work we use the recently proposed notion of Mutual Information-Differential Privacy (MI-DP) to analyze the privacy performance of the schemes. This connects to the natural information-theoretic notion of privacy, as well as enabling the use of more standard tools for analysis. Moreover it is shown in [CY16] that MI-DP directly implies (ϵ, δ)-DP.

Our contributions are as follows: First, we derive closed-form expressions on the relative objective error achievable by additive noise (Theorem 4.1) and random projection schemes (Theorem 4.2), under a privacy constraint, and show that in general random projections achieve better privacy-utility trade-off. We use results from randomized linear algebra [PW15] to prove the utility guarantees. Second, using the MI-DP measure, and using the fact that the random projection matrix is private, we make a connection between the MI-DP and SIMO channel, and show that non-coherent SIMO bounds do not give a stronger scaling guarantee than their coherent counterparts. Third, we present numerical results demonstrating the performance of the two schemes.

Related work. The works in [BST14a, CMS11, KST12] propose perturbing the objective to provide privacy guarantees on the trained model, where the training procedure is trusted and has access to the full database, and the adversary can only access the resulting trained model. In contrast, we assume that the training procedure itself may be adversarial, and is not given access to the raw data samples. In the context of linear regression and related problems, the works in [ZLW09, PW15] propose random projections to provide privacy, by showing that the mutual information between the raw and projected data samples grows sublinearly with dimensions. However, this does not necessarily translate to a formal differential privacy guarantee on the data samples. Random projection as a tool to provide differential privacy has also been considered in [KKMM13] and [KJ16]. The main difference of these works with ours is that they project each data vector individually to a lower-dimensional subspace, whereas we consider mixing samples across the database, such that the effective number of "mixed" samples is fewer than original.

In terms of motivation and techniques, the works in [BBDS12, She17] are the most closely related to ours. These works consider JLT in the context of linear regression, and prove that

it guarantees differential privacy for well-conditioned data matrices. However, no explicit guarantee on the achievable empirical risk is given. In contrast, we directly analyze the privacy-utility trade-off of additive noise and random projections, where utility is measured by the objective value achieved by the trained model under the privacy scheme, normalized by the true minimum of the objective. We also use the stronger MI-DP privacy[1], instead of the traditional (ϵ, δ)-differential privacy. We emphasize that the main novelty of our work lies in the analysis of the algorithms and the resulting theoretical guarantees, and not in the algorithms themselves.

This chapter is organized as follows. In Section 4.2 we give a brief overview on different privacy metrics followed by the precise problem formulation. Section 4.3 includes the main theoretical results of this work. The proof outlines are given in Section 4.4. Section 4.5 gives the numerical results followed by Section 4.6 that concludes this chapter.

4.2 Formulation and Background

In this paper we consider the quadratic optimization

$$\min_{\theta} g(\theta) := \min_{\theta} \|X\theta - y\|_2^2, \tag{4.1}$$

where $X \in \mathbb{R}^{n \times d}$ is the data matrix in which each row corresponds to one user and $y \in \mathbb{R}^n$ are the response variables. We denote a solution of this optimization problem as $\theta^\star$. We use $X_{i,j}$ to denote the j-th feature of the i-th user data point for $i \in \{1. \cdots, n\}, j \in \{1, \cdots, d\}$. We assume the number of data points is greater than the number of features and X is full column rank. We assume that $|X_{i,j}| \leq 1$. Throughout this paper, we use bold letters for random variables to distinguish them from deterministic quantities.

Consider a database $D^N := (D_1, \cdots, D_N)$ along with a query according to a *randomized* mechanism $q(.)$. Let D^{-i} denote the set of database entries excluding D_i.

[1]In [CY16], it is shown that for discrete alphabets, the two notions are equivalent; however MI-DP is strictly stronger for continuous alphabets.

Definition 4.1 (ϵ-mutual information-differential privacy)**.** *A randomized mechanism $q(.)$ satisfies ϵ-Mutual Information-Differential Privacy (MI-DP) if*

$$\sup_{i,P(\mathbf{D}^N)} I(\mathbf{D}_i; q(\mathbf{D}^N)|\mathbf{D}^{-i}) \leq \epsilon \quad bits, \tag{4.2}$$

where the supremum is taken over all distribution on $\mathbf{D}^N$.

We aim to preserve the privacy of each entry of X, therefore, in the context of our work, $D := (X_{1,1}, \cdots, X_{1,d}, X_{2,1}, \cdots, X_{2,d}, \cdots, X_{n,d})$.

The notion of ϵ-MI-DP is closely related to (ϵ, δ)-*differential privacy*[DRV10]. We first define the notion of neighbor in databases:

Definition 4.2 (Neighbor)**.** *Two databases D^N and $\bar{D}^N$ are called neighbor if they differ only in one entry.*

In the context of our problem, two data matrices are neighbors if they only differ in one entry. Now we are ready to define (ϵ, δ) *differential privacy.*

Definition 4.3 ((ϵ, δ)-differential privacy)**.** *A randomized mechanism $q(.)$ satisfies (ϵ, δ) differential privacy (DP) if for all neighboring databases D^N and $\bar{D}^N$ and all $S \subseteq \text{Range}(q(.))$,*

$$\Pr(q(D^N) \in S) \leq e^{\epsilon} \Pr(q(\bar{D}^N) \in S) + \delta. \tag{4.3}$$

We say $q(.)$ satisfies (δ)-DP if it satisfies $(0, \delta)$-differential privacy.

Note that neither of MI-DP nor DP impose distributional assumptions on the database and the probabilities arise completely from the randomization of the mechanism.

Proposition 4.1 (Theorem 1 in [CY16])**.** *ϵ-MI-DP is stronger than (ϵ, δ)-DP in the sense that for all $\epsilon > 0$ if a mechanism is ϵ-MI-DP, there exists ϵ', δ' such that the mechanism satisfies (ϵ', δ')-DP. We denote this relation with ϵ-MI-DP $\succeq$ (ϵ, δ)-DP. Furthermore, we have the following relation:*

$$\epsilon\text{-MI-DP} \overset{(a)}{\succeq} (\delta)\text{-DP} \overset{(b)}{\equiv} (\epsilon, \delta)\text{-DP}, \tag{4.4}$$

where $\succeq$ is interpreted as being stronger and (b) means (δ)-DP $\succeq$ (ϵ, δ)-DP and (ϵ, δ)-DP $\succeq$ (δ)-DP.

Proposition 4.2 (See Lemma 2 in [CY16])**.** *If a mechanism is ϵ-MI-DP then it also satisfies $(0, \sqrt{\frac{2}{\log(e)}\epsilon})$-DP.*

Let us denote a solution to the original problem (4.1) with $\theta^\star$. Let us denote the the cost function of the transformed problem with $\hat{g}(\theta)$ with a minimum of $\hat{\theta} \in \arg\min_\theta \hat{g}(\theta)$. We define the relative error of this transformed problem as the smallest $\eta \geq 1$ such that,

$$g(\hat{\theta}) \leq \eta g(\theta^\star). \tag{4.5}$$

In this chapter we consider the achievable relative error for linear regression given ϵ-MI-DP requirement. We analyze two methods for answering this question in Section 4.3 and find the privacy-utility trade off. In the rest of this section we define some variables that will be used in the subsequent sections.

Notation. We denote the *condition number* of X with

$$\kappa(X) := \|X\|_2 \|X^\dagger\|_2 = \frac{\sigma_{\max}(X)}{\sigma_{\min}(X)}, \tag{4.6}$$

where $X^\dagger$ is the Moore-Penrose pseudoinverse of X and $\|X\|_2$ is the spectral norm of X. We denote the ratio of l_2 norm of the projection of y onto the column space of X over the l_2 norm of the residual with:

$$r(y) := \frac{\|X\theta^\star\|_2}{\|X\theta^\star - y\|_2}. \tag{4.7}$$

where $\|X\theta^\star\|_2$ is the l_2 norm of the vector $X\theta^\star$. We define $f_i(X) := \sqrt{\sum_{j=1}^{n} \frac{1}{n}|X_{i,j}^2| - \max_j \frac{1}{n}|X_{i,j}^2|}$ for $i \in \{1. \cdots, d\}$ which can be roughly seen as the square root of the empirical second moment of the ith feature across the dataset, and $f(X) := \min_i f_i(X)$. In order to give guarantees on the privacy of the projection method the amount of additional noise is expressed in terms of $f(X)$.

4.3 Privacy-Utility Trade off

In this section we analyze two randomized mechanism in terms of utility-privacy trade off. First we investigate the naive scheme of adding noise to the data matrix, we derive the amount of noise needed to satisfy ϵ-MI-DP and we give the utility guarantees. In the second scheme, we first encode the data matrix using random projections and we add noise, if necessary, to satisfy ϵ-MI-DP requirement. We derive the utility bound for the second approach and compare it to the additive noise. Throughout this section, we fix the privacy parameter to be $\epsilon > 0$ and we derive bounds for the relative error based on that.

4.3.1 Additive Gaussian Noise

In order to satisfy privacy, we add Gaussian noise directly to the data,

$$X_{AN}(X) := X + \sigma_{AN}\mathbf{N}, \tag{4.8}$$

where $\mathbf{N} \in \mathbb{R}^{n\times d}$ with i.i.d. entries drawn from $\mathcal{N}(0,1)$ and

$$\sigma^2_{AN}(\epsilon) := \frac{1}{2^{2\epsilon}-1}, \tag{4.9}$$

is the variance of the noise.

$$\theta_{AN} := \arg\min_{\theta} \underbrace{\|X_{AN}\theta - y\|_2^2}_{g_{AN}(\theta)}, \tag{4.10}$$

Theorem 4.1 (Privacy-Utility for Additive Noise)**.** *Given a data set X and the randomized mechanism $X_{AN}(X)$ with ϵ-MI-DP constraint, with probability at least $1-2e^{-\frac{\sigma_{\max}(X)^2}{2\sigma^2_{AN}(\epsilon)}\delta^2}$ we have the following bound on the relative error of the transformed problem:*

$$\eta_{AN} \leq \left(1 + \frac{\kappa(X)(\Delta(X,\epsilon)+\delta)}{1-\kappa(X)(\Delta(X,\epsilon)+\delta)}(\kappa(X)+r(y))\right)^2, \tag{4.11}$$

if $\kappa(X)\Delta(\epsilon,X) < 1$, where $\Delta(X,\epsilon) = \frac{\sigma_{AN}(\epsilon)}{\sigma_{\max}(X)}\left(\sqrt{n}+\sqrt{d}\right)$ and $\delta > 0$ is a free parameter[2].

Note that if $\sigma^2_{\max}(X)$ scales linearly with n then $\Delta(X,\epsilon)$ converges to a constant term. Based on Proposition 4.2, additive noise also satisfies (δ)-DP.

[2] *Note that support of δ is restricted to the set where $\kappa(X)(\Delta+\delta) \leq 1$.*

4.3.2 Gaussian Random Projections

We encode the data matrix using JLT to a lower dimensional space n' and we add Gaussian noise, when necessary, to guarantee ϵ-MI-DP. We denote the encoded data by $X_{RP} \in \mathbb{R}^{n' \times d}$ and $y_{RP} \in \mathbb{R}^{n'}$:

$$X_{RP}(X) := \frac{1}{\sqrt{n'}}\mathbf{S}X + \sigma_{RP}\mathbf{N}, \quad y_{RP} := \frac{1}{\sqrt{n'}}\mathbf{S}y, \tag{4.12}$$

where $\mathbf{S} \in \mathbb{R}^{n' \times n}$ represents a random projection with i.i.d. entries drawn according to $\mathcal{N}(0,1)$ and $\mathbf{N} \in \mathbb{R}^{n' \times d}$ is the noise added to ensure the privacy with i.i.d. elements $\mathcal{N}(0,1)$, and

$$\sigma^2_{RP}(X,\epsilon) := \big(\frac{1}{2^{2\epsilon}-1} - \frac{n}{n'}f^2(X)\big)_+, \tag{4.13}$$

is the variance of the additive noise[3].

We solve the following problem to estimate the model:

$$\theta_{RP} := \arg\min_{\theta} \underbrace{\|X_{RP}\theta - y_{RP}\|_2^2}_{g_{RP}(\theta)}, \tag{4.14}$$

Theorem 4.2 (Privacy-Utility for Random Projection)**.** *Given a dataset X and the randomized mechanism $X_{RP}(X)$ with ϵ-MI-DP constraint and a projection dimension of $n' < n$, with probability at least $1 - c_1 e^{-c_2 n' \delta^2}$, we have the following bound on the relative error of the transformed problem:*

$$\eta_{RP} \leq (1+\delta)^2(1+l_1(X,\epsilon))(1+l_2(X,\epsilon))^2, \tag{4.15}$$

where $l_1(X,\epsilon) := n'\sigma^2_{RP}(X,\epsilon)\left(\max_i \frac{\sigma_i(X)}{\sigma_i^2(X)+n'\sigma^2_{RP}(X,\epsilon)}\right)^2 r^2(y)$, $l_2(X,\epsilon) := \frac{n'\sigma^2_{RP}(X,\epsilon)}{\sigma^2_{\min}(X)+n'\sigma^2_{RP}(X,\epsilon)} r(y)$, $\delta \geq \sqrt{c_0 \frac{d}{n'}}$ *is a free parameter, and c_0, c_1 and c_2 are constants.*

Corollary 4.1. *The random projection methods also satisfies (δ)-DP for $\delta := \sqrt{\frac{2}{\log(e)}\epsilon}$.*

[3]Note that our algorithm does not reveal this quantity explicitly avoiding an extra privacy leakage.

Corollary 4.2. *Note that the amount of noise added to the projected data is* $\sigma^2_{RP}(\epsilon, X) = \left(\frac{1}{2^{2\epsilon}-1} - \frac{n}{n'}f^2(X)\right)_+$. *If* f^2 *does not vanish as* n *grows and* $n' = o(n)$, *asymptotically the noise variance goes to zero,* i.e., *random projection itself guarantees the privacy. Furthermore, for a given* δ, $\eta_{RP} \leq (1+\delta)^2$ *asymptotically as two other terms in* (4.15) *vanish.*

Remark 4.1. *In the proof of Theorem 4.2 in order to derive an upper bound for* (4.2) *we make a connection to the SIMO non-coherent channel. We used the coherent SIMO bound for upper bounding this quantity. One may ask if we can get a tighter bound by using the tighter non-coherent bounds (see for example [LM03]). The known non coherent bound,*

$$C \leq \frac{n'}{2}\log(1 + \frac{1}{n'\sigma^2_{RP} + nf^2(X)}). \tag{4.16}$$

does not give any improvement. This bound (4.16) *is known to be tight for the low-SNR regime [LM03]. Therefore when* $f^2 = \Omega(n)$ *asymptotically both bounds yield the same result.*

Remark 4.2. *Putting* (4.13) *and* (4.9) *together we observe that the noise needed in the random projection method is strictly less than of the additive noise, by at most* $\frac{n}{n'}f^2(X)$.

$$\sigma^2_{RP}(X, \epsilon) = \left(\sigma^2_{AN}(X, \epsilon) - \frac{n}{n'}f^2(X)\right)_+. \tag{4.17}$$

4.4 Proof Outlines

We gives proof outlines for Theorems 4.1 and 4.2. The complete proofs are provided in Appendix D.

4.4.1 Theorem 4.1

Proof Outline. The proof consists of two steps. First we derive the minimum amount of noise needed to ensure ϵ-MI-DP for X_{AN} with respect to any feature of users, which is stated in the following lemma:

Lemma 4.1 (Privacy Guarantee for the additive noise)**.** *If* $\sigma^2_{AN} = \frac{1}{2^{2\epsilon}-1}$ *then* $X_{AN}(.)$ *is* ϵ*-MI-DP with respect to any entry of* X.

Proof. We show that (4.2) is bounded by ϵ for this choice of σ_{AN}^2 and $q(.) := X_{AN}(.)$. Due to the symmetry of the problem, we fix $\mathbf{D}_i$ to be the first feature of the first data point without loss of generality. Note that

$$I(\mathbf{X}_{1,1}; \mathbf{X}_{AN}|\mathbf{X}^{-(1,1)}) = I(\mathbf{X}_{1,1}; \mathbf{X}_{1,1} + \sigma_{AN}\mathbf{N}_{1,1}|\mathbf{X}^{-(1,1)}). \tag{4.18}$$

By expanding the mutual information:

$$\begin{aligned} I(\mathbf{X}_{1,1}; \mathbf{X}_{1,1} + \sigma_{AN}\mathbf{N}_{1,1}|\mathbf{X}^{-(1,1)}) &= h(\mathbf{X}_{1,1} + \sigma_{AN}\mathbf{N}_{1,1}|\mathbf{X}^{-(1,1)}) - h(\mathbf{X}_{1,1} + \sigma_{AN}\mathbf{N}_{1,1}|\mathbf{X}) \\ &\overset{(a)}{=} h(\mathbf{X}_{1,1} + \sigma_{AN}\mathbf{N}_{1,1}|\mathbf{X}^{-(1,1)}) - h(\sigma_{AN}\mathbf{N}_{1,1}), \end{aligned} \tag{4.19}$$

where (a) holds because the noise is independent of the data. Now we need to take the maximization over all possible distribution on $\mathbf{X}$. Note that the absolute value of each entry is bounded by 1 therefore we need to take the supremum over all distribution inside this ball. The absolute value constraint implies the second moment constraint for all distribution defined on it, therefore by using the maximum entropy bound the result follows:

$$\sup_{P(\mathbf{X})} I(\mathbf{X}_{1,1}; \mathbf{X}_{AN}|\mathbf{X}^{-(1,1)}) \leq \frac{1}{2}\log(1 + \frac{1}{\sigma_{AN}^2}) = \epsilon. \tag{4.20}$$

□

The second step bounds the relative error. We use perturbation theory in the least square setup (see Theorem 5.1 in [Wed73]) and probabilistic bounds on the maximum singular value of an i.i.d. Gaussian to derive the result [RV10]. □

4.4.2 Theorem 4.2

Proof Outline. The proof consists of two steps. First we find the variance of noise needed to add to satisfy ϵ-MI-DP that results to ϵ-MI-DP model, θ_{RP}. Following lemma characterizes the amount of noise sufficient to make the mechanism ϵ-MI-DP.

Lemma 4.2. *If* $\sigma_{RP}^2 := (\frac{1}{2^{2\epsilon}-1} - \frac{n}{n'}f^2(X))_+$ *then* $X_{RP}(.)$ *is* ϵ*-MI-DP with respect to any entry of* X*.*

Proof. We show that the conditional mutual information (4.2) is bounded by ϵ for this choice of σ_{RP}^2. Due to the symmetry of the problem, we fix D_i to be the first feature of the first data point.

$$\begin{aligned}
&\max_{P(\mathbf{X})\in\mathbb{P}} I(\mathbf{X}_{1,1};\mathbf{X}_{RP}|\mathbf{X}^{-(1,1)}) \qquad (4.21)\\
&\qquad = \max_{P(\mathbf{X}^{-(1,1)})p(\mathbf{X}_{1,1}|X^{-(1,1)})} \mathbb{E}_{\mathbf{X}^{-(1,1)}}[I(\mathbf{X}_{1,1};\mathbf{X}_{RP}|\mathbf{X}^{-(1,1)}=X^{-(1,1)})],\\
&\qquad \overset{(a)}{\leq} \max_{P(\mathbf{X}^{-(1,1)})} \mathbb{E}_{\mathbf{X}^{-(1,1)}}[\max_{P(\mathbf{X}_{1,1}|X^{-(1,1)})} I(\mathbf{X}_{1,1};\mathbf{X}_{RP}|\mathbf{X}^{-(1,1)}=X^{-(1,1)})],
\end{aligned}$$

where $\mathbb{P}$ is the set of distributions which assign non-zero measure to $\mathbf{X}$ only if the absolute value of each entry is upper bounded by 1 and $f(\mathbf{X})$ is lower bounded by the $f(.)$ evaluated for the original database, (a) follows from the Jensen's inequality and the fact that maximization over the conditional distribution is a convex function. Now we find upper bounds for the term inside of the expectation, note that columns of $\mathbf{X}_{RP}$ rather than first one does not have any term associated with $\mathbf{X}_{1,1}$ and they are conditionally independent, therefore we can write

$$\begin{aligned}
&\max_{P(\mathbf{X}_{1,1}|X^{-(1,1)})} I(\mathbf{X}_{1,1};\mathbf{X}_{RP}|\mathbf{X}^{-(1,1)}=X^{-(1,1)})\\
&\qquad = \underbrace{\max_{P(\mathbf{X}_{1,1}|X^{-(1,1)})} I(\frac{1}{\sqrt{n'}}\mathbf{S}\mathbf{X}^{(:,1)}+\sigma_{RP}\mathbf{N}^{(:,1)};\mathbf{X}_{1,1}|\mathbf{X}^{-(1,1)}=X^{-(1,1)})}_{(\star)},
\end{aligned}$$

where $X^{(:,1)}$ denotes the first column of X. We find an upper bound on $(\star)$ for a fixed set of $X^{-(1,1)}$ We observe that $(\star)$ is same as the capacity of the following non-coherent SIMO channel with Rayleigh fading and a unit power constraint:

$$z = \frac{1}{\sqrt{n'}}\mathbf{S}^{(:,1)}\mathbf{X}_{1,1} + \underbrace{\sum_{i\neq 1}\frac{1}{\sqrt{n'}}\mathbf{S}^{(:,i)}X_{i,1}+\sigma_{RP}\mathbf{N}^{(:,1)}}_{\nu}, \qquad (4.22)$$

where $z\in\mathbb{R}^{n'}$ is the first column of $\mathbf{X}_{RP}$ which we treat here as the output of the channel. Note that $X_{i,1}$ ($i\neq 1$) are treated as constants for this channel and therefore ν is effectively a zero mean i.i.d. Gaussian noise with the covariance of

$$\mathbb{E}[\nu\nu^T] = (\sigma_{RP}^2+\frac{1}{n'}\sum_{i\neq 1}(X_{i,1})^2)I_{n'} = \sigma_\nu^2 I_{n'}. \qquad (4.23)$$

Now we bound the capacity of this channel, We use the coherent upper bound for the capacity of this channel:

$$\begin{aligned}\max_{P(\mathbf{X}_{1,1})} I(\mathbf{X}_{1,1}; z) &\leq \max_{P(\mathbf{X}^1_{1,1})} I(\mathbf{X}_{1,1}; z, \mathbf{S}^{(:,i)}) \\ &\overset{(a)}{\leq} \mathbb{E}_{\mathbf{S}^{(:,i)}}[\frac{1}{2}\log(1 + \frac{\frac{1}{n'}\|\mathbf{S}^{(:,i)}\|^2}{\sigma_\nu^2})] \\ &\overset{(b)}{\leq} \frac{1}{2}\log(1 + \mathbb{E}_{\mathbf{S}^{(:,i)}}[\frac{\frac{1}{n'}\|\mathbf{S}^{(:,i)}\|^2}{\frac{n}{n'}f(X)^2 + \sigma_{RP}^2}]) \overset{(c)}{\leq} \epsilon. \end{aligned} \tag{4.24}$$

Note that the absolute value constraint implies the second moment constraint for all distribution defined on it and (a) follows from the maximum entropy bound, (b) follows directly from the Jensen's Inequality, (c) comes from the fact that the outer maximization is over distributions that assign non-zero measure to $\mathbf{X}$ only if $f(\mathbf{X}) \geq f(X)$. □

Now we derive the utility guarantee for this method. Note that by rewriting $X_{RP} = \frac{1}{\sqrt{n'}}\begin{bmatrix} S & N \end{bmatrix}\begin{bmatrix} X \\ \sqrt{n'}\sigma_{RP} I \end{bmatrix} = \tilde{S}\begin{bmatrix} X \\ \sqrt{n'}\sigma_{RP} I \end{bmatrix}$ we observe that adding direct noise to the projected data can be interpreted as the random projection of the l_2 regularized least square problem (Ridge Regression), *i.e.*,

$$\theta_{RP} = \arg\min_\theta \|X_{RP}\theta - y_{RP}\|_2^2 \tag{4.25}$$

$$= \arg\min_\theta \|\tilde{S}\underbrace{\Big(\begin{bmatrix} X \\ \sqrt{n'}\sigma_{RP} I \end{bmatrix}\theta - \begin{bmatrix} y \\ 0 \end{bmatrix}\Big)}_{\text{RR}}\|_2^2. \tag{4.26}$$

Therefore we can split the utility analysis into two parts,

1. What is the utility loss for the l_2 regularized least square?

2. What is the utility loss for the randomized sketching (JLT)?

We use the standard SVD argument to bound the Ridge Regression relative error and by following Pilanci et. al. [PW15] (see Corollary 2) we give guarantees on the sketching step. The details of the proof are provided in Appendix D. □

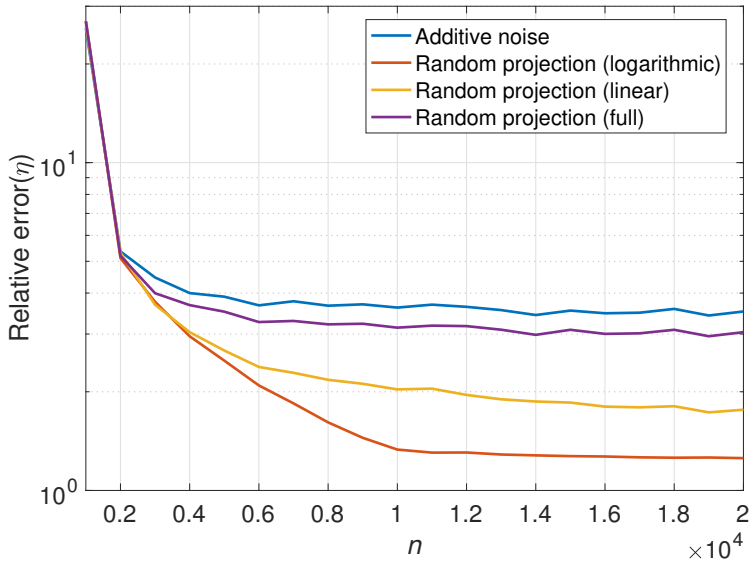


Figure 4.1: Relative error of additive noise and random projection schemes vs. the number of data points, n for $\epsilon = 0.5$ when data generated randomly and for various values of the projected dimension, n'.

4.5 Numerical Results

We numerically evaluate the relative error η achieved by the schemes in Section III subject to an ϵ-MI-DP constraint.

4.5.1 Random Data

We generate the elements $X_{i,j}$ i.i.d. uniformly in the interval $[-1, 1]$, where $X \in \mathbb{R}^{n \times 800}$, and $n = 1000k$ with $k \in \{1, 2, \ldots, 20\}$. For each case, the additive noise parameter σ_{AN} is computed according to (4.9). Similarly, the additive noise σ_{RP} is computed according to (4.13). Given k, we evaluate three choices of n': logarithmic ($n'_1 := 1000\,(\log(k) + 1)$), linear ($n'_2 := 1000\frac{k+1}{2}$), and full ($n' = n = 1000k$). The resulting relative error curves are given in Figure 4.1 for $\epsilon = 0.5$, averaged over 5 trials. We note that random projection results in a uniformly better privacy-utility trade-off compared to additive noise. Further, at this regime of ϵ, lower projection dimensions result in significantly better trade-off. Figure 4.2 plots the

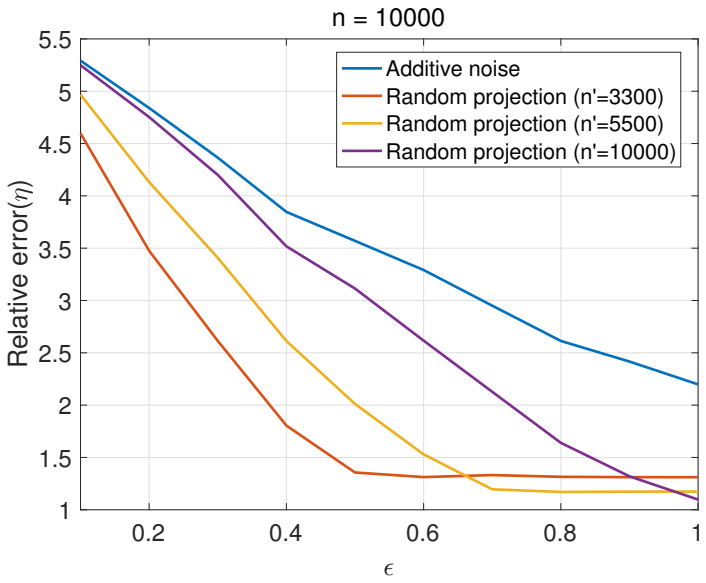


Figure 4.2: Relative error of additive noise and random projection schemes vs. ϵ, for $n = 10000$, when data generated randomly and for various values of the projected dimension, n'.

achieved relative error as a function of ϵ, for $n = 10000$. We observe that the relative error decreases linearly until it saturates for all schemes, and for stricter privacy constraints (small ϵ), lower projection dimension achieves smaller relative error. As ϵ tends higher, the privacy constraint becomes less restrictive, and schemes with higher projection dimension perform better because of the additional rows of information.

4.5.2 MNIST Handwritten Digits Dataset

We consider a reduced version of the MNIST hand-written digits dataset [LCB94], where we only take the digits 4 and 9, leading to 11791 data samples, and only consider the 300 pixels that contain the most energy across these data samples. Mapping the digit labels to $+1$ and -1, and vectorizing each data image, we solve the corresponding linear problem, which generates a model that classifies 4's versus 9's. Figure 4.3 gives the resulting test error (subject to a 80%/20% training/test set partition) for the logarithmic ($n_1' := 500\,(\log(k)+1)$), linear ($n_2' := 500\frac{k+1}{2}$), and full ($n' = n = 1000k$) random projections, as well as additive noise. To evaluate values of n smaller than 11791, we randomly sample the

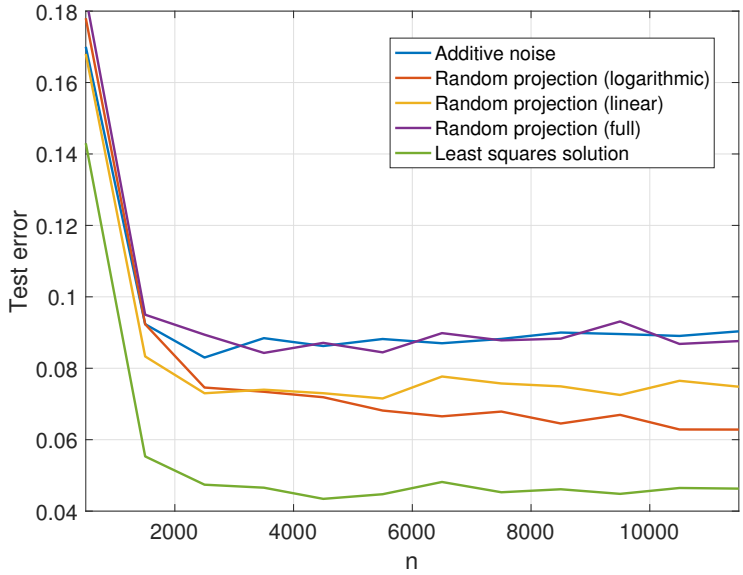


Figure 4.3: Test error of additive noise and random projection schemes for $\epsilon = 0.2$, for MNIST data set.

dataset. The results are averaged over 10 trials. Similar to the random case, we observe that random projection with logarithmic dimensions result in the best performance, while preserving MI-DP with $\epsilon = 0.2$.

4.6 Conclusion

One possible way of fulfilling the machine learning task while preserving user privacy is to train the model on a transformed, noisy version of the data, which does not reveal the data itself directly to the training procedure. In this chapter, we analyzed the privacy-utility trade-off of two such schemes for the problem of linear regression: additive noise, and random projections. In contrast to previous work, we considered a recently proposed notion of differential privacy that is based on conditional mutual information, which is stronger than the conventional (ϵ, δ)-differential privacy, and used relative objective error as the utility metric. We found that projecting the data to a lower-dimensional subspace before adding noise attains a better trade-off in general. We also made a connection between privacy problem

and (non-coherent) SIMO, which has been extensively studied in wireless communication, and use tools from there for the analysis.

CHAPTER 5

Private Distributed Optimization

5.1 Introduction

Data privacy is a central concern in statistics and machine learning when utilizing sensitive databases such as financial accounts and health-care. Thus, it is important to design machine learning algorithms which protect users' privacy while maintaining acceptable level of accuracy. In this paper, we consider a distributed framework in which N nodes aim to minimize a global cost function $f(x) = \sum_{i\in[N]} f_i(x), x \in \mathcal{X}$ where f_i is only available to the i-th node and contains sensitive information about individuals trusting this node. We study the consensus-based gradient distributed algorithm in which nodes broadcasts their local estimates and update them accordingly based on what they received from the neighbors. However, revealing the local estimate may expose privacy of individual data points and cares must be taken into account to protect the sensitive information from an adversary that oversees all messages among nodes.

Differential Privacy (DP) is a well-known notion of privacy [DR$^+$14] and found application in many domains (we refer readers to [DR$^+$14] and [SC13]). DP assumes a strong adversary that has access to all data points except one and rigorously limits inferences of an adversary about each individual, thereby ensuring robustness of the privacy guarantee to side information. Furthermore, it does not assume any distribution on the underlying data and guarantees it gives do not depend on such assumptions. In this framework, there has been a long line of work studying differentialy private machine learning algorithms, see [SC13] and references therein. Empirical Risk Minimization (ERM) plays an important role

in the supervised learning setup and our work is tied to private ERM in distributed setup.

The algorithm in this work is not new and is a small modification of the (sub)-distributed gradient descent (DGD) algorithm [NO09, YLY16] which has been analyzed before in the literature of differential privacy [HMV15]. The main novelty of our work lies in the new analysis that leads to a *stronger* convergence results. Contribution of this work is as follows:

- We determine the variance of the noise needed to ensure DP privacy in this *iterative* and *distributed* setup (Theorem 5.1) from basic calculations, instead of using composition theorems [DR$^+$14]. This approach gives a tighter bound for the noise variances and let us increase the accuracy by optimizing over the variances.

- We derive the non-ergodic convergence behavior in this setup, thereby showing that by suitably choosing the noise variance the parameter converges to a ball around the optimal point. We further characterize the radius of the ball as a function of privacy parameters, *i.e.*, ϵ and δ (Theorem 5.2).

Related work. There is long line of research devoted to differentially private ERM in the centralized set-up, in which a trusted party has access to a private and sensitive database while the adversary observes only the final end model [CMS11, BST14b]. A number of approaches exist for this set up with the convex loss function, which can be roughly classified into three categories. The first type of approaches is to inject properly scaled additive noise to the output of a non-DP algorithm which was first proposed by [CM09] for this problem and later on is extended by [ZZMW17] and [WLK$^+$17]. The second type of approaches is to perturb the objective function which is again introduced in [CM09]. The third approach delves into the first order optimization algorithms and perturbs gradients at each step to maintain the DP, [BST14b] was one of the earliest work in this domain.

In our work, there does not exist a central trusted entity and data is distributed among N nodes that motivates the use of the distributed optimization algorithms. Differentially private distributed optimization has been explored before in [HMV15], where authors considered the similar problem under ϵ-DP constraint. It is well-known that ϵ-DP is too stringent

condition and often rises to non-acceptable accuracy. The convergence bound of [HMV15] does not diminish as the privacy requirement weakens, *i.e.*, the convergence is not exact even without any privacy requirement. We emphasize that the main novelty of our work lies in the analysis of the algorithms and the resulting theoretical guarantees.

Paper organization. In section 5.2, we give a brief overview of differential privacy followed by introducing the problem. Section 5.3 introduces the main results in which we establish the condition under which the distributed algorithm is differentially private and the convergence results. In section 5.4, we give the proof outline of Theorems 5.1 and 5.2. We demonstrate our numerical experiments in Section 5.5. Section 5.6 concludes the paper.

5.2 Background and Problem Formulation

In this section, we review the notion of differential privacy and the Gaussian mechanism which are building block of our algorithm. In the second part, we give the precise problem formulation along with the overview of the algorithm which we consider in this work.

Differential Privacy. Let $D := \{d_1, \cdots, d_N\}$ be a database containing N points in the universe $\mathbb{D}$. Two databases D and D' are called neighbors when they differ in *at most* one data point, we use the notation $D' \sim D$ to denote this relation.

A randomized mechanism $\mathcal{M}$ is differentially private if evaluated on D and D' produces outputs that have similar statistical distributions. Formally speaking,

Definition 5.1 ((ϵ, δ)-Differential Privacy)**.** *A randomized mechanism $\mathcal{M}$ is (ϵ, δ)-differentially private, if for any $S \subseteq Range(\mathcal{M})$,*

$$\mathbb{P}\left(\mathcal{M}(D) \in S\right) \leq e^{\epsilon}\mathbb{P}\left(\mathcal{M}(D') \in S\right) + \delta. \tag{5.1}$$

An equivalent characterization of (ϵ, δ)-DP can be stated based on the tail bound on the *privacy loss random variable* that is the log ratio of the probability density functions[1] of

[1]In this work, we assume the induced measures are absolutely cts wrt to the Lebesgue so pdf always exists.

$\mathcal{M}(D)$ and $\mathcal{M}(D')$.

Proposition 5.1 (See Lemma 3.17 in [DR$^+$14]). *A randomized mechanism M is (ϵ, δ)-differentially private if the log-likelihood ratio when evaluating on two neighboring databases remains bounded with probability at least $1 - \delta$, i.e.,*

$$\mathbb{P}\left(|\log \frac{pdf_D(o)}{pdf_{D'}(o)}| \leq \epsilon\right) \geq 1 - \delta, \tag{5.2}$$

where pdf_D ($pdf_{D'}$) is the pdf of $\mathcal{M}(D)$ ($\mathcal{M}(D')$) and o is drawn according to pdf_D.

A common design paradigm to approximate a deterministic function $q : \mathbb{D}^{|D|} \to R^p$ with a differentailly private mechanism is by adding a properly scaled Gaussian noise to the output of q. The scale of the noise depends on how far that query maps two neighboring databases which is formalized through the notion of sensitivity.

Definition 5.2 (L_2 sensitivity). *Let q be a deterministic function that maps a database to a vector in $\mathbb{R}^p$. The L_2 sensitivity of q is defined as*

$$\Delta_2(q) = \max_{D' \sim D} \|q(D) - q(D')\|_2. \tag{5.3}$$

We can define the L_1 sensitivity similarly.

Throughout this work, we focus on Gaussian Mechanism to ensure differential privacy.

Definition 5.3 (Gaussian mechanism). *Given any (deterministic) function $q : \mathbb{D}^{|D|} \to R^p$, the Gaussian Mechanism is defined as:*

$$\mathcal{M}(D, q, \sigma) = q(D) + n, \tag{5.4}$$

where n is a Gaussian random variable with a zero mean and the variance of σ.

It is well known [DR$^+$14] that for the proper value of the noise variance, Gaussian Mechanism preserves (ϵ, δ)-DP.

Proposition 5.2 (See for example Theorem 3.22 in [DR$^+$14]). *For a deterministic function $q : \mathbb{D}^{|D|} \to R^p$, Gaussian mechanism $\mathcal{M}(D, q, \sigma)$ preserves (ϵ, δ)-DP for $\epsilon < 1$ if $\sigma^2 \geq 2\log(1.25/\delta)\Delta_2(q)/\epsilon^2$, where $\Delta_2(q)$ is the L_2-sensitivity of the function q.*

5.2.1 Problem Formulation

We consider a distributed optimization set-up where N nodes aim to collaboratively minimize an additive cost function $f(x) = \sum_{i\in[N]} f_i(x), x \in \mathcal{X}$ where $\mathcal{X}$ is the domain of the problem and $[N] \stackrel{\triangle}{=} \{1, \ldots, N\}$. In this problem, nodes want to minimize $f(x)$ while keeping each data point private. We adopt the (ϵ, δ)-differential privacy (DP) as a measure of the privacy.

Assumption 5.1 (Domain)**.** *The domain of the optimization $\mathcal{X} \subseteq \mathbb{R}^p$ is a closed compact and convex set and $x^* \in \mathcal{X}$ where $x^* \in \arg\min_{x\in\mathbb{R}^p} f(x)$.*

In this setup, N nodes communicate over a connected and undirected graph $\mathcal{G} := (\mathcal{V}, \mathcal{E})$ where $\mathcal{V} := \{1, \cdots, N\}$ is the set of vertices and $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ denotes the set of edges. Nodes can only communicate with their neighbors, which we denote neighbors of node i with $\mathcal{N}_i$ for $i \in [N]$. We assume that each node has access to one of the summands of the global objective function, $f_i(x)$. Throughout this paper, the following assumption holds for the local objective functions[2].

Assumption 5.2 (Bounded Gradient)**.** *$f_i(x)$ for $i \in [N]$ are G-Lipschitz for $x \in \mathcal{X}$,*

$$\|f_i(x) - f_i(y)\|_2 \leq G\|x - y\|_2, \ \forall x, y \in \mathcal{X}, \ i \in [N]. \tag{5.5}$$

Assumption 5.3 (Smoothness)**.** *$f_i(x)$ for $i \in [N]$ are L-smooth over an open set containing $\mathcal{X}$,*

$$\|\nabla f_i(x) - \nabla f_i(y)\|_2 \leq L\|x - y\|_2, \ \forall x, y \in \mathcal{X}, \ i \in [N]. \tag{5.6}$$

In order to have convergence of the parameter itself, it is well-known that the strong convexity of the cost functions are needed.

Assumption 5.4 (Strong convexity)**.** *$f_i(x)$ for $i \in [N]$ are μ-strongly convex over an open set containing $\mathcal{X}$,*

$$\langle f_i(x) - f_i(y), x - y\rangle \geq \mu\|x - y\|_2^2, \ \forall x, y \in \mathcal{X}, \ i \in [N]. \tag{5.7}$$

[2]Assumption 5.3 and 5.4 inherently state that the smoothness and strong convexity parameters should be the same, however, without loss of generality we can take the maximum L and minimum μ across all nodes.

In the context of empirical risk minimization local cost functions are the empirical risk associated with data points stored in nodes, *i.e.*,

$$f_i(x) := \sum_{d \in D_i} l(x; d),$$

where $D_i \subseteq \mathbb{D}$ is the set of sensitive data points stored in the i-th node and $l(x, d)$ is the loss function.

Adversary model. In this problem, the adversary can overhear all the messages between nodes without any computational assumption. Nodes aim to preserve DP with respect to the sensitive data points $\cup_{i \in [N]} D_i$.

5.2.2 Overview of the Algorithm

We study the consensus-based distributed optimization algorithm where nodes update their local estimate by combining the information received from the neighbors. As opposed to the standard Distributed Gradient Descent (DGD), our proposed algorithm consists of two phases.

Stage I. In the first stage of the algorithm, similarly to the distributed gradient descent, nodes iteratively perform the consensus step followed by a local Gradient Descent (GD) step. Let us denote the local estimate of the i-th node with $x_i(t)$.

$$x_i(t) = \text{Proj}_{\mathcal{X}} \left(z_i(t) - \eta_t \nabla f_i(z_i(t)) \right), \quad t \leq T \tag{5.8}$$

where

$$z_i(t) = \text{Proj}_{\mathcal{X}} \left(\sum_{j \in \mathcal{N}_i} w_{ij} y_j(t) \right), \tag{5.9}$$

here $y_j(t)$ is the message sent by node $j \in \mathcal{N}_i$ to its neighbors, and

$$\text{Proj}_{\mathcal{X}}(x) \stackrel{\triangle}{=} \arg\min_{y \in \mathcal{X}} \|x - y\|_2 \tag{5.10}$$

is the Euclidean projection onto the set $\mathcal{X}$. The update (5.8) shows that node i updates it's local estimate by taking a proper average of the messages sent by its neighbors and

descending it through $\nabla f_i(z_i(t))$, w_{ij} is the weight associated to neighbors of node i and it's zero for $j \notin \mathcal{N}_i$, and $T \in \mathbb{N}$ is the number of steps in Stage II.

Assumption 5.5 (Doubly Stochastic Weight Matrix)**.** *The weight matrix, $W := [w_{ij}]$ is a doubly stochastic matrix with non-negative entries. We denote the second largest eigenvalue of W in absolute value with $\beta := \max\{|\lambda_2(W)|, |\lambda_N(W)|\}$, where $1 = \lambda_1(W) \geq \lambda_2(W) \ldots \geq \lambda_N(W) > -1$ are eigenvalues of W sorted in a descending order.*

It is well-known that using a fixed step size the classical DGD converges to a neighborhood of the optimal point with the size proportional to the step-size [NO09, YLY16]. Convergence to the exact point can be derived by using a diminishing step size (see [YLY16] and references therein). Therefore throughout this work, step-size η_t is chosen $\Theta(\frac{1}{t})$.

As opposed to the classical DGD, nodes do not send their local estimate directly since each step of GD may reveal information about the underlying sensitive data points. Nodes perturb their local estimate by a Gaussian mechanism to control the privacy leakage. In particular, nodes broadcast

$$y_i(t+1) = x_i(t) + n_i(t), \quad i \in \{1, \cdots, N\}, t \leq T, \tag{5.11}$$

where $n_i(t)$ is a zero mean additive Gaussian noise with the variance of M_t^2, and $x_i(t)$ is the local estimate of node i at t.

Each step of GD exposes the sensitive data points to the adversary. Hence to ensure the same level of privacy additional noise needed as the number of GD steps increases. Noise added to each stage of GD avoids the local estimates to converge to a common value therefore in our proposed method nodes apply GD only for T steps, and afterwards nodes switch to a purely consensus mode to agree on a common value.

Stage II. After T steps of Gradient descent, nodes iterates only though the consensus steps. Note that ∇f_i in (5.8) is the only source in which the privacy of sensitive data points may leak. Therefore in the second stage of the algorithm, nodes broadcast their local estimate

precisely and update their local estimate according to:

$$y_i(t) = x_i(t-1), \qquad i \in [N], \quad t > T$$
$$x_i(t) = \sum_{j \in [N]} w_{ij} y_j(t), \tag{5.12}$$

.

Note that projection operator is not needed in (5.12) due to the convexity of the optimization domain (Assumption 5.1). Nodes update their local estimates until a stopping criterion is met. One common stopping criterion is the relative change in the value of each node.

Algorithm 8: Steps at Node i for the proposed private distributed algorithm.

Set $y_i(1) = 0$;
for $t = 1, \cdots, T$ **do**
    Update $y_i(t)$ according to (5.11) and broadcast $y_i(t)$;
    Receive $y_j(t)$ from $j \in \mathcal{N}_j$;
    Update $x_i(t)$ according to (5.8) ;
end
for $t = T+1, \cdots$ **do**
    Broadcast $y_i(t) := x_i(t-1)$;
    Receive $y_j(t)$ from $j \in \mathcal{N}_j$;
    Update $x_i(t)$ according to (5.12) ;
end

Notation. We represent set of natural and real numbers with $\mathbb{N}$ and $\mathbb{R}$ respectively. Throughout this work, we reserve the lowercase bold letters for the aggregated parameters of nodes at a given time t, *i.e.*, we use the notation $\mathbf{x}(t) = [x_1(t); \cdots; x_N(t)]$, $\mathbf{y}(t) = [y_1(t); \cdots; y_N(t)]$, $\mathbf{z}(t) = [z_1(t); \cdots; z_N(t)]$ and $\mathbf{n}(t) = [n_1(t); \cdots; n_N(t)]$. The identity matrix is denoted with $I_p \in \mathbb{R}^p$ and we use $\mathbf{1_N}$ to represent a vector of length N of ones. We use $\|a\|$ to denote l_2-norm of a vector a while $\|A\|$ represents the operator norm of a matrix

A, $\|A\| \triangleq \sup_{\|x\|=1} \|Ax\|$, and $\otimes$ denotes the Kronecker product. In this paper we reserve the notation $[N]$ to represent $\{1, \cdots, N\}$ for any $N \in \mathbb{N}$.

5.3 Main results

In this section, we give the main results of this work. First we derive conditions on the noise variances under which the distributed problem is (ϵ, δ)-differentially private against an adversary that oversees all the communications among nodes. We emphasize that one of the main contribution of this work is to relate the noise variances at different steps to the privacy parameters (ϵ and δ) directly rather than using basic or advanced composition theorems. Afterward we state the convergence result.

The variance of the noise to ensure (ϵ, δ)-DP depends on the L_2 sensitivity of the algorithm. In the context of distributed optimization, we need to make sure DP is satisfied despite multiple rounds of communications in this iterative scheme.

Definition 5.4 (Conditional L_2-sensitivity)**.** *Conditional L_2-sensitivity at round t, $\Delta(t)$ defined to be the maximum L_2-norm difference of $x_i(t)$ evaluated on two neighboring databases D and D' while having the same set of messages $\{\mathbf{y}(k)\}_{k=1}^{t}$ up until round t,* i.e.*,*

$$\Delta(t) := \sup_{D \sim D'} \sup_{i \in [N]} \sup_{\{\mathbf{y}(t)\}_{t=1}^{t}} \|x_i^D(t) - x_i^{D'}(t)\|_2, \tag{5.13}$$

where $x_i^{D'}(t)$ corresponds to the local parameter of node i of the neighboring instance of the problem.

Proposition 5.3. *The conditional L_2-sensitivity of Algorithm 8 at round $t \leq T$ is bounded by $2\eta_t G$, provided that Assumption 5.2 holds.*

Proof. See Appendix. □

Now we have the machinery to state the main theorem of this section.

Theorem 5.1. *The distributed algorithm is (ϵ, δ)-DP if nodes perturb their local estimates by adding independent Gaussian noise* (5.11) *and the following holds,*

$$\sum_{t=1}^{T} \frac{\Delta^2(t)}{M_t^2} \leq \frac{\epsilon^2}{\epsilon + 2\log\frac{2}{\delta}}, \tag{5.14}$$

where M_t is the scale of noise added in round $t \leq T$.

Corollary 5.1. *If Assumption 5.2 holds and*

$$\sum_{t=1}^{T} \frac{\eta_t^2}{M_t^2} \leq \frac{\epsilon^2}{4G^2\left(\epsilon + 2\log\frac{2}{\delta}\right)} \triangleq \kappa(\epsilon, \delta). \tag{5.15}$$

then the distributed algorithm is (ϵ, δ)-DP.

Remark 5.1. *A common practice to ensure differential privacy in an iterative mechanism is to make each step differentially private (with a stronger privacy guarantees) and combine the privacy leakage using the basic or advanced composition theorems [DR+14]. Composition theorems do not take into account the specific noise distribution and they often give loose results for a given distribution while Theorem 5.1 takes the noise distribution into account thereby giving a tighter result,* i.e., *smaller noise variances and therefore ensure a better utility.*

Remark 5.2. *In the literature of DP for iterative processes, it is common that in order to ensure (ϵ, δ)-DP we make each step (ϵ', δ')-DP, where ϵ' and δ' are computed using a composition theorem. It implicitly assumes that we need to have the same privacy requirement at each step, which is not necessary needed. Theorem 5.1 connects the noise variances across time to the privacy parameters ϵ and δ directly and allows for a meaningful assignment of privacy budget to different steps.*

Remark 5.3. *In the regime where $\epsilon \ll 1$ and $\delta \ll \frac{1}{N}$, the bound in Theorem 5.1 can be written as,*

$$\sum_{t=1}^{T} \frac{\Delta^2(t)}{M_t^2} \leq \frac{\epsilon^2}{2\log\frac{2}{\delta}}. \tag{5.16}$$

Theorem 5.1 extends Lemma 2 the result of [HMV15] to (ϵ, δ)-DP. It gives us the condition under which the distributed algorithm is (ϵ, δ)-DP. Working with (ϵ, δ)-DP as opposed to ϵ-DP in [HMV15], enables us to derive a convergence result with a diminishing regret bound when the privacy requirement weakens.

In the rest of this section, we state the convergence result. Let us denote the average of the local estimates with $\bar{x}(t) \stackrel{\triangle}{=} \frac{1}{N} \sum_{i \in [N]} x_i(t)$. Theorem 5.2 summarizes the convergence result for the mean parameter $\bar{x}(t)$.

Theorem 5.2. *Under Assumptions 5.2, 5.3 and 5.4 with the step size* $\eta_t = \frac{\mu+L}{2\mu L} \frac{1}{t}$ *and noise scales* $M_t^2 = \frac{2}{\kappa(\epsilon,\delta)} \left(\frac{\mu+L}{2\mu L}\right)^2 \frac{\sqrt{T}}{t\sqrt{t}}$*, the distributed algorithm 8 is differentially private and the following bound holds on* $\mathbb{E}[\|\bar{x}(T) - x^*\|_2^2]$*:*

$$\mathbb{E}[\|\bar{x}(T) - x^*\|_2^2] \leq C_T \frac{1}{T} + C_{\log T} \frac{\log T}{T} + C_{\sqrt[4]{T}} \frac{1}{\sqrt[4]{T}} + C_{(\epsilon,\delta)}$$

where x^* *minimizes,* C_T*,* $C_{\log T}$*,* $C_{\sqrt[4]{T}}$ *and* $C_{(\epsilon,\delta)}$ *are constants:*

$$\begin{aligned}
C_T &= \frac{S(0)}{N} \\
C_{\log T} &= G^2 \left(1 + \frac{1}{1-\beta}\right) \left(\frac{\mu+L}{2\mu L}\right)^2 \\
C_{\sqrt[4]{T}} &= \frac{2\sqrt{2p}G}{\sqrt{\kappa(\epsilon,\delta)}} \left(4 + \frac{3}{1-\beta}\right) \left(\frac{\mu+L}{2\mu L}\right)^2 \\
C_{(\epsilon,\delta)} &= \frac{2p}{\kappa(\epsilon,\delta)} \left(\frac{\mu+L}{2\mu L}\right)^2 \\
\kappa(\epsilon,\delta) &= \frac{\epsilon^2}{4G^2\left(\epsilon + 2\log\frac{2}{\delta}\right)}
\end{aligned}$$

Theorem 5.2 states that the average parameter converges to a neighborhood of the optimal point with the rate of $O(\frac{1}{\sqrt[4]{T}})$. The neighborhood scales with the privacy parameters (ϵ, δ) which is $O(\frac{\log \frac{1}{\delta}}{\epsilon^2})$. Recall the second stage of the distributed algorithm only consists of the consensus steps in which local parameters converge to a common value in a linear rate.

Corollary 5.2. *Under Assumptions 5.2, 5.3 and 5.4 with the step size* $\eta_t = \frac{\mu+L}{2\mu L} \frac{1}{t}$ *and noise scales* $M_t^2 = \frac{2}{\kappa(\epsilon,\delta)} \left(\frac{\mu+L}{2\mu L}\right)^2 \frac{\sqrt{T}}{t\sqrt{t}}$*, the distributed algorithm 8 is deferentially private and the*

following bound holds on the local parameters in the second stage of the algorithm, $t > T$:

$$\mathbb{E}[\|x_i(t) - x^*\|_2^2] \leq \tag{5.17}$$
$$2C_{exp}\beta^{2t-2T} + 2C_T\frac{1}{T} + 2C_{\log T}\frac{\log T}{T} + 2C_{\sqrt[4]{T}}\frac{1}{\sqrt[4]{T}} + 2C_{(\epsilon,\delta)}$$

where C_T, $C_{\log T}$, $C_{\sqrt[4]{T}}$ and $C_{(\epsilon,\delta)}$ defined in Theorem 5.2 and $C_{exp} = 2\|\mathbf{x}(T)\|^2$.

Proof. We observe that in Stage II, the mean parameter $\bar{x}(t)$ remains constant since

$$\begin{aligned}\bar{x}(t+1) &\overset{(a)}{=} \frac{1}{N}\left(\mathbf{1}_N^T \otimes I_P\right)\mathbf{x}(t+1)\\ &\overset{(b)}{=} \frac{1}{N}\left(\mathbf{1}_N^T \otimes I_P\right)(W \otimes I_p)\,\mathbf{x}(t)\\ &\overset{(c)}{=} \frac{1}{N}\left(\mathbf{1}_N^T \otimes I_P\right)\mathbf{x}(t) = \bar{x}(t),\end{aligned} \tag{5.18}$$

where we rewrote the mean parameter using the Kronecker product in (a), (b) follows directly from (5.12) and we used double stochasity of W in (c). Now we are ready to conclude the result,

$$\begin{aligned}\mathbb{E}[\|x_i(t) - x^*\|_2^2] &\overset{(a)}{=} \mathbb{E}[\|x_i(t) - \bar{x}(t) + \bar{x}(T) - x^*\|_2^2]\\ &\overset{(b)}{\leq} 2\mathbb{E}[\|x_i(t) - \bar{x}(t)\|_2^2] + 2\mathbb{E}[\|\bar{x}(T) - x^*\|_2^2],\end{aligned} \tag{5.19}$$

where (a) follows from (5.18), and we used the inequality $\|a+b\|_2^2 \leq 2\|a\|_2^2 + 2\|b\|_2^2$ in (b). Using Theorem 5.2 and Lemma 5.1 it is straightforward to conclude the result. □

5.4 Privacy and Convergence Analysis

In this section we outline the proofs for Theorems 5.1 and 5.2 followed by explanation and intuition.

5.4.1 Theorem 5.1

In the context of differential privacy, the corresponding mechanism for the distributed algorithm maps $D := \cup_{i\in[N]} D_i$ to a sequence of messages $\{\mathbf{y}(t)\}_{t=1}$. In order to satisfy (ϵ, δ)-DP,

the output of the mechanism should satisfy the condition (5.2) in Proposition 5.1. We proceed by writing the privacy loss random and bounding it using the concentration inequalities. The complete proof is included in Appendix.

5.4.2 Theorem 5.2

It is straightforward to verify this choice of η_t and M_t satisfy (5.2) and therefore the distributed algorithm is deferentially private. The proof of convergence consists of two parts. First we show that the local parameters are bounded away from mean in expectation, which is depicted in Lemma 5.1. The proof proceeds by bounding deviation of the mean parameter to the optimal point. Putting these together, the result follows.

Lemma 5.1. *Under Assumption 5.2, at round t, the following bound holds on the distance of local parameters to the mean for $t < T$,*

$$\|\mathbf{z}(t) - \mathbf{1}_N \otimes \bar{z}(t)\| \leq \|\mathbf{n}(t)\| + 2\sum_{s=1}^{t-1} \beta^{t-s}\|\mathbf{n}(s)\| + \sqrt{N}G\sum_{s=1}^{t-1}\eta_s\beta^{t-s}, \tag{5.20}$$

where $\bar{z}(t) \stackrel{\triangle}{=} \frac{1}{N}\sum_{i\in[N]} z_i(t)$. And the following holds for $t \geq T$,

$$\|\mathbf{x}(t) - \mathbf{1}_N \otimes \bar{x}(t)\| \leq \beta^{t-T}\|\mathbf{x}(T)\|. \tag{5.21}$$

where $\otimes$ denotes the Kronecker product.

Proof. See Appendix. □

Let us define $S(t) \stackrel{\triangle}{=} \sum_{i\in[N]} \mathbb{E}[\|x_i(t) - x^*\|^2]$ where x^* is the global minimum of $f(x)$. Observe that,

$$\mathbb{E}[\|\bar{x}(t) - x^*\|^2] \leq \frac{1}{N}S(t),$$

where we used $\left(\sum_{i\in[N]} \|a_i\|\right)^2 \leq N\sum_{i\in[N]}\|a_i\|^2$. Therefore we proceed by bounding $\mathbb{E}[\|x_i(t) - x^*\|^2]$ for $i \in [N]$ in order to bound $\mathbb{E}[\|\bar{x}(t) - x^*\|^2]$.

Using the standard techniques, we bound terms $\|x_i(t) - x^*\|^2$ one by one (for the clarity of the presentation, the time index dropped wherever it is clear from the context). It is well known that the projection operator is *non-expansive*, *i.e.*, $\|\text{Proj}_{\mathcal{X}}(x) - \text{Proj}_{\mathcal{X}}(y)\| \leq \|x - y\|$ for $x, y \in \mathbb{R}^p$. Putting this together with Assumption 5.1 ($x^* \in \mathcal{X}$) we have,

$$\begin{aligned}\|x_i(t) - x^*\|^2 &= \|\text{Proj}_{\mathcal{X}}\left(z_i - \eta_t \nabla f_i(z_i)\right) - \text{Proj}_{\mathcal{X}}\left(x^*\right)\|^2 \\ &\leq \|z_i(t) - \eta_t \nabla f_i(z_i) - x^*\|^2 \end{aligned} \tag{5.22}$$

In order to bound RHS of (5.22) we first present a lemma.

Lemma 5.2 (see for example Theorem 2.1.12 in [Nes07])**.** *Suppose that f is L-smooth and μ-strongly over an open set containing $\mathcal{X}$, then we have,*

$$\langle x - y, \nabla f(x) - \nabla f(y)\rangle \geq c_1 \|x - y\|^2 + c_2 \|\nabla f(x) - \nabla f(y)\|^2, \tag{5.23}$$

where $c_1 = \frac{\mu L}{\mu + L}$ and $c_2 = \frac{1}{\mu + L}$.

We proceed by expanding (5.22) and by adding and subtracting $\nabla f_i(x^*)$,

$$\begin{aligned}&\|x_i(t) - x^*\|^2 \\ &\quad\leq \|z_i - x^*\|^2 - 2\eta_t \langle z_i - x^*, \nabla f_i(z_i) - \nabla f_i(x^*) + \nabla f_i(x^*)\rangle \\ &\qquad\qquad + \eta_t^2 \|\nabla f_i(z_i)\|^2 \\ &\quad\overset{(a)}{\leq} (1 - \frac{2\mu L}{\mu + L}\eta_t)\|z_i - x^*\|^2 - \frac{2\eta_t}{\mu + L}\|\nabla f_i(z_i) - \nabla f_i(x^*)\|_2^2 \\ &\qquad\qquad + 2\eta_t \langle \nabla f_i(x^*), x^* - z_i\rangle + \eta_t^2 \|\nabla f_i(z_i)\|^2 \\ &\quad\overset{(b)}{\leq} (1 - \frac{2\mu L}{\mu + L}\eta_t)\|z_i - x^*\|^2 + \eta_t^2 G^2 + 2\eta_t \langle \nabla f_i(x^*), x^* - z_i\rangle \\ &\quad\overset{(c)}{\leq} (1 - \frac{2\mu L}{\mu + L}\eta_t)\|z_i - x^*\|^2 + \eta_t^2 G^2 + 2\eta_t \langle \nabla f_i(x^*), x^* - \bar{z}\rangle \\ &\qquad\qquad - 2\eta_t \langle \nabla f_i(x^*), z_i - \bar{z}\rangle \end{aligned} \tag{5.24}$$

where (a) follows from Lemma 5.2. We used Assumption 5.2 for (b), and (c) comes from adding and subtracting $\bar{z}$. The following lemma is useful in order to connect (5.24) to $S(t)$.

Lemma 5.3. *For any (fixed) $x \in \mathcal{X}$ the following holds,*

$$\sum_{i\in[N]} \mathbb{E}[\|z_i(t) - x\|^2] \leq \sum_{i\in[N]} \mathbb{E}[\|x_i(t-1) - x\|^2] + dNM_{t-1}^2. \tag{5.25}$$

Proof. See Appendix. □

By summing up both sides of (5.24) across all nodes and Lemma 5.3 we have:

$$\begin{aligned} S(t) \leq (1 - \frac{2\mu L}{\mu + L}\eta_t)S(t-1) + dN(1 - \frac{2\mu L}{\mu + L}\eta_t)M_{t-1}^2 \\ + N\eta_t^2 G^2 + 2\eta_t \sum_{i\in[N]} \mathbb{E}[\nabla\langle \nabla f_i(x^*), \bar{z} - z_i\rangle], \end{aligned} \tag{5.26}$$

where we used the fact that x^* is the global minimum and therefore $\sum_{i\in[N]} \nabla f_i(x^*) = 0$. It remains to bound the last term in RHS of (5.26). Note that,

$$\begin{aligned} \sum_{i\in[N]} \langle \nabla f_i(x^*), \bar{z} - z_i\rangle &\overset{(a)}{\leq} G \sum_{i\in[N]} \|\bar{z}(t) - z_i(t)\| \\ &\overset{(b)}{\leq} G\sqrt{N}\|\mathbf{z}(t) - \mathbf{1}_N \otimes \bar{z}(t)\|, \end{aligned} \tag{5.27}$$

where (a) follows from Cauchy-Schwartz inequality along with Assumption 5.2 and we used $\left(\sum_{i\in[N]} \|a_i\|\right)^2 \leq N \sum_{i\in[N]} \|a_i\|^2$ in (b). By applying Lemma 5.1 and taking expectation from both sides of (5.27) we have,

$$\begin{aligned} &\sum_{i\in[N]} \mathbb{E}[\nabla\langle \nabla f_i(x^*), \bar{z} - z_i\rangle] \\ &\qquad \leq \sqrt{pN}M_t + 2\sqrt{p}\sqrt{N}\sum_{s=1}^{t-1} M_s\beta^{t-s} + G\sqrt{N}\sum_{s=1}^{t-1}\eta_s. \end{aligned} \tag{5.28}$$

Together with (5.26) we have the following recursion for $S(t)$,

$$\begin{aligned} S(t) \leq (1 - \frac{2\mu L}{\mu + L}\eta_t)S(t-1) + pN(1 - \frac{2\mu L}{\mu + L}\eta_t)M_{t-1}^2 \\ + N\eta_t^2 G^2 \\ + G\sqrt{p}NM_t + 2G\sqrt{p}N\sum_{s=1}^{t-1} M_s\beta^{t-s} + G^2 N\sum_{s=1}^{t-1}\eta_s\beta^{t-s}. \end{aligned} \tag{5.29}$$

By taking step size of $\eta_t = \frac{\mu+L}{2\mu L}\frac{1}{t}$, we bound the cumulative effect of each term in (5.29) for $S(T)$, where T is the number of steps we are evaluating the gradient.

$$
\begin{aligned}
S(T) \leq & \prod_{t=2}^{T}(1-\frac{1}{t})S(0) && (5.30)\\
& + pN\sum_{t=1}^{T} M_{t-1}^2 \prod_{s=t}^{T}(1-\frac{1}{s}) \\
& + G^2 N\sum_{t=1}^{T} \eta_t^2 \prod_{s=t+1}^{T}(1-\frac{1}{s}) \\
& + 2G\sqrt{p}N\sum_{t=1}^{T} \eta_t M_t \prod_{s=t+1}^{T}(1-\frac{1}{s}) \\
& + 2G\sqrt{p}N\sum_{t=2}^{T} \eta_t \left(\sum_{s=1}^{t-1} M_s \beta^{t-s}\right) \prod_{s=t+1}^{T}(1-\frac{1}{s}) && (*1)\\
& + G^2 N\sum_{t=2}^{T} \eta_t \left(\sum_{s=1}^{t-1} \eta_s \beta^{t-s}\right) \prod_{s=t+1}^{T}(1-\frac{1}{s}) && (*2)
\end{aligned}
$$

The second term (5.30) is dominant in terms of the noise variance, and the noise scales $M_t^2 = \frac{2}{\kappa(\epsilon,\delta)}\left(\frac{\mu+L}{2\mu L}\right)^2 \frac{\sqrt{T}}{t\sqrt{t}}$ are found by minimizing this term while taking condition (5.15) in Theorem 5.1 as the constraint. Therefore,

$$
\begin{aligned}
S(T) \leq & \frac{S(0)}{T} + \frac{4pN}{\kappa(\epsilon,\delta)}\left(\frac{\mu+L}{2\mu L}\right)^2 && (5.31)\\
& + G^2 N\left(\frac{\mu+L}{2\mu L}\right)^2 \frac{\log T}{T} && (5.32)\\
& + \frac{8\sqrt{2p}GN}{\sqrt{\kappa(\epsilon,\delta)}}\left(\frac{\mu+L}{2\mu L}\right)^2 \frac{1}{\sqrt[4]{T}} && (5.33)\\
& + \frac{2\sqrt{2p}GN}{\sqrt{\kappa(\epsilon,\delta)}}\left(\frac{\mu+L}{2\mu L}\right)^2 \left(\frac{3}{1-\beta}\right)\frac{1}{\sqrt[4]{T}} && (5.34)\\
& + G^2 N\left(\frac{\mu+L}{2\mu L}\right)^2 \left(\frac{1}{1-\beta}\right)\frac{\log T}{T}, && (5.35)
\end{aligned}
$$

where we used $\sum_{t=1}^{T}\frac{1}{t} \leq \log(T)$ and $\sum_{t=1}^{T}\frac{1}{t^\alpha} \leq \frac{1}{\alpha+1}T^{\alpha+1}$ for $\alpha > -1$ to derive (5.31), (5.32)

and (5.33). Going from (∗1) to (5.34) follows from

$$\sum_{t=2}^{T}\frac{1}{t}\left(\sum_{s=1}^{t-1}\frac{t}{\sqrt{s}\sqrt[4]{s}}\beta^{t-s}\right)=\sum_{s=1}^{T-1}\frac{1}{s^{3/4}}\left(\sum_{t=s+1}^{T}\beta^{t-s}\right)$$
$$\leq\frac{1}{1-\beta}\sum_{s=1}^{T-1}\frac{1}{s^{1/2+c/4}}$$
$$\leq\frac{4}{1-\beta}\sqrt[4]{T}. \tag{5.36}$$

And we used the following inequality to go from (∗2) to (5.35),

$$\sum_{t=2}^{T}\frac{1}{t}\left(\sum_{s=1}^{t-1}\frac{1}{s}\beta^{t-s}\right)=\sum_{s=1}^{T-1}\frac{1}{s}\left(\sum_{t=s+1}^{T}\beta^{t-s}\right)$$
$$\leq\frac{1}{1-\beta}\sum_{s=1}^{T-1}\frac{1}{s}\leq\frac{1}{1-\beta}\log T. \tag{5.37}$$

The result follows immediately by $\mathbb{E}[\|\bar{x}(T)-x^*\|_2^2]\leq\frac{1}{N}S(T)$.

5.5 Numerical Experiments

In this section, we assess the performance of our method on decentralized mean estimation and we demonstrate the effect of privacy parameters, number of gradient evaluation and graph topology on the error. For the simulations, the communication graph is a connected Erdos-Renyi with edge probability of $p_c=0.6$ and the weight matrix is $W=I-\frac{2}{3\lambda_{\max}(L)}L$ where L is the Laplacian of the graph.

5.5.1 Distributed Mean Estimation

Distributed mean estimation is one of the classical problems in the domain of differential privacy. Suppose data points lie in a cube, $\mathcal{X}:=[-R,R]^p$, where p is the dimension of the points and R is the length of each side. In this setup, each node has several data points and they aim to collaboratively find the mean while keeping each data private and preserve (ϵ,δ)-DP against an adversary that oversees all the messages. We can write down this as the following distributed problem:

$$\bar{d} = \min_{x \in \mathcal{X}} f(x) := \frac{1}{2} \sum_{d \in \cup_{i \in [N]} D_i} \|x - d\|_2^2, \tag{5.38}$$

where D_i is the set of points stored in node i, and $f_i(x) = \frac{1}{2} \sum_{d \in D_i} \|x - d\|_2^2$ for $i \in [N]$. We generate data randomly according to a truncated Gaussian distribution with mean of $0.7R$ and the unit variance. Sensitive data points are distributed among 10 nodes each of which has 100 data points *i.e.*, $|D_i| = 100$. The conditional l_2 sensitivity of the distributed algorithm

$$\begin{aligned} \Delta(t) &\stackrel{\triangle}{=} \sup_{i \in [N]} \sup_{D \sim D'} \|x_i^D(t) - x_i^{D'}(t)\|_2 \\ &\leq \sup_{d, d' \in \mathbb{D}} \eta_t \|d - d'\|_2 \leq 2R\sqrt{p}\eta_t, \end{aligned}$$

and we generate Gaussian noise accordingly.

In order to demonstrate the convergence rate of the algorithm, we run the distributed algorithm for different values of T, number of gradient descent steps. Figure 5.1 illustrates the effect of T on the normalized error $\frac{\|\bar{x}(T) - \bar{d}\|_2^2}{\|\bar{d}\|_2^2}$. It shows that the error reduces until reaching a neighborhood of x^*, which agrees with intuition provided by Theorem 5.2.

Figure 5.2 demonstrates the normalized error versus ϵ for different values of $\delta \in \{1/(N * N_i), 1/(N * N_i)^2, 1/(N * N_i)^3\}$ where N_i denotes the number of data points in each node, and $T = 1000$. We observe that for a fixed value of δ by strengthening the privacy guarantee the error increases $O(\frac{1}{\epsilon^2})$.

To observe the effect of the graph topology, we fix the privacy parameters and vary the connectivity probability of the underlying graph by choosing $p_c \in \{0.1, 0.3, 0.6, 1\}$. Figure 5.3 illustrates the effect of connectivity on error of an individual node $\frac{\|x_i(T) - \bar{d}\|_2^2}{\|\bar{d}\|_2^2}$.

Figure 5.4 illustrates the regret bound when the number of data points per each node is increasing. We observe a gain in the utility bound that is inline with Theorem 5.2. Increasing the number of data points doesn't increase the conditional sensitivity while making the Gradient bigger hence the effective added noise is reduced.

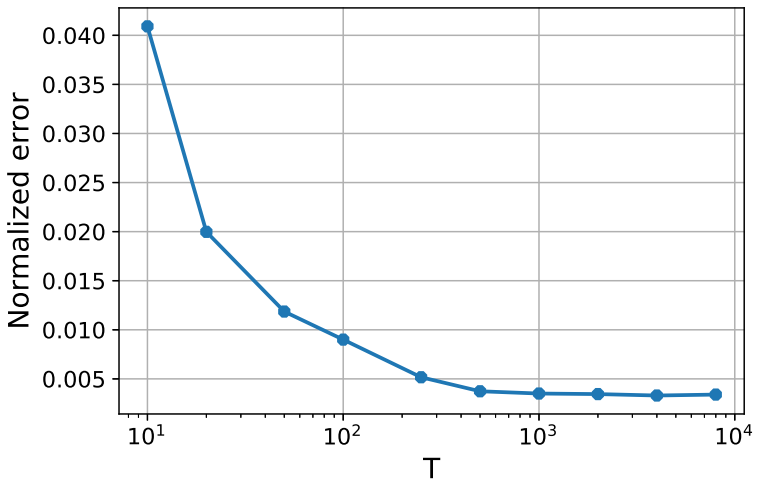


Figure 5.1: The normalized error vs. the number of gradient descent steps, T for $\epsilon = 4$ and $\delta = 1/(N * N_i)$.

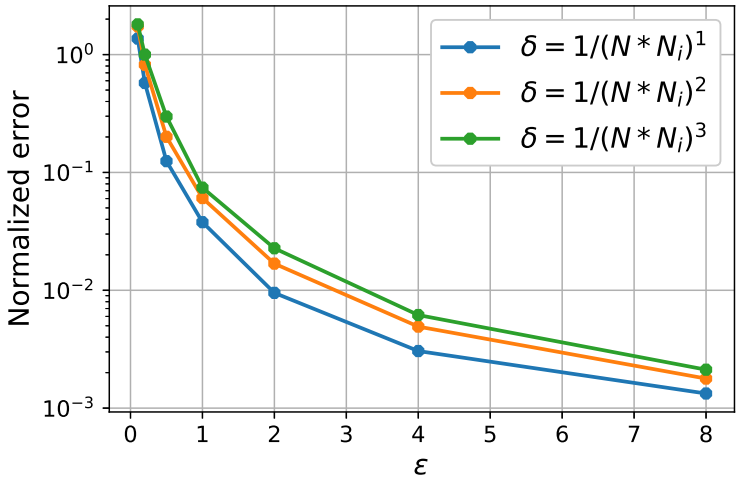


Figure 5.2: The normalized error of the distributed mean estimation vs. ϵ for a fixed number of nodes ($N = 10$) and data points per node ($N_i = 100$) with $T = 1000$.

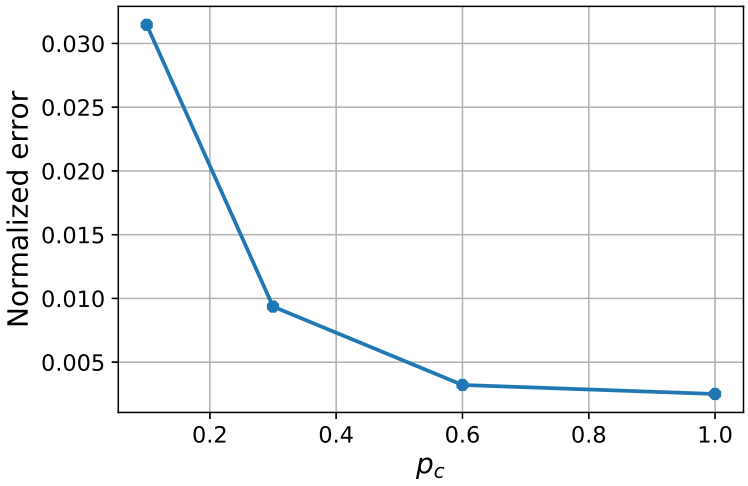


Figure 5.3: The normalized error of the first node vs. the edge probability p_c, for $\epsilon = 4$, $\delta = 1/(N * N_i)$, and $T = 1000$. As the connectivity of the graph increases, we observe a decrease in the error of the first node.

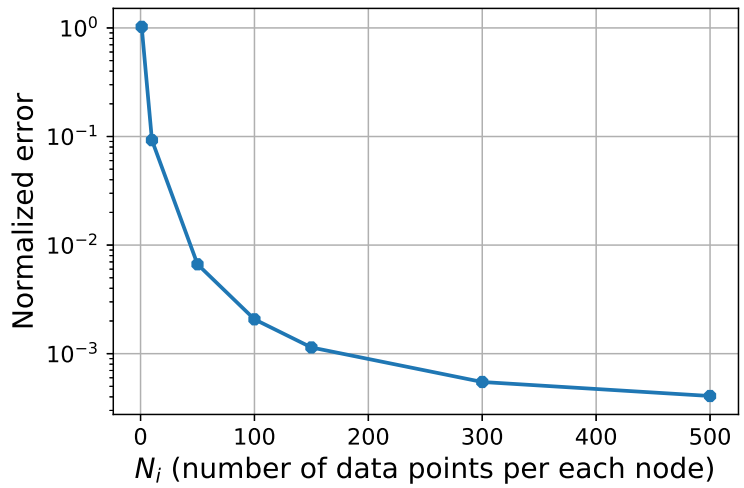


Figure 5.4: The normalized error vs. the number of data points per each node for $T = 1000$, $\epsilon = 4$ and $\delta = 1/(N * N_i)$.

5.6 Conclusion

In this work, we studied the consensus-based distributed optimization algorithm when the data points are distributed across several trusted nodes and the privacy of each data point is important against an adversary that oversees communications across nodes. In order to protect the privacy of users, each node perturbs it's local state with an additive noise before sending it out to its neighbors. Differential privacy is a rigorous privacy criterion for data analysis that provides meaningful guarantees regardless of what an adversary knows ahead of time about individuals' data. We considered (ϵ, δ)-DP as the privacy measure and we derived the amount of noise needed in order to guarantee privacy of each data point. We further showed that the parameters converge to a neighborhood of the optimal point, and the size of the neighborhood is proportional to the privacy metrics.

CHAPTER 6

Conclusions and Future Work

In the first part this dissertation, we addressed two problems concerning cyber-physical systems: secure state estimation and secure system identification. In Chapter 2, we studied the state estimation problem while *both* inputs and outputs are subject to adversarial attacks. We introduced the notion of sparse-strong observability and we showed this is the key property that systems should have in order to be resilient against such attacks. In Chapter 3 we studied the problem of system identification of linear time invariant systems under adversarial attacks on sensors. Given a bound on the number of attacked sensors, and under certain sparse-observability assumption, we showed that it is still possible to construct a meaningful model that enables stabilization of this system. We defined the notion of similarity modulo outputs and showed that all of such models are similar modulo outputs. This notion generalized the idea of equivalent systems in classical linear system theory.

Although we analyzed the problem only for linear time invariant systems, it is an straightforward task to generalize these notion for non-linear system, however, verifying these properties for this broader class of systems can be quite challenging. Specifically, an interesting research direction would be to gain a more comprehensive understanding of these properties for non-linear systems.

Data privacy is an important concern in machine learning, and is fundamentally at odds with the task of training useful learning models. The second part of this dissertation is devoted to privacy concerns in learning algorithms by considering differential privacy as the privacy metric. We studied two problems in private data analysis: private linear regression and private distributed optimization. One possible way of fulfilling the machine learning

task while preserving user privacy is to train the model on a transformed, noisy version of the data, which does not reveal the data itself directly to the training procedure. In Chapter 4 we analyzed the privacy-utility trade-off of two such schemes for the problem of linear regression: additive noise, and random projections. We illustrated that projecting the data to a lower-dimensional subspace before adding noise attains a better trade-off in *linear* model. Extending the idea of random projections to more sophisticated models is one possible future direction.

In Chapter 5 we turned to the distributed setup in which there are several trusted nodes that collaboratively aim to learn a model. Each node has a local objective that depends on their sensitive data. Particularly, we modified the vanilla distributed gradient descent algorithm to ensure the privacy of users against an adversary that overhears communications among nodes. In order to protect the privacy of users, each node perturbs its local state with an additive Gaussian noise before sending it out to its neighbors. We showed the parameter converges to a ball around the optimal point, and the size of the neighborhood scales with the privacy metrics. There are many exciting and interesting future directions in understanding how to design and implement private learning algorithms in distributed setup.

The private optimization framework we developed in this thesis assumes convex loss function across nodes. Given a recent rise of deep learning models, it is indeed interesting to extend the ideas to non-convex models. We hope that this thesis contributes to the development of the foundational ideas for privacy.

APPENDIX A

Proofs for Chapter 2

A.1 Proof of Lemma 2.1

We first prove the sufficiency part. For the sake of contradiction, suppose that the underlying system is not strongly observable but the property of Corollary 2.1 is true. If the underlying system (2.6) is not strongly observable, it means there exist two initial conditions, denoted by $x^{(1)}(0)$ and $x^{(2)}(0)$ possibly with different input sequences denoted by $\{u^{(1)}(t)\}$ and $\{u^{(2)}(t)\}$, respectively, that correspond to the same output sequence $\{y(t)\}$. The underlying system is linear, therefore the nonzero initial condition of $x^{(1)}(0) - x^{(2)}(0)$ with the input sequence $\{u^{(1)}(t) - u^{(2)}(t)\}$ produces the zero output sequence which contradicts the property given in Corollary 2.1. The necessity can be concluded using the similar argument. For the sake of contradiction let us assume this property does not hold, i.e., there exists a non zero initial state $x(0) \neq 0$ that corresponds to the zero output sequence. This contradicts the strong observability since the zero output sequence can be generated from both zero and $x(0) \neq 0$ as initial conditions under (possibly different) input sequences.

A.2 Proof of Lemma 2.2

We prove this lemma with contradiction. We show that if $\text{TEST}(\Gamma_u^{\text{cert}}, \Gamma_y^{\text{temp}} \cup \{i\})$ returns true for all $i \in \Gamma_y^{\text{SAT}} \setminus \Gamma_y^{\text{temp}}$ then $\text{TEST}(\Gamma_u^{\text{cert}}, \Gamma_y^{\text{SAT}})$ would also return true, which contradicts the assumption of the lemma. By applying the following lemma successively, the result follows directly.

Lemma A.1. *Assume that the system S is $(2r, 2s)$-sparse strongly observable. Pick any subset of inputs and outputs denoted by Γ_u^{cert} and Γ_y^{temp} with $|\Gamma_u^{cert}| \leq 2r$ and $|\Gamma_y^{temp}| \geq p-2s$. Then for any subsets of outputs denoted by Γ_y^1 and Γ_y^2, the first statement implies the second:*

1. *TEST($\Gamma_u^{cert}, \Gamma_y^{temp} \cup \Gamma_y^1$) and TEST($\Gamma_u^{cert}, \Gamma_y^{temp} \cup \Gamma_y^2$) return true.*
2. *TEST($\Gamma_u^{cert}, \Gamma_y^{temp} \cup \Gamma_y^1 \cup \Gamma_y^2$) returns true.*

Proof. Without loss and generality we can assume Γ_y^1, Γ_y^2 and Γ_y^{temp} are all disjoint sets. Since TEST($\Gamma_u^{\text{cert}}, \Gamma_y^{\text{temp}} \cup \Gamma_y^i$) returns true for $i \in \{1, 2\}$, therefore we have

$$\begin{bmatrix} \mathbf{Y}|_{\Gamma_y^{\text{temp}}} \\ \mathbf{Y}|_{\Gamma_y^1} \end{bmatrix} = \begin{bmatrix} \mathcal{O}_{\Gamma_y^{\text{temp}}} \\ \mathcal{O}_{\Gamma_y^1} \end{bmatrix} \hat{x}^1 + \begin{bmatrix} \mathcal{N}_{\Gamma_u^{\text{cert}} \to \Gamma_y^{\text{temp}}} \\ \mathcal{N}_{\Gamma_u^{\text{cert}} \to \Gamma_y^1} \end{bmatrix} \hat{\mathbf{U}}^1, \tag{A.1}$$

$$\begin{bmatrix} \mathbf{Y}|_{\Gamma_y^{\text{temp}}} \\ \mathbf{Y}|_{\Gamma_y^2} \end{bmatrix} = \begin{bmatrix} \mathcal{O}_{\Gamma_y^{\text{temp}}} \\ \mathcal{O}_{\Gamma_y^2} \end{bmatrix} \hat{x}^2 + \begin{bmatrix} \mathcal{N}_{\Gamma_u^{\text{cert}} \to \Gamma_y^{\text{temp}}} \\ \mathcal{N}_{\Gamma_u^{\text{cert}} \to \Gamma_y^2} \end{bmatrix} \hat{\mathbf{U}}^2, \tag{A.2}$$

where $\hat{x}^1, \hat{x}^2 \in \mathbb{R}^n$ are states that T-solver.check returns, $\hat{\mathbf{U}}^1, \hat{\mathbf{U}}^2$ are matrices with appropriate dimensions that satisfy TEST. Note that the underlying system is $(2r, 2s)$-sparse strongly observable, $|\Gamma_u^{\text{cert}}| \leq 2r$ and $|\Gamma_y^{\text{temp}}| \geq p - s$ therefore $\hat{S} := (A, B_{(.,\Gamma_u^{\text{cert}})}, C_{(\Gamma_y^{\text{temp}},.)}, D_{(\Gamma_y^{\text{temp}},\Gamma_u^{\text{cert}})})$ is strongly observable. One can reinterpret $(\hat{\mathbf{U}}^1, \mathbf{Y}|_{\Gamma_y^{\text{temp}}})$ and $(\hat{\mathbf{U}}^2, \mathbf{Y}|_{\Gamma_y^{\text{temp}}})$ as two (possibly different) valid trajectories of a strongly observable system $\hat{S}$ with identical output sequences. Strong observability implies that the state can be uniquely determined from the output with a delay bounded by n, therefore $\hat{x}^1 = \hat{x}^2$. Furthermore, the equality of right hand sides of (A.1) and (A.2) implies that,

$$\mathcal{N}_{\Gamma_u^{\text{cert}} \to \Gamma_y^{\text{temp}}}(\hat{\mathbf{U}}^2 - \hat{\mathbf{U}}^1) = 0, \tag{A.3}$$

i.e., $\hat{\mathbf{U}}^2 - \hat{\mathbf{U}}^1$ is a zero dynamic of $\hat{S}$. By $(2r, 2s)$-sparse strongly observablity of S, we conclude that $\hat{\mathbf{U}}^2 - \hat{\mathbf{U}}^1$ is also a zero dynamic of S, and therefore,

$$\mathcal{N}_{\Gamma_u^{\text{cert}} \to \Gamma_y^1}(\hat{\mathbf{U}}^2 - \hat{\mathbf{U}}^1) = 0, \quad \mathcal{N}_{\Gamma_u^{\text{cert}} \to \Gamma_y^2}(\hat{\mathbf{U}}^2 - \hat{\mathbf{U}}^1) = 0. \tag{A.4}$$

Putting (A.1), (A.2) and (A.4) together with $\hat{x}^1 = \hat{x}^2$, we conclude that:

$$\begin{bmatrix} \mathbf{Y}|_{\Gamma_y^{\text{temp}}} \\ \mathbf{Y}|_{\Gamma_y^1} \\ \mathbf{Y}|_{\Gamma_y^2} \end{bmatrix} = \begin{bmatrix} \mathcal{O}_{\Gamma_y^{\text{temp}}} \\ \mathcal{O}_{\Gamma_y^1} \\ \mathcal{O}_{\Gamma_y^2} \end{bmatrix} \hat{x}^1 + \begin{bmatrix} \mathcal{N}_{\Gamma_u^{\text{cert}} \to \Gamma_y^{\text{temp}}} \\ \mathcal{N}_{\Gamma_u^{\text{cert}} \to \Gamma_y^1} \\ \mathcal{N}_{\Gamma_u^{\text{cert}} \to \Gamma_y^2} \end{bmatrix} \hat{\mathbf{U}}^1, \tag{A.5}$$

i.e., $\text{TEST}(\Gamma_u^{\text{cert}}, \Gamma_y^{\text{temp}} \cup \Gamma_y^1 \cup \Gamma_y^2)$ returns false.

□

A.3 Proof of Lemma 2.3

Let us revisit the optimization (2.20) inside the consistency check $\text{TEST}(\Gamma_u, \Gamma_y)$,

$$\min_{\hat{x}, \hat{\mathbf{U}}} \left\| \mathbf{Y}|_{\Gamma_y} - \begin{bmatrix} \mathcal{O}_{\Gamma_y}, \mathcal{N}_{\Gamma_u \to \Gamma_y} \end{bmatrix} \begin{bmatrix} \hat{x} \\ \hat{\mathbf{U}} \end{bmatrix} \right\| \tag{A.6}$$

For a generic LTI system, the matrix $\begin{bmatrix} \mathcal{O}_{\Gamma_y}, \mathcal{N}_{\Gamma_u \to \Gamma_y} \end{bmatrix} \in \mathbb{R}^{n|\Gamma_y| \times n(1+|\Gamma_u|)}$ is of full rank, where n is the order of the LTI system. If $\begin{bmatrix} \mathcal{O}_{\Gamma_y}, \mathcal{N}_{\Gamma_u \to \Gamma_y} \end{bmatrix} \in \mathbb{R}^{n|\Gamma_y| \times n(1+|\Gamma_u|)}$ is of full row rank, then $\text{TEST}(\Gamma_u, \Gamma_y)$ is satisfied irrespectively of the actual values of $\mathbf{Y}|_{\Gamma_y}$. Therefore in order to have a certificate constructed by inputs in $\overline{\Gamma}_u$ and outputs in Γ_y, $\begin{bmatrix} \mathcal{O}_{\Gamma_y}, \mathcal{N}_{\Gamma_u \to \Gamma_y} \end{bmatrix} \in \mathbb{R}^{n|\Gamma_y| \times n(1+|\Gamma_u|)}$ should be a full column rank matrix, therefore

$$n|\Gamma_y| \geq n(1 + |\Gamma_u|). \tag{A.7}$$

The certificate consists of inputs in $\overline{\Gamma}_u$ and outputs in Γ_y, therefore the length of certificate is:

$$|\overline{\Gamma}_u| + |\Gamma_y| = m - |\Gamma_u| + |\Gamma_y| \geq m + 1. \tag{A.8}$$

APPENDIX B

Proofs for Chapter 3

B.1 Proposition B.1

Not all behaviors can be identified solely based on input and output trajectories. In order to identify a behavior, notion of controllability is necessary, see Chapter 8 in [MWVHDM06].

Definition B.1 (Controllablity)**.** *The system $\mathcal{B}$ is controllable if for any two trajectories $\mathbf{w}_1, \mathbf{w}_2 \in \mathcal{B}$, there is a third trajectory $\mathbf{w} \in \mathcal{B}$ and $t_1 \geq 0$, such that $w_1(t) = w(t)$, for all $t < 0$, and $w_2(t) = w(t)$, for all $t \geq t_1$.*

Proposition B.1 (See Theorem 8.16 in [MWVHDM06])**.** *The behavior $\mathcal{B} \in \mathcal{L}^w$ is identifiable from the exact data $\mathbf{w} := (\mathbf{u}, \mathbf{y}) \in \mathcal{B}$ if $\mathcal{B}$ is controllable and $\mathcal{H}_{l(\mathcal{B})+n(\mathcal{B})+1}(\mathbf{u})$ is of full row rank.*

Note that applying Proposition B.1 requires the knowledge of $l(\mathcal{B})$ and $n(\mathcal{B})$ a priori. One should not expect to know these parameters exactly, however, upper bounds assumed to be known, denoted by $l_{\max}$ and $n_{\max}$, respectively. We can then use these bounds in proposition B.1 rather than using the exact values, see chapter 7 in [MWVHDM06].

Putting these together, Assumption 3.2 can be further simplified to the following conditions for LTI systems,

1. The underlying behavior, $\mathcal{B}$, is identifiable, which follows from controllability in the LTI case. See chapter 8, section 8.4 of [MWVHDM06].

2. The complexity of the underlying system is bounded, i.e., $l(\mathcal{B})$ and $n(\mathcal{B})$ are bounded by $l_{\max}$ and $n_{\max}$, respectively.

3. (Persistency of excitation) The input sequence is sufficiently rich to enable the identification of the plant in the absence of attacks, i.e., $\mathcal{H}_{l_{\max}+n_{\max}+1}(\mathbf{u})$ is of full row rank.

B.2 Proof of Lemma 3.1

Given that condition (3.5) holds, there exists an n dimensional subspace defined by $\mathcal{X} := \{(x, x') \in \mathbb{R}^n \times \mathbb{R}^n | x' = Px\}$ which is clearly invariant under the dynamics of the parallel system. For the other direction, let us denote the n dimensional subspace by $\mathcal{X} \subset \mathbb{R}^n \times \mathbb{R}^n$. We need to show that there exists a linear change of coordinates, denoted by $P \in \mathbb{R}^{n \times n}$ and satisfying condition (3.5). First, we show that $\Pi_1 \mathcal{X} = \mathbb{R}^n$ and $\Pi_2 \mathcal{X} = \mathbb{R}^n$, where $\Pi_i : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}^n, \quad i \in \{1, 2\}$ is the projection onto the coordinates corresponding to the i-th system. Let us pick an arbitrary point $x \in \mathbb{R}^n$. We show there exists $x' \in \mathbb{R}^n$ such that $(x, x') \in \mathcal{X}$. Note that (A, B, C, D) is a minimal realization therefore the pair (A, B) is reachable. This implies there exists an input sequence that can drive the state of the first system from 0 to x. This input sequence drives the state of S_2 to some state x'. Given that $(0, 0) \in \mathcal{X}$ and $\mathcal{X}$ is an invariant subspace, we conclude that $(x, x') \in \mathcal{X}$ and $\Pi(x, x') = x$. Note that x was an arbitrary point so it directly follows $\Pi_1 \mathcal{X} = \mathbb{R}^n$. The same exact argument applies to S_2, i.e., $\Pi_2 \mathcal{X} = \mathbb{R}^n$.

Let $\{u_1, \ldots, u_n\}$ be a basis for the subspace $\mathcal{X}$. Since the projections Π_1 and Π_2 are surjective, $\{v_1, \ldots, v_n\} := \Pi_1\{u_1, \ldots, u_n\}$ and $\{w_1, \ldots, w_n\} := \Pi_2\{u_1, \ldots, u_n\}$ are basis for $\Pi_1 \mathcal{X} = \mathbb{R}^n$ and $\Pi_2 \mathcal{X} = \mathbb{R}^n$, respectively. Define P as the linear transformation sending $\{v_1, \ldots, v_n\}$ to $\{w_1, \ldots, w_n\}$. Since $\{v_1, \ldots, v_n\}$ and $\{w_1, \ldots, w_n\}$ are basis, P is well defined and we can represent $\mathcal{X}$ as $\{(x, x') \in \mathbb{R}^n \times \mathbb{R}^n | x' = Px\}$. Given any initial condition in $\mathcal{X}$ denoted by (x, Px), the trajectory should remain in $\mathcal{X}$ even with the zero input sequence. Therefore $A'Px = PAx$ should hold for any $x \in \mathbb{R}^n$, i.e., $A' = PAP^{-1}$. A similar argument applies for the input vector fields starting from the zero initial condition which results in $B' = PB$. We conclude that condition (3.5) holds.

B.3 Proof of Lemma 3.2

- $1 \Longrightarrow 2$:

  Pick an arbitrary minimal realization of S, denoted by $(A, B, [C_1^T, C_2^T]^T, [D_1^T, D_2^T]^T)$. Observability of y_2 from (u, y_1) implies that (A, B, C_1, D_1) is observable. Clearly (A, B, C_1, D_1) is a state-space realization of the system S_Q. Since (A, B, C_1, D_1) is observable and controllable, it is a minimal realization of S_Q. By considering the change of coordinates, $P = I$, Lemma 3.1 implies that S and S_Q are similar modulo outputs.

- $2 \Longrightarrow 1$:

  Pick arbitrary minimal realizations of S and S_Q, denoted by $(A, B, [C_1^T, C_2^T]^T, [D_1^T, D_2^T]^T)$ and (A_Q, B_Q, C_Q, D_Q), respectively. We need to show that (A, B, C_1, D_1) is an observable realization of S_Q, from which directly follows that y_2 is observable from (u, y_1). Lemma 3.1 implies that there exists a linear change of coordinates denoted by P such that $A_Q = PAP^{-1}$ and $B_Q = PB$. S_Q is a quotient of S therefore if both systems start at the zero initial condition, then for any input sequence we have

  $$\begin{bmatrix} y_{S_Q}(t) \\ u(t) \end{bmatrix} = \Pi \begin{bmatrix} y_S(t) \\ u(t) \end{bmatrix}, \tag{B.1}$$

  Using the closed-form expression for the output of a linear system, we can rewrite the previous equality as follow:

  $$\sum_{n=0}^{k-1} C_Q A_Q^{k-1-n} B_Q u(n) + D_Q u(k) = \sum_{n=0}^{k-1} C_1 A^{k-1-n} B u(n) + D_1 u(k), \forall k \in \mathbb{N}. \tag{B.2}$$

  Equation (B.2) holds for all possible input sequences, therefore $D_1 = D_Q$ and $C_Q A_Q^n B_Q = C_1 A^n B$ for all $n \in \mathbb{N}_0$, lemma 3.1 implies that $C_Q P A^n B = C_1 A^n B$ hence for any $k \in \mathbb{N}$ we have

  $$C_Q P[A^k B, A^{k-1} B, \ldots, B] = C_1[A^k B, A^{k-1} B, \ldots, B]. \tag{B.3}$$

  Controllability of the pair (A, B) and (B.3) imply that $C_1 = C_Q P$. Note that $(A, C_1) = (P^{-1} A_Q P, C_Q P)$ and (A_Q, C_Q) is observable, therefore (A, C_1) is an observable pair.

B.4 Proof of Proposition 3.2

Clearly, reflexivity and symmetry hold based on the definition. We only need to show that transitivity holds, i.e., we need to prove that if $S_1 \sim S_2$ and $S_2 \sim S_3$ then $S_1 \sim S_3$. Corollary 3.1 implies that there exist invertible matrices, P_1 and P_2 such that:

$$S_1 \sim S_2 \quad \Rightarrow \begin{cases} A_2 = P_1 A_1 P_1^{-1}, \\ B_2 = P_1 B_1 \end{cases} \tag{B.4}$$

$$S_2 \sim S_3 \quad \rightarrow \begin{cases} A_3 = P_2 A_2 P_2^{-1}, \\ B_3 = P_2 B_2 \end{cases} \tag{B.5}$$

Combining (B.4) and (B.5) it is easy to verify that the linear change of coordinates $P_2 P_1$ makes S_1 and S_3 similar modulo outputs.

B.5 Proof of Lemma 3.3

Note that $\mathbb{U}^{l_{\max}+1}$, $\mathbb{Y}_1^{l_{\max}+1}$ and $\mathbb{Y}_2$ are all finite dimensional spaces, therefore existence of these linear mappings is equivalent to the existence of matrices L_u and L_{y_1} such that

$$y_2(t) = \begin{bmatrix} L_u & L_{y_1} \end{bmatrix} \begin{bmatrix} u(t-l_{\max}) \\ \vdots \\ u(t) \\ y_1(t-l_{\max}) \\ \vdots \\ y_1(t) \end{bmatrix}, \quad \forall t \in \{l_{\max}, \ldots, T-1\}.$$

We can represent these linear constraints as:

$$\Big[y_2(l_{\max}),\ldots,y_2(T-1)\Big] = \begin{bmatrix} L_u & L_{y_1} \end{bmatrix} \times \underbrace{\begin{bmatrix} u(0) & u(1) & \ldots & u(T-l_{\max}-1) \\ \vdots & \vdots & \vdots & \vdots \\ u(l_{\max}) & u(l_{\max}+1) & \ldots & u(T-1) \\ y_1(0) & y_1(1) & \ldots & y_1(T-l_{\max}) \\ \vdots & \vdots & \vdots & \vdots \\ y_1(l_{\max}) & y_1(l_{\max}+1) & \ldots & y_1(T-1) \end{bmatrix}}_{\mathcal{H}_{l_{\max}+1}(\mathbf{u},\mathbf{y}_1)}, \tag{B.6}$$

therefore we conclude that L_u and L_{y_1} exist if and only if

$$\Big[y_2(l_{\max}-1),\ldots,y_2(T-1)\Big] \in \textbf{Row Space}\big(\mathcal{H}_{l_{\max}}(\mathbf{u},\mathbf{y})\big). \tag{B.7}$$

APPENDIX C

Subspace Identification

In this appendix, we briefly review subspace identification methods. In the literature of system identification, there exist several methods that guarantee exact identification when the length of the training sequence tends to infinity (see for example [VODM12, Lju87]). Subspace identification algorithms are one of the most prominent such methods. We briefly review the preliminaries followed by statement of the main results in this line of work.

Projections. Assume that matrices $A \in \mathbb{R}^{p\times j}$, $B \in \mathbb{R}^{q\times j}$ and $C \in \mathbb{R}^{r\times j}$ are given. One can think of the rows of each matrix as the coordinates of a vector in the j-dimensional ambient space, therefore rows of each of A, B and C can be considered as a basis for a linear vector space in this ambient space. Now we can define the orthogonal projection of row space of one matrix onto other one,

$$\begin{aligned} A/\mathbf{B} &:= A.\Pi_B, \\ \Pi_B &:= B^T.(BB^T)^\dagger.B \end{aligned} \tag{C.1}$$

where $(.)^\dagger$ denotes the Moore-Penrose pseudo-inverse of the matrix $(.)$. The operator Π_B projects the row space of A onto the row space of B. It can be shown that,

$$A = A/\mathbf{B} + A/\mathbf{B}^\perp, \tag{C.2}$$

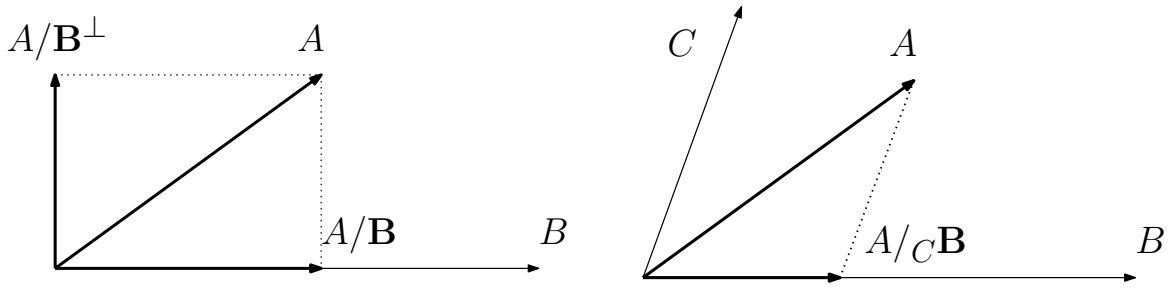
where $A/\mathbf{B}^\perp$ denotes the orthogonal projection of row space of A onto the orthogonal complement of the row space of matrix B.

Instead of decomposing A as linear combinations of two orthogonal matrices (B and $B^\perp$), one can think of decomposing A as a linear combination of non-orthogonal matrices B and

C and of the orthogonal complement of B and C. In the same way, we can define the oblique projection of the row space of A along the row space of C on the row space of B by,

$$A/_C\mathbf{B} := A\begin{bmatrix} B^T & C^T \end{bmatrix}.\Big(\begin{bmatrix} BB^T & BC^T \\ CB^T & CC^T \end{bmatrix}^{\dagger}\Big)_{\text{first } j \text{ colmuns}}.B. \tag{C.3}$$

It can be shown that $A/_C\mathbf{B} = (A/\begin{bmatrix} \mathbf{B} \\ \mathbf{C} \end{bmatrix})/\mathbf{B}$. Figure C.1 gives an interpretation of these projection operators for 2-dimensional ambient space.



(a) Interpretation of Orthogonal projection in the j- dimensional space ($j = 2$).

(b) Interpretation of Oblique projection in the j- dimensional space ($j = 2$).

Figure C.1: An illustration of the Orthogonal and Oblique projections in 2-dimensional ambient space.

In subspace identification methods we typically assume that a long sequence of data is available and that the data is ergodic. Consider input and noise sequences denoted by $\{u_k\}_{k=0}^{j} \in \mathbb{R}^n$ and $\{e_k\}_{k=0}^{j} \in \mathbb{R}^p$, respectively. Noise sequence $\{e_k\}$ is a zero-mean sequence and independent of $\{u_k\}$, i.e.,

$$\begin{aligned} \mathbb{E}[e_k] &= 0, \\ \mathbb{E}[u_k e_k^T] &= 0. \end{aligned} \tag{C.4}$$

Due to ergodicity and long sequence of data points, we can replace the expectation operator with different operator $\mathbb{E}_{\mathbf{j}}[.]$ which essentially is the average over time for only one experiment (sample path of the processes),

$$\mathbb{E}_j[.] := \lim_{j\to\infty} \frac{1}{j}[.]. \tag{C.5}$$

Ergodicity implies that $\mathbb{E}[u_k e_k^T] = \lim_{j\to\infty} \frac{1}{j}\sum_{i=0}^{j}[a_i e_i^T] = \mathbb{E}_j[u.e^T]$, where u and e are row-vectors representing sequences $\{u_k\}_{k=0}^{j}$ and $\{e_k\}_{k=0}^{j}$, respectively. In subspace identification, we use this implication to overcome the effect of noise, $\mathbb{E}_j[u.e^T] = 0$ i.e., the row space of input is orthogonal to the row space of noise with respect to this operator. This property lies at the heart of subspace identification methods for the noisy scenario to get rid of the effect of disturbances.

Now we are ready to relate statistical assumptions to geometric properties. In the statistical framework we define the projection using the operator $\mathbb{E}_j[.]$. We define the covariance between two matrices as:

$$\Phi_{[A,B]} := \mathbb{E}_j[A.B^T] \tag{C.6}$$

We can extend the geometric tools introduced for deterministic matrices to stochastic ones by replacing the inner product $A.B^T$ by the $\Phi_{[A,B]}$ in (C.1) and (C.3), i.e.,

$$A/\mathbf{B} = \Phi_{[A,B]}.\Phi_{[A,B]}^{\dagger}.B, \tag{C.7}$$

$$A/_C\mathbf{B} = \begin{bmatrix}\Phi_{[A,B]} & \Phi_{[A,C]}\end{bmatrix}.\Big(\begin{bmatrix}\Phi_{[B,B]} & \Phi_{[B,C]}\\ \Phi_{[C,B]} & \Phi_{[C,C]}\end{bmatrix}^{\dagger}\Big)_{\text{first } j \text{ colmuns}}.B. \tag{C.8}$$

Now we formally introduce conditions under which Subspace identification methods are guaranteed to identify the underlying LTI system. As it was discussed in the Assumption 3.2, we say the time-series $\mathbf{u}$ is persistently exciting of order i if $\mathcal{H}_i(\mathbf{u})$ is of full row rank, i.e., $\text{rank}(\mathcal{H}_i(\mathbf{u})) = m.i$, where m is the dimension of the signal space.

Definition C.1 (Quasi stationary, see page 27 of [Lju87])**.** *For a deterministic sequence* $\mathbf{u}$*, quasi-stationary means that it is a bounded sequence such that the limits,*

$$R_u(\tau) = \lim_{N\to\infty}\frac{1}{N}\sum_{t=\tau}^{N}\mathbf{u}(t)\mathbf{u}(t-\tau) \tag{C.9}$$

exist.

The following theorem states the main result for subspace identification algorithms.

Theorem C.1. *(See Theorem 2 in Ch. 2 and Theorem 12 in Ch. 4 [VODM12]) Let* $\{u_k\}_{k=0}^{j}$ *and* $\{y_k\}_{k=0}^{j}$ *denote the input and output sequences. We partition the input (output) Hankel matrix into Future Input,* U_f *(Future Output,* Y_f*) and Past Input,* U_p *(Past Output,* U_f*), i.e.,*

$$\mathcal{H}_{2i}(\mathbf{u}) = \begin{bmatrix} U_p \\ U_f \end{bmatrix}, \quad \mathcal{H}_{2i}(\mathbf{y}) = \begin{bmatrix} Y_p \\ Y_f \end{bmatrix}. \tag{C.10}$$

Under the assumptions that:

1. *The deterministic input is uncorrelated with measurement noise and is quasi stationary.*
2. *The input sequence is persistently exciting of order* $2i$*.*
3. *The length of available data points goes to infinity, i.e.,* $j \to \infty$
4. *Row space of future inputs (*U_f*) does not intersect with the row space of the past states.*
5. *The user-defined square weighting matrices* W_1 *and* W_2 *are such that* W_1 *is of full rank and* W_2 *obeys:* $rank(W_p) = rank(W_pW_2)$*, where* W_p *is the block Hankel matrix containing the* U_p *and* Y_f*.*

And with $\mathcal{O}_i$ *defined as the oblique projection:*

$$\mathcal{O}_i := Y_f/_{U_f}\mathbf{W}_p, \tag{C.11}$$

and the singular value decomposition:

$$\mathcal{O}_i = \begin{bmatrix} U_1 & U_2 \end{bmatrix} \begin{bmatrix} S_1 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} V_1^T & V_2^T \end{bmatrix}, \tag{C.12}$$

Then we have:

1. *The matrix* $\mathcal{O}_i$ *is equal to the product of extended observability matrix and the Kalman*

filter state sequence $\tilde{X}$*:*

$$\mathcal{O}_i = \Gamma_i \tilde{X}_i, \quad \Gamma_i := \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{i-1} \end{bmatrix}, \tag{C.13}$$

2. *The order of the underlying system is equal to the rank of* $\mathcal{O}_i$*.*

3. *The extended observability matrix* Γ_i *is equal to:*

$$\Gamma_i = W_1^{-1} U_1 S_1^{1/2}.T, \tag{C.14}$$

where $T \in \mathbb{R}^{n \times n}$ *is an arbitrary similarity transformation.*

4. *The state sequence* $\tilde{X}$ *is equal to:*

$$\tilde{X} = \Gamma_i^{\dagger} \mathcal{O}_i. \tag{C.15}$$

It can be shown that three subspace algorithms of the literature (**N4SID**, **MOESP** and **CVA**) are special cases of Theorem C.1 depending on the specific weighting matrices W_1 and W_2 [VODM95]. Note that, for the secure system identification, any of the above algorithms would work and the user can choose any of the aforementioned methods.

It is worth mentioning that Theorem C.1 requires an infinite amount of data, however in practical applications such a sequence is not available and we need to approximate the operator $\mathbb{E}_j$ by a finite average over the available data points.

APPENDIX D

Proofs for Chapter 4

D.1 Proof of Theorem 4.1

In order to derive bounds for the utility performance of additive noise, we use the perturbation theory in the least square setup [Wed73]. For a given N and σ_{AN} we have the following deterministic bound on the utility,

Lemma D.1 (See Theorem 5.1 in [Wed73]). *Assuming rank(X) = rank$(X + \sigma_{AN}N) = d$ and $\kappa(X)\Delta(\epsilon, N, X) < 1$:*

$$\frac{\|X\theta_{AN} - y\|_2}{\|X\theta^\star - y\|_2} \leq 1 + \frac{\kappa(X)\Delta(\epsilon, N, X)}{1 - \kappa(X)\Delta(\epsilon, N, X)}(\kappa(X) + r(y)), \tag{D.1}$$

where $\Delta(\epsilon, N, X) = \sigma_{AN}\frac{\|N\|_2}{\|X\|_2}$.

It is well-known that the maximum singular value of $N \in \mathbb{R}^{n\times d}$ converges almost surely to $\sqrt{n} + \sqrt{d}$ asymptotically. For the non-asymptotic bounds, we use the following lemma:

Proposition D.1 (See [RV10]). *If $N \in \mathbb{R}^{n\times d}$ is a Gaussian random matrix with entries drawn from $\mathcal{N}(0, 1)$, then*

$$P(\sigma_{\max}(N) \leq \sqrt{n} + \sqrt{d} + t) \geq 1 - 2e^{-\frac{t^2}{2}}, \quad t \geq 0. \tag{D.2}$$

By combining Lemma D.1 and Proposition D.1 and the choice of σ_{AN} (D.1), Theorem 4.1 directly follows.

D.2 Proof of Theorem 4.2

In this section, we derive utility guarantee on the performance of random projection for the given value of σ_{RP}. Note that by rewriting $X_{RP} = \frac{1}{\sqrt{n'}} \begin{bmatrix} S & N \end{bmatrix} \begin{bmatrix} X \\ \sqrt{n'}\sigma_{RP} I \end{bmatrix} = \tilde{S} \begin{bmatrix} X \\ \sqrt{n'}\sigma_{RP} I \end{bmatrix}$ we observe that adding direct noise to the projected data can be interpreted as the random projection of the l_2 regularized least square problem (Ridge Regression), i.e.,

$$\theta_{RP} = \arg\min_{\theta} \|X_{RP}\theta - y_{RP}\|_2^2 \tag{D.3}$$

$$= \arg\min_{\theta} \|\tilde{S} \underbrace{\Big(\begin{bmatrix} X \\ \sqrt{n'}\sigma_{RP} I \end{bmatrix} \theta - \begin{bmatrix} y \\ 0 \end{bmatrix} \Big)}_{\text{RR}} \|_2^2, \tag{D.4}$$

Let us denote the solution to the Ridge Regression problem with $\theta_{RR} = \arg\min_{\theta} \|X\theta - y\|^2 + n'\sigma_{RP}^2\|\theta\|^2$, therefore we can write:

$$\begin{aligned} \frac{\|X\theta_{RP} - y\|_2^2}{\|X\theta^\star - y\|_2^2} = & \underbrace{\frac{\|X\theta_{RR} - y\|_2^2}{\|X\theta^\star - y\|_2^2}}_{\eta_1} \\ & \times \underbrace{\frac{\|X\theta_{RR} - y\|_2^2 + n'\sigma_{RP}^2\|\theta_{RR}\|_2^2}{\|X\theta_{RR} - y\|_2^2}}_{\eta_2} \\ & \times \underbrace{\frac{\|X\theta_{RP} - y\|_2^2 + n'\sigma_{RP}^2\|\theta_{RP}\|_2^2}{\|X\theta_{RR} - y\|_2^2 + n'\sigma_{RP}^2\|\theta_{RR}\|_2^2}}_{\eta_3} \\ & \times \underbrace{\frac{\|X\theta_{RP} - y\|_2^2}{\|X\theta_{RP} - y\|_2^2 + n'\sigma_{RP}^2\|\theta_{RP}\|_2^2}}_{\eta_4}. \end{aligned} \tag{D.5}$$

It is clear that $\eta_4 < 1$, therefore we find bounds for each of η_1, η_2 and η_3.

Using the following Lemma, $\eta_1 \leq \left(1 + \frac{n'\sigma_{RP}^2}{\sigma_{\min}^2 + n'\sigma_{RP}^2} r(y)\right)^2$.

Lemma D.2. *Let us denote the solution to the l_2 regularized least square problem with $\theta_{RR}(\lambda) := \arg\min_{\theta} \|X\theta - y\|_2^2 + \lambda\|\theta\|_2^2$ and $\theta^\star = \arg\min_{\theta} \|X\theta - y\|_2^2$, then we have the following bound on the empirical risk loss given that X is full rank:*

$$\frac{\|X\theta_{RR}(\lambda) - y\|_2}{\|X\theta^\star - y\|_2} \leq 1 + \frac{\lambda}{\sigma_{min}^2 + \lambda} r(y). \tag{D.6}$$

Proof. Using triangle inequality we can write the LHS of (D.6):

$$\frac{\|X\theta^\star - y + X(\theta_{RR}(\lambda) - \theta^\star)\|_2}{\|X\theta^\star - y\|_2} \leq 1 + \frac{\|X(\theta_{RR}(\lambda) - \theta^\star)\|_2}{\|X\theta^\star - y\|_2}. \tag{D.7}$$

Let us denote the SVD decomposition of X by $X = U\Sigma V^T$, where $U \in \mathbb{R}^{n\times d}$ spans the column space, $\Sigma \in \mathbb{R}^{d\times d}$ is the diagonal matrix of the singular values and $V \in \mathbb{R}^{d\times d}$ spans the row space of X. We use the close form solution for $\theta^\star$ and θ_{RR} to derive bounds for $\|X(\theta_{RR}(\lambda) - w^\star)\|_2$.

$$\theta^\star = (X^TX)^{-1}X^Ty = V\Sigma^{-1}U^Ty \tag{D.8}$$

$$\theta_{RR} = (X^TX + \lambda I)^{-1}X^Ty = V(\Sigma^2 + \lambda I)^{-1}\Sigma U^Ty, \tag{D.9}$$

therefore

$$\begin{aligned}
\|X(\theta_{RR}(\lambda) - \theta^\star)\|_2 &= \|U \underbrace{\Sigma[(\Sigma^2 + \lambda I)^{-1} - \Sigma^{-2}]\Sigma}_{-D} U^Ty\|_2 \\
&\leq \|UDU^Ty\|_2 \\
&\overset{(a)}{\leq} \sigma_{\max}(D)\|U^Ty\|_2 \\
&\overset{(b)}{=} \left(\frac{\lambda}{\sigma_{\min}^2 + \lambda}\right)\|X\theta^\star\|_2.
\end{aligned} \tag{D.10}$$

(a) and (b) follow directly since D is a diagonal matrix with i-th entry of $\frac{\lambda}{\sigma_i^2+\lambda}$, where σ_i is i-th singular value of X so $\sigma_{\max}(D) \leq \frac{\lambda}{\sigma_{\min}^2+\lambda}$. By combining (D.7) and (D.10), (D.6) follows directly. □

Corollary D.1. *Let us denote the solution to the l_2 regularized least square problem with $\theta_{RR}(\lambda) := \arg\min_\theta \|X\theta - y\|_2^2 + \lambda\|\theta\|_2^2$ and $\theta^\star = \arg\min_\theta \|X\theta - y\|_2^2$, we have the following bound on the norm of the θ_{RR}:*

$$\|\theta_{RR}\|_2 \leq \left(\max_i \frac{\sigma_i}{\sigma_i^2 + \lambda}\right)\|X\theta^\star - y\|_2, \tag{D.11}$$

Proof. Proof directly follows by using the closed form solution for θ_{RR},

$$\begin{aligned}
\|\theta_{RR}\| &= \|V(\Sigma^2 + \lambda I)^{-1}\Sigma U^T y\|_2 \\
&= \|\underbrace{(\Sigma^2 + \lambda I)^{-1}}_{D'}\Sigma U^T y\|_2, \\
&\leq \sigma_{\max}(D')\|U^T y\|_2 \qquad \text{(D.12)} \\
&= \left(\max_i \frac{\sigma_i}{\sigma_i^2 + \lambda}\right)\|X\theta^\star\|_2. \qquad \text{(D.13)}
\end{aligned}$$

□

By Corollary D.1 we have the following bound on η_2:

$$\eta_2 \leq 1 + n'\sigma_{RP}^2\left(\max_i \frac{\sigma_i}{\sigma_i^2 + n'\sigma_{RP}^2}\right)^2 r^2(y), \qquad \text{(D.14)}$$

We use results of Pilanci et. al. [PW15] for η_3:

Proposition D.2 (See Corollary 2 in [PW15])**.** *Suppose* $\theta_{RP} = \arg\min_\theta \|\tilde{\mathbf{S}}\big(\begin{bmatrix} X \\ \sqrt{n'}\sigma_{RP} I \end{bmatrix}\theta - \begin{bmatrix} y \\ 0 \end{bmatrix}\big)\|_2^2$ *and* $\theta_{RR} := \arg\min_\theta \|X\theta - y\|_2^2 + n'\sigma_{RP}^2\|\theta\|_2^2$ *where* $\tilde{\mathbf{S}} \in \mathbb{R}^{n'\times n}$ *is a random Gaussian matrix with entries drawn from* $\mathcal{N}(0,1)$*. With probability at least* $1 - c_1 e^{-c_2 n'\delta^2}$ *for* $\delta \geq \sqrt{c_0\frac{d}{n'}}$*:*

$$\frac{\|X\theta_{RP} - y\|_2^2 + n'\sigma_{RP}^2\|\theta_{RP}\|_2^2}{\|X\theta_{RR} - y\|_2^2 + n'\sigma_{RP}^2\|\theta_{RR}\|_2^2} \leq (1+\delta)^2, \qquad \text{(D.15)}$$

where c_0*,* c_1 *and* c_2 *are constants.*

Result of Theorem 4.2 follows directly by using bounds on η_1, η_2 and η_3.

APPENDIX E

Proofs for Chapter 5

E.1 Proof of Proposition 5.3

Recall from (5.8) that $z_i(t)$ is a function of $\{\mathbf{y}(t)\}_{t=1}^t$, therefore conditioned on the same set of messages $\{\mathbf{y}(t)\}_{t=1}^t$ we have the following ,

$$\begin{aligned} &\|x_i^D(t) - x_i^{D'}(t)\|_2 \\ &= \|\mathrm{Proj}_{\mathcal{X}}\left(z_i^D(t) - \eta_t \nabla f_i^D(z_i^D(t))\right) \\ &\qquad\qquad - \mathrm{Proj}_{\mathcal{X}}\left(z_i^{D'}(t) - \eta_t \nabla f_i^{D'}(z_i^{D'}(t))\right)\| \\ &\overset{(a)}{\leq} \eta_t \|\nabla f_i^D(z_i^D(t)) - \nabla f_i^{D'}(z_i^{D'}(t))\| \overset{(b)}{\leq} 2\eta_t G, \end{aligned} \tag{E.1}$$

where (a) follows from non-expansiveness of the projection operator and Assumption 5.2 implies (b). Taking the supermom from both sides of (E.1) implies the result directly.

E.2 Proof of Theorem 5.1

In order to prove Algorithm 8 satisfies (ϵ, δ)-DP, we check condition (5.14) in Proposition 5.1. Recall that the adversary can only observe messages among nodes and the privacy loss random variable is a function of these observations. We derive an analytical expression for the privacy loss random variable and bound it using concentration inequalities.

Note that we can write the pdf of $\mathbf{y}(1), \cdots, \mathbf{y}(T)$ as follows:

$$\mathrm{pdf}_D(\mathbf{y}(1), \cdots, \mathbf{y}(T)) = \prod_t \mathrm{pdf}_D(\mathbf{y}(t+1) | \mathbf{y}(1), \cdots, \mathbf{y}(t))$$

$$\stackrel{(a)}{=} \prod_t \mathrm{pdf}_D(\mathbf{y}(t+1)|\mathbf{y}(t))$$
$$\stackrel{(b)}{=} \prod_t \prod_{k\in[N]} \mathrm{pdf}_D(y_k(t+1)|\mathbf{y}(t))$$
$$\stackrel{(c)}{=} \prod_t \prod_{k\in[N]} p_{M_t}(y_k(t+1) - x_k(t))),$$

where (a) follows since the randomness comes from the additive noise (5.11) and $\mathbf{y}(t+1)$ conditioned on $\mathbf{n}(t)$ (and therefore $\mathbf{y}(t)$) is independent of all the previous coin tosses of the algorithm. Noise injected independently across nodes which implies (b). In (c), we wrote the conditional pdf of $y_k(t+1)$ in terms of density function of a Gaussian, p_{M_t}.

Let us denote the privacy loss random variable for T stages with $c(\mathbf{y}(1), \cdots, \mathbf{y}(T))$. We distinguish the variables associated with neighboring database D' with $'$. In the context of this problem, neighboring databases differ in at most one data point, *i.e.*, at most one node may have a different function. We denote this node with $k^* \in [N]$.

$$c(\mathbf{y}(1), \cdots, \mathbf{y}(T)) = \log \frac{\mathrm{pdf}_D(\mathbf{y}(1), \cdots, \mathbf{y}(T))}{\mathrm{pdf}_{D'}(\mathbf{y}(1), \cdots, \mathbf{y}(T))} \tag{E.2}$$
$$= \sum_t \sum_{k\in[N]} \frac{-\|y_k(t+1) - x_k(t)\|^2}{2M_t^2} + \frac{\|y_k(t+1) - x'_k(t)\|^2}{2M_t^2}$$
$$= \sum_t \sum_{k\in[N]} \frac{\|x_k(t) - x'_k(t)\|^2}{2M_t^2} + \frac{2\langle y_k(t+1) - x_k(t), x_k(t) - x'_k(t)\rangle}{2M_t^2}$$
$$= \sum_t \sum_{k\in[N]} \frac{\|x_k(t) - x'_k(t)\|^2}{2M_t^2} + \sum_t \sum_{k\in[N]} \langle n_k(t), \frac{(x_k(t) - x'_k(t))}{M_t^2}\rangle$$
$$\stackrel{(a)}{=} \sum_t \frac{\|x_{k^*}(t) - x'_{k^*}(t)\|^2}{2M_t^2} + \sum_t \langle n_{k^*}(t), \frac{(x_{k^*}(t) - x'_{k^*}(t))}{M_t^2}\rangle,$$

where (a) follows because only one of the cost functions is different among neighboring databases, therefore at most one of the these terms is non-zero (note that we are conditioning on the same observations $\{\mathbf{y}(t)\}$ across two neighboring problems).

Recall from the definition of $\Delta^2(t)$ (5.13),

$$\sum_t \frac{\|x_k(t) - x'_k(t)\|^2}{M^2(t)} \leq \sum_t \frac{\Delta^2(t)}{M^2(t)} \triangleq \alpha, \tag{E.3}$$

therefore by putting (E.2) and (E.3) together,

$$c(\mathbf{y}(1), \cdots, \mathbf{y}(T)) \leq \frac{\alpha}{2} + \sum_{t=1}^{T} \langle n_{k^*}(t), \frac{(x_{k^*}(t) - x'_{k^*}(t))}{M_t^2} \rangle. \tag{E.4}$$

In order to bound c we first show the second term in (E.4) $C_T \triangleq \sum_{t=1}^{T} \langle n_{k^*}(t), \frac{(x_{k^*}(t) - x'_{k^*}(t))}{M_t^2} \rangle$ is sub-Gaussian.

Definition E.1 (sub-Gaussian)**.** *A zero mean random variable X is sub-Gaussian*[1] *if for some $\sigma > 0$,*

$$\mathbb{E}e^{\lambda X} \leq e^{\sigma^2 \lambda^2 / 2}, \forall \lambda \in \mathbb{R}. \tag{E.5}$$

Lemma E.1. *The second term in* (E.4)*, C_T is sub-Gaussian with the parameter bounded by $\sqrt{\alpha}$, where α is defined in* (E.3)*.*

Proof. Recall the definition of C_T and let us define w_t as follows:

$$C_T \triangleq \sum_{t=1}^{T} \langle n_{k^*}(t), \frac{(x_{k^*}(t) - x'_{k^*}(t))}{M_t^2} \rangle,$$
$$w_t \triangleq \mathbb{E}[C_T | \mathcal{F}_t] - \mathbb{E}[C_T | \mathcal{F}_{t-1}] = \langle n_{k^*}(t), \frac{(x_{k^*}(t) - x'_{k^*}(t))}{M_t^2} \rangle,$$

where $\mathcal{F}_t$ is the sigma algebra generated by $n_k(1), \cdots, n_k(t)$. Note that w_t is conditionally sub-Gaussian with $\sigma_t \leq \Delta(t)/M_t$ since:

$$\begin{aligned} \mathbb{E}[e^{\lambda w_t} | n_k(1), \cdots, n_k(t-1)] &\overset{(a)}{=} e^{\frac{\|x_k(t) - x'_k(t)\|^2}{M_t^2} \lambda^2 / 2} \\ &\overset{(b)}{\leq} e^{\frac{\Delta^2(t)}{M_t^2} \lambda^2 / 2}, \quad \forall \lambda \in \mathbb{R} \end{aligned} \tag{E.6}$$

here (a) follows since $n_k(t)$ is an i.i.d. zero mean random Gaussian vector with variance of M_t^2 and it's independent of $\mathcal{F}_{t-1}$, and definition of $\Delta^2(t)$ implies (b). Taking the conditional expectation of C_T with respect to $\mathcal{F}_{t-1}$ implies,

$$\mathbb{E}[e^{\lambda C_T} | \mathcal{F}_{T-1}] = e^{\lambda \sum_{t=1}^{T-1} w_t} \mathbb{E}[e^{\lambda w_T} | \mathcal{F}_{T-1}]$$

[1] *There exists several equivalent definitions of sub-Gaussianity in the literature, see for example [Ver18] Proposition 2.5.2*

$$\overset{(a)}{\leq} e^{\lambda \sum_{t=1}^{T-1} w_t} e^{\frac{\Delta^2(T)}{M_T^2}\lambda^2/2}, \forall \lambda \in \mathbb{R} \tag{E.7}$$

where (a) follows from (E.6). By repeating the same argument and taking the conditional expectation with respect to $\mathcal{F}_{T-1}$, $\mathcal{F}_{T-2}$ up until $\mathcal{F}_1$ we conclude that,

$$\begin{aligned}\mathbb{E}[e^{\lambda C_T}] &\leq e^{\sum_{t=1}^{T} \frac{\Delta^2(t)}{M_t^2}\lambda^2/2} \\ &\leq e^{\alpha\lambda^2/2}, \quad \forall \lambda \in \mathbb{R}.\end{aligned}$$

□

In order to bound the privacy random variable, we use the following tail bound for sub-Gaussian random variables.

Proposition E.1. *Assume X is a random variable satisfying* (E.5)*. Then we have*

$$\mathbb{P}\left(X > t\right) \leq e^{-\frac{t^2}{2\sigma^2}}, \forall t \geq 0, \tag{E.8}$$

Proof. Proof follows directly by applying Chernoff bound along with condition in (E.5). □

Now we are ready to prove the claim. We show that $\mathbb{P}\left(|c| \geq \epsilon\right) \leq \delta$ where $c \overset{\triangle}{=} c(\mathbf{y}(1), \cdots, \mathbf{y}(T))$ as in (E.2). Note that,

$$\mathbb{P}\left(c \geq +\epsilon\right) \overset{(a)}{\leq} \mathbb{P}\left(\frac{\alpha}{2} + C_T \geq \epsilon\right) \overset{(b)}{\leq} e^{-(\epsilon - \frac{\alpha}{2})^2/2\alpha}, \tag{E.9}$$

$$\mathbb{P}\left(c \leq -\epsilon\right) \overset{(c)}{\leq} \mathbb{P}\left(C_T \leq -\epsilon\right) \overset{(d)}{\leq} e^{-\frac{\epsilon^2}{2\alpha}}, \tag{E.10}$$

where (a) follows from inequality (E.4), (b) is the direct consequence of Proposition E.1, (c) comes from $c \geq C_T$ and we use the fact that $-C_T$ is a sub-Gaussian random variable along with Proposition 5.3 in (d). Therefore,

$$\mathbb{P}\left(|c| \geq \epsilon\right) = \mathbb{P}\left(c \geq \epsilon\right) + \mathbb{P}\left(c \leq -\epsilon\right) \leq 2e^{-\frac{(\epsilon - \frac{\alpha}{2})^2}{2\alpha}} \overset{(a)}{\leq} \delta. \tag{E.11}$$

It is straightforward to verify that (a) holds for

$$0 \leq \alpha \leq 2\epsilon + 4\log\frac{2}{\delta} - 2\sqrt{\left(\epsilon + 2\log\frac{2}{\delta}\right)^2 - \epsilon^2} \tag{$*$}$$

By using the inequality $a/2 \leq 1 - \sqrt{1-a}$ for $|a| \leq 1$ we conclude that an specific choice of $\alpha = \frac{\epsilon^2}{\epsilon + 2\log\frac{2}{\delta}}$ in the statement of Theorem 5.1 lies in $(*)$.

E.3 Proof of Lemma 5.1

Compared to the classical DGD (see for example [YLY16, NO09]) we have an additional projection operator that we need to take into account. In this case, with a minor modification we can still bound the distance of individual's z_i to the average parameter. Let us rewrite $x_i(t)$ as follows:

$$\begin{aligned} x_i(t) = \mathrm{Proj}_{\mathcal{X}}\left(z_i(t) - \eta_t \nabla f_i(z_i(t))\right) &= z_i(t) + v_i(t) \\ &= \hat{z}_t(t) + u_i(t) + v_i(t), \end{aligned} \tag{E.12}$$

where $\hat{z}_i(t) \stackrel{\triangle}{=} \sum_{j\in\mathcal{N}_i} w_{ij} y_j(t)$ and $u_i(t)$ and $v_i(t)$ are defined as

$$v_i(t) \stackrel{\triangle}{=} \mathrm{Proj}_{\mathcal{X}}\left(z_i(t) - \eta_t \nabla f_i(z_i(t))\right) - z_i(t), \tag{E.13}$$

$$u_i(t) \stackrel{\triangle}{=} z_i(t) - \hat{z}_i(t). \tag{E.14}$$

We use the notation $\hat{\mathbf{z}}(t) = [\hat{z}_1(t); \cdots, \hat{z}_N(t)]$, $\mathbf{u}(t) = [u_1(t); \cdots; u_N(t)]$ and $\mathbf{v}(t) = [v_1(t); \cdots; v_N(t)]$. In the rest of proof, we rewrite the mean parameter using Kronecker product $\bar{z}(t) = \frac{1}{N}\left(\mathbf{1}_N^T \otimes I_P\right)\mathbf{z}(t)$. In order to bound $\|\mathbf{z}(t) - \frac{1}{N}\left(\mathbf{1}_N\mathbf{1}_N^T \otimes I_p\right)\mathbf{z}(t)\|$, we first present a lemma that bounds $\|\hat{\mathbf{z}}(t) - \frac{1}{N}\left(\mathbf{1}_N\mathbf{1}_N^T \otimes I_p\right)\hat{\mathbf{z}}(t)\|$.

Lemma E.2. *Under Assumption 5.2, at any time $t \leq T$, the following holds:*

$$\begin{aligned} &\|\hat{\mathbf{z}}(t) - \frac{1}{N}\left(\mathbf{1}_N\mathbf{1}_N^T \otimes I_p\right)\hat{\mathbf{z}}(t)\| \\ &\qquad \leq 2\sum_{s=1}^{t-1}\beta^{t-s}\|\mathbf{n}(s)\| + \sqrt{N}G\sum_{s=1}^{t-1}\eta_s\beta^{t-s}. \end{aligned} \tag{E.15}$$

Proof. Observer that plugging (E.12) into (5.9) we can write:

$$\begin{aligned}\hat{\mathbf{z}}(t) = (W \otimes I_p)\,(\hat{\mathbf{z}}(t-1) &+ \mathbf{u}(t-1) + \mathbf{v}(t-1)) \\ &+ (W \otimes I_p)\,\mathbf{n}(t-1),\end{aligned} \tag{E.16}$$

where $\mathbf{u}$ and $\mathbf{v}$ are defined according to (E.14) and (E.13) respectively.

We proceed by bounding each of $\mathbf{u}(t)$ and $\mathbf{v}(t)$ for $t \leq T$. It is clear that both terms are zero in Stage II of the algorithm. **Bounding u:** Note that

$$\begin{aligned}\|v_i(t)\|_2 &= \|\mathrm{Proj}_{\mathcal{X}}\left(z_i(t) - \eta_t \nabla f_i(z_i(t))\right) - z_i(t)\| \\ &\overset{(a)}{=} \|\mathrm{Proj}_{\mathcal{X}}\left(z_i(t) - \eta_t \nabla f_i(z_i(t))\right) - \mathrm{Proj}_{\mathcal{X}} z_i(t)\| \\ &\overset{(b)}{\leq} \|\eta_t \nabla f_i(z_i(t))\|_2 \\ &\overset{(c)}{\leq} G\eta_t,\end{aligned}$$

where (a) holds since $z_i(t) \in \mathcal{X}$, (b) follows from non-expansiveness property of the Euclidean projection and Assumption 5.2 implies (c). Therefore,

$$\|\mathbf{v}(t)\|_2 = \sqrt{\sum_i^N \|v_i(t)\|_2^2} \leq G\sqrt{N}\eta_t. \tag{E.17}$$

In order to bound $\mathbf{u}(t)$,

$$\begin{aligned}\|u_i(t)\|_2 = \|z_i - \hat{z}_i\|_2 &= \|\mathrm{Proj}_{\mathcal{X}} \hat{z}_i - \hat{z}_i\|_2 \\ &\overset{(a)}{\leq} \|\,(W_i \otimes I_p)\,\mathbf{n}(t)\|_2,\end{aligned}$$

where (a) follows from $\|\mathrm{Proj}_{\mathcal{X}} a - b\| \leq \|c - b\|, \forall c \in \mathcal{X}$[2] and the fact that $\sum_j w_{i,j} x_j(t) \in \mathcal{X}$, W_i is the ith row of the weight matrix. Therefore, we have the following bound on $\|\mathbf{u}\|^2$:

$$\|\mathbf{u}\|^2 = \|\,(W \otimes I_p)\,\mathbf{n}(t)\|^2 \leq \|\mathbf{n}(t)\|^2. \tag{E.18}$$

[2]This property follows directly from the definition of projection (5.10)

The rest follows from the classical case [NO09, YLY16], by bounding the deviation of $\hat{z}_i$ from $\frac{1}{N}\sum_{i=1}^{N}\hat{z}_i$ assuming zero initial states[3] by expanding (E.16),

$$\begin{aligned}
&\|\hat{\mathbf{z}}(t)-\frac{1}{N}\left(\mathbf{1}_N\mathbf{1}_N^T\otimes I_p\right)\hat{\mathbf{z}}(t)\| \\
&\quad\overset{(a)}{=}\|\sum_{s=0}^{t-1}\left(W^{t-s}\otimes I_p-\frac{1}{N}\mathbf{1}_N\mathbf{1}_N^T\otimes I_p\right)\mathbf{n}(s) \\
&\qquad+\sum_{s=1}^{t-1}\left(W^{t-s}\otimes I_p-\frac{1}{N}\mathbf{1}_N\mathbf{1}_N^T\otimes I_p\right)(\mathbf{u}(s)+\mathbf{v}(s))\| \\
&\quad\overset{(b)}{\leq}\sum_{s=0}^{t-1}\|W^{t-s}\otimes I_p-\frac{1}{N}\mathbf{1}_N\mathbf{1}_N^T\otimes I_p\|\|\mathbf{n}(s)\| \\
&\qquad+\sum_{s=1}^{t-1}\|W^{t-s}\otimes I_p-\frac{1}{N}\mathbf{1}_N\mathbf{1}_N^T\otimes I_p\|\|\mathbf{u}(s)+\mathbf{v}(s)\| \\
&\quad\overset{(c)}{\leq}\sum_{s=0}^{t-1}\beta^{t-s}\|\mathbf{n}(s)\|+\sum_{s=1}^{t-1}\beta^{t-s}\|\mathbf{u}(s)\|+\sum_{s=1}^{t-1}\beta^{t-s}\|\mathbf{v}(s)\| \\
&\quad\overset{(d)}{\leq}\sum_{s=0}^{t-1}\beta^{t-s}\|\mathbf{n}(s)\|+\sum_{s=1}^{t-1}M_s\beta^{t-s}\|\mathbf{n}(s)\|+\sqrt{N}G\sum_{s=1}^{t-1}\eta_s\beta^{t-s} \\
&\quad=\sum_{s=1}^{t-1}2\beta^{t-s}\|\mathbf{n}(s)\|+\sqrt{N}G\sum_{s=1}^{t-1}\eta_s\beta^{t-s},
\end{aligned}\tag{E.19}$$

where (a) holds since W is doubly stochastic, (b) follows from the triangle inequality together with the inequality $\|Ax\|\leq\|A\|\|x\|$ where $\|A\|$ denotes the operator norm[4], we use the spectral property of the weight matrix in (c) and β is defined as the second largest eigenvalue of W, we plugged in (E.18) and (E.17) to derive (d). □

Having set Lemma E.2, we bound the the deviation of $z_i(t)$ from the mean parameter for Stage I as follows.

$$\|\mathbf{z}(t)-\frac{1}{N}\left(\mathbf{1}_N\mathbf{1}_N^T\otimes I_p\right)\mathbf{z}(t)\|\tag{E.20}$$

[3]Without loss of generality we assume the initial condition is zero, otherwise there is an extra term that goes to zero exponentially fast.

[4]$\|Ax\|\leq\|A\|\|x\|$ holds for matrix $A\in\mathbb{R}^{n\times n}$ and $x\in\mathbb{R}^n$ where $\|A\|$ is the operator norm of a matrix.

$$
\begin{aligned}
&= \| \left(I_{pN} - \frac{1}{N} \left(\mathbf{1}_N \mathbf{1}_N^T \otimes I_p \right) \right) \left(\hat{\mathbf{z}}(t) + \mathbf{u}(t) \right) \| \\
&\leq \| \hat{\mathbf{z}}(t) - \frac{1}{N} \left(\mathbf{1}_N \mathbf{1}_N^T \otimes I_p \right) \hat{\mathbf{z}}(t) \| \\
&\qquad + \| \left(I_{pN} - \frac{1}{N} \left(\mathbf{1}_N \mathbf{1}_N^T \otimes I_p \right) \right) \mathbf{u}(t) \| \\
&\leq \sum_{s=1}^{t-1} 2M_s \beta^{t-s} \|\mathbf{n}(s)\| + \sqrt{N} G \sum_{s=1}^{t-1} \eta_s \beta^{t-s} + \|\mathbf{u}(t)\|,
\end{aligned}
$$

where we used Lemma E.2, (E.18) along with triangle inequality and the inequality $\|Ax\| \leq \|A\|\|x\|$.

In Stage II, $\mathbf{x}(t) = (W \otimes I_P)\, \mathbf{x}(t-1)$ for $t > T$, therefore

$$
\begin{aligned}
&\| \mathbf{x}(t) - \frac{1}{N} \left(\mathbf{1}_N \mathbf{1}_N^T \otimes I_p \right) \mathbf{x}(t) \| \\
&\qquad = \| (W^{t-T} \otimes I_p - \frac{1}{N} \left(\mathbf{1}_N \mathbf{1}_N^T \otimes I_p \right)) \mathbf{x}(T) \| \\
&\qquad \leq \beta^{t-T} \mathbf{x}(T), \qquad \text{(E.21)}
\end{aligned}
$$

where we use the spectral property of W.

E.4 Proof of Lemma 5.3

Let us define $\hat{z}_i(t) \stackrel{\triangle}{=} \sum_{j \in \mathcal{N}_i} w_{ij} y_j(t)$, *i.e.*, $z_i(t) = \text{Proj}_{\mathcal{X}} \hat{z}_i(t)$ therefore

$$
\begin{aligned}
\|z_i(t) - x\| &\stackrel{(a)}{=} \|\text{Proj}_{\mathcal{X}} \hat{z}_i(t) - \text{Proj}_{\mathcal{X}} x\| \\
&\stackrel{(b)}{\leq} \|\hat{z}_i(t) - x\|^2, \qquad \text{(E.22)}
\end{aligned}
$$

where (a) follows since $x \in \mathcal{X}$ and we used non-expansiveness property of the projection in (b). Summing up both sides of (E.22) results in

$$
\begin{aligned}
\sum_{i \in [N]} \|z_i(t) - x\|^2 &\leq \sum_{i \in [N]} \|\hat{z}_i(t) - x\|^2 \qquad \text{(E.23)} \\
&= \|\hat{\mathbf{z}}(t) - \mathbf{1}_N \otimes x\|^2 \\
&\stackrel{(a)}{=} \| \left(W \otimes I_p \right) y(t) - \mathbf{1}_N \otimes x \|^2
\end{aligned}
$$

$$\overset{(b)}{=} \| \left(W \otimes I_p\right) \left(y(t) - \mathbf{1}_N \otimes x\right) \|^2$$
$$\overset{(c)}{\leq} \|y(t) - \mathbf{1}_N \otimes x\|^2 = \sum_{i \in [N]} \|y_i(t) - x\|^2,$$

where (a) follows directly from definition of $\hat{z}_i(t)$, W being doubly stochastic implies (b) and (c) is from the spectral properties of W.

Note that,

$$\|y_i(t) - x\|^2 = \|x_i(t-1) - x\|^2 + \|n_i(t)\|^2 + 2\langle x_i(t-1) - x, n_i(t)\rangle. \tag{E.24}$$

The proof is complete by plugging (E.24) into (E.23) together and taking the expectation from both sides.

References

[AM06] Panos J Antsaklis and Anthony N Michel. *Linear systems*. Springer Science & Business Media, 2006.

[ASH13] Saurabh Amin, Galina A Schwartz, and Amir Hussain. In quest of benchmarking security risks to cyber-physical systems. *IEEE Network*, 27(1):19–24, 2013.

[BBDS12] Jeremiah Blocki, Avrim Blum, Anupam Datta, and Or Sheffet. The johnson-lindenstrauss transform itself preserves differential privacy. In *Foundations of Computer Science (FOCS), 2012 IEEE 53rd Annual Symposium on*, pages 410–419. IEEE, 2012.

[BGP17] Cheng-Zong Bai, Vijay Gupta, and Fabio Pasqualetti. On kalman filtering with compromised sensors: Attack stealthiness and performance bounds. *IEEE Trans. Autom. Control*, 62(12):6641–6648, 2017.

[BKL+06] Mogens Blanke, Michel Kinnaert, Jan Lunze, Marcel Staroswiecki, and J Schröder. *Diagnosis and fault-tolerant control*, volume 691. Springer, 2006.

[BPG17] Cheng-Zong Bai, Fabio Pasqualetti, and Vijay Gupta. Data-injection attacks in stochastic control systems: Detectability and performance tradeoffs. *Automatica*, 82:251–260, 2017.

[BSST09] Clark W Barrett, Roberto Sebastiani, Sanjit A Seshia, and Cesare Tinelli. Satisfiability modulo theories. *Handbook of satisfiability*, 185:825–885, 2009.

[BST14a] R. Bassily, A. Smith, and A. Thakurta. Private empirical risk minimization: Efficient algorithms and tight error bounds. In *2014 IEEE 55th Annual Symposium on Foundations of Computer Science*, pages 464–473, Oct 2014.

[BST14b] Raef Bassily, Adam Smith, and Abhradeep Thakurta. Private empirical risk minimization: Efficient algorithms and tight error bounds. In *Foundations of Computer Science (FOCS), 2014 IEEE 55th Annual Symposium on*, pages 464–473. IEEE, 2014.

[CAS08] Alvaro A Cárdenas, Saurabh Amin, and Shankar Sastry. Research challenges for the security of control systems. In *HotSec*, 2008.

[CM09] Kamalika Chaudhuri and Claire Monteleoni. Privacy-preserving logistic re-

gression. In D. Koller, D. Schuurmans, Y. Bengio, and L. Bottou, editors, *Advances in Neural Information Processing Systems 21*, pages 289–296. Curran Associates, Inc., 2009.

[CMS11] Kamalika Chaudhuri, Claire Monteleoni, and Anand D Sarwate. Differentially private empirical risk minimization. *Journal of Machine Learning Research*, 12(Mar):1069–1109, 2011.

[CWH15a] Michelle S Chong, Masashi Wakaiki, and Joao P Hespanha. Observability of linear systems under adversarial attacks. In *American Control Conference (ACC)*, pages 2439–2444, 2015.

[CWH15b] Michelle S Chong, Masashi Wakaiki, and Joao P Hespanha. Observability of linear systems under adversarial attacks. In *American Control Conference (ACC)*, pages 2439–2444, 2015.

[CY16] Paul Cuff and Lanqing Yu. Differential privacy as a mutual information constraint. In *Proceedings of the 2016 ACM*, CCS '16, pages 43–54, New York, NY, USA, 2016. ACM.

[DPT15] Claudio De Persis and Pietro Tesi. Input-to-state stabilizing control under denial-of-service. *IEEE Transactions on Automatic Control*, 60(11):2930–2944, 2015.

[DR+14] Cynthia Dwork, Aaron Roth, et al. The algorithmic foundations of differential privacy. *Foundations and Trends® in Theoretical Computer Science*, 9(3–4):211–407, 2014.

[DRV10] C. Dwork, G. N. Rothblum, and S. Vadhan. Boosting and differential privacy. In *2010 IEEE 51st Annual Symposium on Foundations of Computer Science*, pages 51–60, Oct 2010.

[DV93] James J Downs and Ernest F Vogel. A plant-wide industrial process control problem. *Computers & chemical engineering*, 17(3):245–255, 1993.

[FGA11] S. Farahmand, G. B. Giannakis, and D. Angelosante. Doubly robust smoothing of dynamical processes via outlier sparsity constraints. *IEEE Transactions on Signal Processing*, 59(10):4529–4543, Oct 2011.

[FTD14a] Hamza Fawzi, Paulo Tabuada, and Suhas Diggavi. Secure estimation and control for cyber-physical systems under adversarial attacks. *IEEE Transactions on Automatic Control*, 59(6):1454–1467, 2014.

[FTD14b] Hamza Fawzi, Paulo Tabuada, and Suhas Diggavi. Secure estimation and control for cyber-physical systems under adversarial attacks. *IEEE Transactions on Automatic Control*, 59(6):1454–1467, 2014.

[GLB10] Abhishek Gupta, Cédric Langbort, and Tamer Basar. Optimal control in the presence of an intelligent jammer with limited actions. In *49th IEEE Conference on Decision and Control (CDC)*, pages 1096–1101, 2010.

[Gre15] Andy Greenberg. Hackers remotely kill a jeep on the highway, with me in it. *[online] http://www.wired.com/2015/07/hackers-remotely-kill-jeep-highway*, 2015.

[GUC+18] Jairo Giraldo, David Urbina, Alvaro Cardenas, Junia Valente, Mustafa Faisal, Justin Ruths, Nils Ole Tippenhauer, Henrik Sandberg, and Richard Candell. A survey of physics-based attack detection in cyber-physical systems. *ACM Computing Surveys (CSUR)*, 51(4):76, 2018.

[Hau83] M.L.J. Hautus. Strong detectability and observers. *Linear Algebra and its Applications*, 50(Supplement C):353 – 368, 1983.

[HMV15] Zhenqi Huang, Sayan Mitra, and Nitin Vaidya. Differentially private distributed optimization. In *Proceedings of the 2015 International Conference on Distributed Computing and Networking*, ICDCN '15, pages 4:1–4:10, New York, NY, USA, 2015. ACM.

[HO16] Farshad Harirchi and Necmiye Ozay. Guaranteed model-based fault detection in cyber-physical systems: A model invalidation approach. *arXiv preprint arXiv:1609.05921*, 2016.

[How12] Norman R Howes. *Modern analysis and topology*. Springer Science & Business Media, 2012.

[Jon73] Harold Lee Jones. *Failure detection in linear systems.* PhD thesis, Massachusetts Institute of Technology, 1973.

[Jun01] Ulrich Junker. Quickxplain: Conflict detection for arbitrary constraint propagation algorithms. In *IJCAI01 Workshop on Modelling and Solving problems with constraints*, 2001.

[Kel16] Leo Kelion. Nissan leaf electric cars hack vulnerability disclosed. *[online] http://www.bbc.com/news/technology-35642749*, 2016.

[KJ16] Shiva Prasad Kasiviswanathan and Hongxia Jin. Efficient private empirical

risk minimization for high-dimensional learning. In *International Conference on Machine Learning*, pages 488–497, 2016.

[KKMM13] Krishnaram Kenthapadi, Aleksandra Korolova, Ilya Mironov, and Nina Mishra. Privacy via the johnson-lindenstrauss transform. *Journal of Privacy and Confidentiality*, 5, 2013.

[KST12] Daniel Kifer, Adam Smith, and Abhradeep Thakurta. Private convex empirical risk minimization and high-dimensional regression. In *Conference on Learning Theory*, pages 25–1, 2012.

[Lan11] Ralph Langner. Stuxnet: Dissecting a cyberwarfare weapon. *IEEE Security & Privacy*, 9(3):49–51, 2011.

[LBP10] Daniel Le Berre and Anne Parrain. The sat4j library, release 2.2, system description. *Journal on Satisfiability, Boolean Modeling and Computation*, 7:59–64, 2010.

[LCB94] Yann LeCun, Corinna Cortes, and C. J. Burges. The mnist database of handwritten digits. In *Pattern Recognition, 1994. Vol. 2-Conference B: Computer Vision & Image Processing., Proceedings of the 12th IAPR International. Conference on*, volume 2, pages 77–82. IEEE, 1994.

[Lju87] Lennart Ljung. *System identification: Theory for the User*. Englewood Cliffs, 1987.

[LM03] A. Lapidoth and S. M. Moser. Capacity bounds via duality with applications to multiple-antenna systems on flat-fading channels. *IEEE Transactions on Information Theory*, 49(10):2426–2467, Oct 2003.

[MB10] J. Mattingley and S. Boyd. Real-time convex optimization in signal processing. *IEEE Signal Processing Magazine*, 27(3):50–61, May 2010.

[MCS14] Yilin Mo, Rohan Chabukswar, and Bruno Sinopoli. Detecting integrity attacks on scada systems. *IEEE Transactions on Control Systems Technology*, 22(4):1396–1407, 2014.

[MGCS10] Yilin Mo, Emanuele Garone, Alessandro Casavola, and Bruno Sinopoli. False data injection attacks against state estimation in wireless sensor networks. In *49th IEEE Conference on Decision and Control (CDC)*, pages 5967–5972, 2010.

[MHS14] Yilin Mo, Joao P Hespanha, and Bruno Sinopoli. Resilient detection in

the presence of integrity attacks. *IEEE Transactions on Signal Processing*, 62(1):31–43, 2014.

[MKB+12] Yilin Mo, Tiffany Hyun-Jin Kim, Kenneth Brancik, Dona Dickinson, Heejo Lee, Adrian Perrig, and Bruno Sinopoli. Cyber–physical security of a smart grid infrastructure. *Proceedings of the IEEE*, 100(1):195–209, 2012.

[MS09] Yilin Mo and Bruno Sinopoli. Secure control against replay attacks. In *47th Annual Allerton Conference on Communication, Control, and Computing*, pages 911–918. IEEE, 2009.

[MS15] Yilin Mo and Bruno Sinopoli. Secure estimation in the presence of integrity attacks. *Automatic Control, IEEE Transactions on*, 60(4):1145–1151, 2015.

[MS16] Yilin Mo and Bruno Sinopoli. On the performance degradation of cyber-physical systems under stealthy integrity attacks. *IEEE Transactions on Automatic Control*, 61(9):2618–2624, 2016.

[MSK+17] Shaunak Mishra, Yasser Shoukry, Nikhil Karamchandani, Suhas Diggavi, and Paulo Tabuada. Secure state estimation: Optimal guarantees against sensor attacks in the presence of noise. *IEEE Transactions on Control of Network Systems*, 4(1):49–59, 2017.

[MWVHDM06] Ivan Markovsky, Jan C Willems, Sabine Van Huffel, and Bart De Moor. *Exact and approximate modeling of linear systems: A behavioral approach*, volume 11. SIAM, 2006.

[Nes07] Yurii Nesterov. Gradient methods for minimizing composite objective function, 2007.

[NM15] Yorie Nakahira and Yilin Mo. Dynamic state estimation in the presence of compromised sensory data. In *54th Annual Conference on Decision and Control (CDC)*, pages 5808–5813. IEEE, 2015.

[NO09] A. Nedic and A. Ozdaglar. Distributed subgradient methods for multi-agent optimization. *IEEE Transactions on Automatic Control*, 54(1):48–61, Jan 2009.

[PDB13] Fabio Pasqualetti, Florian Dorfler, and Francesco Bullo. Attack detection and identification in cyber-physical systems. *IEEE Transactions on Automatic Control*, 58(11):2715–2729, 2013.

[PW15] M. Pilanci and M. J. Wainwright. Randomized sketches of convex pro-

grams with sharp guarantees. *IEEE Transactions on Information Theory*, 61(9):5096–5115, Sept 2015.

[PWB+14a] Miroslav Pajic, James Weimer, Nicola Bezzo, Paulo Tabuada, Oleg Sokolsky, Insup Lee, and George J Pappas. Robustness of attack-resilient state estimators. In *ICCPS'14: ACM/IEEE 5th International Conference on Cyber-Physical Systems (with CPS Week 2014)*, pages 163–174, 2014.

[PWB+14b] Miroslav Pajic, James Weimer, Nicola Bezzo, Paulo Tabuada, Oleg Sokolsky, Insup Lee, and George J Pappas. Robustness of attack-resilient state estimators. In *ICCPS'14: ACM/IEEE 5th International Conference on Cyber-Physical Systems (with CPS Week 2014)*, pages 163–174, 2014.

[Ric93] N Lawrence Ricker. Model predictive control of a continuous, nonlinear, two-phase reactor. *Journal of Process Control*, 3(2):109–123, 1993.

[RV10] Mark Rudelson and Roman Vershynin. Non-asymptotic theory of random matrices: extreme singular values. Technical Report arXiv:1003.2990, Mar 2010. Comments: Submission for International Congress of Mathematicians, Hydebabad, India, 2010.

[SC13] Anand D Sarwate and Kamalika Chaudhuri. Signal processing and machine learning with differential privacy: Algorithms and challenges for continuous data. *IEEE signal processing magazine*, 30(5):86–94, 2013.

[SCW+15] Yasser Shoukry, Michelle Chong, Masashi Wakiaki, Pierluigi de Nuzzo, Alberto D Sangiovanni-Vincentelli, Sanjit Seshia, Joao P Hespanha, and Paulo Tabuada. Smt-based observer design for cyber physical systems under sensor attacks. In *American Control Conference (ACC)*, pages 2439–2444, 2015.

[She17] Or Sheffet. Differentially private ordinary least squares. In *International Conference on Machine Learning*, pages 3105–3114, 2017.

[SKD18] Mehrdad Showkatbakhsh, Can Karakus, and Suhas Diggavi. Privacy-utility trade-off of linear regression under random projections and additive noise. In *IEEE International Symposium on Information Theory (ISIT*. IEEE, 2018.

[SKD19] Mehrdad Showkatbakhsh, Can Karakus, and Suhas Diggavi. Differentially private consensus-based distributed optimization. *Submitted for publication. Preprint available online, arxiv.org*, 2019.

[Smi15] Roy S Smith. Covert misappropriation of networked control systems: Presenting a feedback structure. *Control Systems Magazine, IEEE*, 35(1):82–

92, 2015.

[SNB+15] Yasser Shoukry, Pierluigi Nuzzo, Nicola Bezzo, Alberto L Sangiovanni-Vincentelli, Sanjit A Seshia, and Paulo Tabuada. Secure state reconstruction in differentially flat systems under sensor attacks using satisfiability modulo theory solving. In *54th Annual Conference on Decision and Control (CDC)*, pages 3804–3809. IEEE, 2015.

[SNP+17] Yasser Shoukry, Pierluigi Nuzzo, Alberto Puggelli, Alberto L Sangiovanni-Vincentelli, Sanjit A Seshia, and Paulo Tabuada. Secure state estimation for cyber physical systems under sensor attacks: a satisfiability modulo theory approach. *IEEE Transactions on Automatic Control*, 62(10):4917–4932, 2017.

[SPH+10] Shreyas Sundaram, Miroslav Pajic, Christoforos N Hadjicostis, Rahul Mangharam, and George J Pappas. The wireless control network: monitoring for malicious behavior. In *49th IEEE Conference on Decision and Control (CDC)*, pages 5979–5984, 2010.

[SSC+17] Mehrdad Showkatbakhsh, Yasser Shoukry, Robert H Chen, Suhas Diggavi, and Paulo Tabuada. An SMT-based approach to secure state estimation under sensor and actuator attacks. In *Decision and Control (CDC), IEEE 56th Conference on*, pages 7177–7182. IEEE, 2017.

[SSDT18] Mehrdad Showkatbakhsh, Yasser Shoukry, Suhas Diggavi, and Paulo Tabuada. Securing state estimation under sensor and actuator attacks: Theory and design. *Submitted for publication. Preprint available online, arxiv.org*, 2018.

[ST16a] Henrik Sandberg and André MH Teixeira. From control system security indices to attack identifiability. In *Science of Security for Cyber-Physical Systems Workshop (SOSCYPS)*, pages 1–6. IEEE, 2016.

[ST16b] Yasser Shoukry and Paulo Tabuada. Event-triggered state observers for sparse sensor noise/attacks. *IEEE Transactions on Automatic Control*, 61(8):2079–2091, 2016.

[STD16a] Mehrdad Showkatbakhsh, Paulo Tabuada, and Suhas Diggavi. Secure system identification. In *54th Annual Allerton Conference on Communication, Control, and Computing*, pages 1137–1141. IEEE, 2016.

[STD16b] Mehrdad Showkatbakhsh, Paulo Tabuada, and Suhas Diggavi. System identification in the presence of adversarial outputs. In *Decision and Control*

(CDC), IEEE 55th Conference on, pages 7177–7182. IEEE, 2016.

[STDP16] Danial Senejohnny, Pietro Tesi, and Claudio De Persis. A jamming-resilient algorithm for self-triggered network coordination. *arXiv preprint arXiv:1603.02563*, 2016.

[TDJ+14] Ashish Tiwari, Bruno Dutertre, Dejan Jovanović, Thomas de Candia, Patrick D Lincoln, John Rushby, Dorsa Sadigh, and Sanjit Seshia. Safety envelope for security. In *ACM Proceedings of the 3rd international conference on High confidence networked systems*, pages 85–94, 2014.

[TSSJ15] Antonio Teixeira, Kin Cheong Sou, Henrik Sandberg, and Karl H Johansson. Secure control systems: A quantitative risk management approach. *IEEE Control Systems Magazine*, 35(1):24–45, 2015.

[Vem05] Santosh S Vempala. *The random projection method*, volume 65. American Mathematical Soc., 2005.

[Ver18] Roman Vershynin. *High-dimensional probability: An introduction with applications in data science*, volume 47. Cambridge University Press, 2018.

[VODM95] Peter Van Overschee and Bart De Moor. A unifying theorem for three subspace system identification algorithms. *Automatica*, 31(12):1853–1864, 1995.

[VODM12] Peter Van Overschee and BL De Moor. *Subspace identification for linear systems: Theory-Implementation-Applications*. Springer Science & Business Media, 2012.

[Wed73] Per-Åke Wedin. Perturbation theory for pseudo-inverses. *BIT Numerical Mathematics*, 13(2):217–232, Jun 1973.

[WLK+17] Xi Wu, Fengan Li, Arun Kumar, Kamalika Chaudhuri, Somesh Jha, and Jeffrey Naughton. Bolt-on differential privacy for scalable stochastic gradient descent-based analytics. In *Proceedings of the 2017 ACM International Conference on Management of Data*, pages 1307–1322. ACM, 2017.

[WP13] Jan C Willems and Jan W Polderman. *Introduction to mathematical systems theory: a behavioral approach*, volume 26. Springer Science & Business Media, 2013.

[YB75] T Yoshikawa and S Bhattacharyya. Partial uniqueness: Observability and

input identifiability. *IEEE Transactions on Automatic Control*, 20(5):713–714, 1975.

[YFF16] Sze Zheng Yong, Ming Qing Foo, and Emilio Frazzoli. Robust and resilient estimation for cyber-physical systems under adversarial attacks. In *American Control Conference (ACC), 2016*, pages 308–315. IEEE, 2016.

[YLY16] Kun Yuan, Qing Ling, and Wotao Yin. On the convergence of decentralized gradient descent. *SIAM Journal on Optimization*, 26(3):1835–1854, 2016.

[YZF15] S Yong, M Zhu, and E Frazzoli. Resilient state estimation against switching attacks on stochastic cyber-physical systems. In *IEEE International Conference on Decision and Control (CDC)*, 2015.

[ZLW09] Shuheng Zhou, John Lafferty, and Larry Wasserman. Compressed and privacy-sensitive sparse regression. *IEEE Transactions on Information Theory*, 55(2):846–866, 2009.

[ZM14] Minghui Zhu and Sonia Martinez. On the performance analysis of resilient networked control systems under replay attacks. *IEEE Transactions on Automatic Control*, 59(3):804–808, 2014.

[ZZMW17] Jiaqi Zhang, Kai Zheng, Wenlong Mou, and Liwei Wang. Efficient private erm for smooth objectives. *arXiv preprint arXiv:1703.09947*, 2017.