Secure Electric Power Grid Operation

by

Ming Qing Foo

B.A. Electrical and Information Sciences
University of Cambridge, 2013

Submitted to the School of Engineering
in partial fulfillment of the requirements for the degree of

Master of Science in Computation for Design and Optimization

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

September 2015

Signature redacted

Author .....
School of Engineering
August 7, 2015

Certified by..... Signature redacted .....
Mardavij Roozbehani
Principal Research Scientist
Laboratory for Information and Decision Systems
Thesis Supervisor

Certified by..... Signature redacted .....
Saurabh Amin
Assistant Professor of Civil and Environmental Engineering
Thesis Reader

Accepted by..... Signature redacted .....
Nicolas Hadjiconstantinou
Professor of Mechanical Engineering
Co-Director, Computation for Design and Optimization

Secure Electric Power Grid Operation

by

Ming Qing Foo

Submitted to the School of Engineering
on August 7, 2015, in partial fulfillment of the
requirements for the degree of
Master of Science in Computation for Design and Optimization

Abstract

This thesis examines two problems concerning the secure and reliable operation of the electric power grid. The first part studies the distributed operation of the electric power grid using the power flow problem, which is vital to the operation of the grid. The power flow problem is a feasibility problem for finding an assignment of complex bus voltages that satisfies the power flow equations and is within operational and safety limits. For reliability and privacy reasons, it is desirable to solve the power flow problem in a distributed manner. Two novel distributed algorithms are presented for solving convex feasibility problems for networks based on the Method of Alternating Projections (MAP) and the Projected Consensus algorithm. These algorithms distribute computation among the nodes of the network and do not require any form of central coordination. The original problem is equivalently split into small local sub-problems, which are coordinated locally via a thin communication protocol. Although the power flow problem is non-convex, the new algorithms are demonstrated to be powerful heuristics using IEEE test beds. Quadratically Constrained Quadratic Programs (QCQP), which occur in the projection sub-problems, are studied and methods for solving them efficiently are developed. The second part addresses the robustness and resiliency of state estimation algorithms for cyber-physical systems. The operation of the electric power grid is modeled as a dynamical system that is supported by numerous feedback control mechanisms, which depend heavily on state estimation algorithms. The electric power grid is constantly under attack and, if left unchecked, these attacks may corrupt state estimates and lead to severe consequences. This thesis proposes a novel dynamic state estimator that is resilient against data injection attacks and robust to modeling errors and additive noise signals. By leveraging principles of robust optimization, the estimator can be formulated as a convex optimization problem and its effectiveness is demonstrated in simulations of an IEEE 14-bus system.

Thesis Supervisor: Mardavij Roozbehani
Title: Principal Research Scientist
Department: Laboratory for Information and Decision Systems

Acknowledgments

First, I would like to thank my thesis advisor, Dr. Mardavij Roozbehani, for his patience and guidance during my graduate studies. Mardavij has always set aside time to help me move forward with this work, event if it meant reviewing basic mathematics principles or debugging computer programs. I greatly admire Mardavij's passion towards research and I am grateful for the opportunity to collaborate with Mardavij over the past two years.

I would also like to thank members of the Laboratory for Information and Decision Systems for their invaluable feedback and discussion on this research, especially Sze Zheng Yong, who has been my mentor and my friend. Many of the difficulties that I have encountered in the course of this research were resolved thanks to his insightful comments during afternoon discussions.

Finally, I would like to dedicate this thesis to my family. This accomplishment would not have been possible without their unwavering love and support.

Contents

Chapter 1

Introduction

The electric power grid is the largest and most complex engineering system in the world. For over a century, it has contributed to the growth of the world's economy and quality of life. As our dependence on electricity grows, it will face serious challenges in the near future that will require the intervention of new technologies. This thesis examines two problems concerning the privacy, security and reliability of electric power grid operation.

The first part of this thesis studies the distributed operation of the electric power grid. The term "*distributed*" will be used to denote an approach that is fully decentralized, which does not require any form of central coordination, which evolves by local message exchanges, and which is scalable. Current operations are dominated by centralized schemes that require complicated communication protocols to monitor operating conditions throughout the grid and a powerful central computer to process the large amount of data, which become impractical as the size of the grid grows. The advent of the smart grid and development of computational abilities in network components will also encourage distributed operation schemes by means of communication between the components, while privacy and security concerns discourage the collection and storage of information. Distributed approaches are also naturally suited for improving system reliability in the presence of faulty processes.

Our motivating application is the power flow problem, which is vital to the operation of the electric power grid. The power flow problem is a feasibility problem for

finding an operating point in a network that is consistent with the physical laws and within operational and safety limits of the network. Due to the quadratic relations between voltage and power, the power flow problem is non-convex. If the optimization of a cost function is also sought, then the problem is equivalent to the Optimal Power Flow (OPF) problem [15]. In the event of a contingency (e.g. a generator or transmission line failure), the priority of the operator is the restoration of grid operation, while optimization and cost are secondary. For this reason, the focus of this thesis is on solving the power flow problem in a quick and reliable manner.

The second part of this thesis studies the cyber-physical security of the electric power grid. The electric power network can be characterized as a Cyber-Physical System (CPS), which is defined as a computer-based system that monitors and controls physical processes using embedded sensors, actuators, control processing units and communication devices. Many other infrastructure that sustain our modern society, such as oil and natural gas distribution, water and waste-water treatment and transportation systems fall under this category.

Unlike traditional research in computer security that has focused on the protection of information [1, 13], the study of cyber-physical security considers how these attacks affect estimation and control algorithms, and ultimately, the physical infrastructure. The operation of a cyber-physical system can be modeled as a dynamical system that is supported by numerous feedback control mechanisms, which depend heavily on state estimation algorithms. As a CPS is connected to the internet for remote monitoring and control, it becomes vulnerable to cyber attacks on its communication channels, while its large scale also makes it challenging to secure every system component. Failure to check these attacks may corrupt state estimates and lead to physical consequences in the forms of faults and failures.

The term "*resilient*" will be used to describe a system that can withstand attacks, while a "*robust*" system is insensitive to random disturbances and modeling errors, i.e., discrepancies between the model used to design the state estimate and the real dynamics of the controlled system. It is necessary to distinguish between resilient estimation from the relatively well-studied field of robust estimation. The latter assumes

that the disturbance signals are natural and that they follow a specific probabilistic model. These assumptions are not justifiable when adversarial actions are involved.

Early research on the design of resilient systems have focused on the characterization of undetectable attacks and on attack detection and identification techniques. These range from a simple application of data time-stamps [54] to hypothesis testing using residuals [14, 37, 42]. More recent works have addressed the problem of state estimation despite attacks, but assume the absence of modeling errors and noise signals. These assumptions are difficult to guarantee in practice, which motivates the goal of this thesis – to design a dynamic state estimator that is both *robust* and *resilient.*

1.1 Literature Review

1.1.1 Electric Power Grid Operation

The electric power grid consists of three interacting physical elements: i) generating units produces electric energy by harnessing other forms of energy; ii) loads are the end users that consume electric energy in a multitude of ways; and iii) the transmission and distribution network connects the generating units to the loads. In this section, we briefly describe the main physical elements of the electric power grid and how they operate in tandem.

Electric power is produced by generating units that convert primary energy into electric energy. Primary energy comes from a number of sources, such as fossil and nuclear fuels and other renewable energy sources. The transmission system carries electric power over long distances in transmission lines from the generating units to the distribution system, which subsequently transmits electric power from distribution substations to loads that are in close proximity. Distribution networks are distinguished from transmission networks by their voltage levels and topologies. Transmission networks have meshed topologies with higher voltage levels to reduce transmission losses, while distribution networks usually have radial topologies with

lower voltage levels to cater to load requirements.

At the end users, electric power is consumed by a wide variety of loads. Loads can have impedances that are resistive, reactive, or a combination of the two. Resistive loads consume only real power while reactive loads may consume or supply reactive power. To cope with different types of loads, generating units must be able to absorb and supply both real and reactive power.

The objective of real-time operation of the electric power grid is to ensure that the system remains stable and protected while meeting end user demands for electric power. Unlike oil and gas, electricity cannot be stored economically except in small quantities. The development of new storage technologies and high penetrations of electric vehicles may change this, but these developments are unlikely to occur in the near future. The main challenge for operators is that the supply of electricity must match consumption at all times. Since the load is changing all the time in ways that cannot be perfectly predicted, generation must follow the load in real time. To this end, the operator must solve the power flow problem, which will be discussed in detail in Appendix B.

1.1.2 Distributed Feasibility Algorithms for Networks

A feasibility or constraint satisfaction problem is defined by its three main components: variables, values and constraints. A solution to a feasibility problem is an assignment of values to variables that satisfies all constraints.

We are interested in solving feasibility problems that can be distributed across many agents. The classical approach to distributed algorithms has been decomposition: based on the specific structure of the constraints, the problem is decomposed into a number of sub-problems. These sub problems can be solved independently, but they typically require a centralized coordinator to ensure that the local decisions converge to a global feasible solution. In addition, the algorithm imposes a certain computation and communication structure among the individual agents.

However, the situation is the reverse in certain emerging applications of network problems: the communication and computation structure of the problem is given and

the implementation of a centralized coordinator is undesirable or infeasible. Examples include ad hoc wireless communication networks [16] and sensor networks [47] that are characterized by a lack of centralized control and access to information. Furthermore, a distributed approach may be preferred in the case of very-large scale networks where sub-problems are easier to solve simply because they are smaller or because they have a special structure that can be exploited.

We shall focus on solving feasibility problems for networks that have the following characteristics:

1. The network has a variable-based model – there exist m nodes $1, 2, \cdots, m$, and each variable x_i belongs to the node i.

2. There exist constraints on each node and inter-node constraints, in which case an edge is drawn between each pair of nodes.

3. Communication is restricted to immediate neighbors.

4. Knowledge (i.e., domains, variables and constraints) is local to nodes and their neighbors, and cannot be centralized for different reasons.

5. A solution is an assignment of values to nodes that satisfies every constraint, i.e. globally consistent.

6. All nodes cooperate to find a solution.

The feasibility problem considered here shares similarities with the distributed constraint satisfaction problem that was first studied by Yokoo [52] and which has since gained considerable interest in the field of artificial intelligence. Unlike Yokoo's problem, we assume a static communication structure between nodes and do not restrict ourselves to discrete variables and constraint sets.

In any distributed algorithm, a communication and timing model is necessary. We will use a synchronous model, which is the simplest model to describe, program, and reason about. Steps are performed in a synchronized fashion across agents. Synchronization solely requires that the algorithm instances each have access to a

shared sense of time, and that this can be achieved without information sharing or other communication between agents. A suitable protocol is the Network Time Protocol (NTP).

In addition, we assume the existence of a reliable underlying communication structure among the nodes and are not concerned about the implementation of the physical communication network. This is because our primary concern here is cooperation among intelligent nodes rather than solving feasibility problems by certain multiprocessor architectures.

Although distributed feasibility algorithms appear to be similar to parallel processing methods for solving feasibility problems, research motivations are fundamentally different. The primary concern in parallel processing is efficiency, while distributed feasibility algorithms assume that we can choose any type of parallel computer architecture.

1.1.3 Secure State Estimation

For linear systems under attack, [22] maps the resilient state estimation problem onto an ℓ_0 optimization problem when the attacks are unbounded and provides a characterization of the maximum number of attacks that can be tolerated. The estimator is subsequently relaxed using the "ℓ_1/ℓ_r" norm and demonstrated to be effective under the prescribed conditions. However, this approach assumes the absence of modeling errors and noise signals.

[41] extends the previous estimator for linear systems with bounded noise signals and modeling errors by constraining them in the optimization problem. However, this approach generates an estimate for the case where the noise signals and modeling errors are benign, i.e., they cancel out the attack signals, and hence does not provide the robustification that we seek. [53] proposes a robust and resilient estimator that generates unbiased estimates asymptotically when the system is perturbed by noise signals that are zero mean, Gaussian white processes. Both estimators require the solution of a combinatorial problem, which is intractable for large systems.

A zero-sum game theoretic approach to robust and resilient estimation can be

formulated using H_∞ filtering [46]. The problem is viewed as a dynamic game between two players with competing goals. The first player is the CPS operator, whose objective is to minimize a cost function that depends on the estimation errors. The second player is the adversary, who wants to maximize the same objective. In this setting, the noise signals and modeling errors are assumed to be adversarial.

Even in the absence of attacks, the robust estimation problem with modeling errors and noise signals is of significant interest and has been primarily considered from the Bayesian perspective, i.e., with the assumption of known priors. The robust Kalman filtering approach in [49] minimizes the mean squared state estimation error asymptotically using multiple steady-state Riccatti equations, whereas the set-valued filtering approach in [12] utilizes semidefinite relaxation techniques for computing minimal size ellipsoids that bound the solution set of a system of uncertain linear equations.

Another set of relevant literature pertains to that of robust optimization, which addresses the problem of optimization under uncertainty, in which the uncertainty model is not stochastic, but rather deterministic and set-based (e.g., [3, 5]). Of particular relevance is the subject of robust regression and specifically of the equivalence of robustification and regularization in linear regression under some assumptions on the uncertainty sets [23, 6]. This equivalence is a key tool that we will make use of in our design of a robust and resilient estimator.

1.2 Contributions

Distributed Feasibility Algorithms with Application to the Power Flow Problem

The first part of this thesis presents two novel feasibility algorithms that are based on the Method of Alternating Projections (MAP) and the Projected Consensus algorithms. Our algorithms solve convex feasibility problems by distributing computation among nodes and require only local information exchanges. The main application of

our algorithms is the power flow problem, which finds a feasible set of complex bus voltages and is vital to the operation of the electric power grid. Although the power flow problem is non-convex, our algorithms are demonstrated to be effective heuristics using various IEEE test beds. Furthermore, it will be shown that the projection sub-problems in our algorithms can be formulated as non-convex Quadratically Constrained Quadratic Programs (QCQPs) that can be solved efficiently.

Robust and Resilient State Estimation with Application to Electric Power Grid Operation

The second part of this thesis presents a novel state estimation algorithm that is resilient to data injection attacks and robust to modeling errors and additive noise signals. By leveraging principles of robust optimization, the estimator can be formulated as a convex optimization problem and the use of cross-validation to determine its hyperparameters is advocated. The effectiveness of our estimator is demonstrated in simulations of an IEEE 14-bus system.

1.3 Notation and Basic Results

For any vector $\mathbf{v} \in \mathbb{R}^n$, $\mathbf{v}_{a:b}$, $1 \leq a \leq b \leq n$, denotes the subset of $\mathbf{v}$ comprising the a-th to b-th entries of $\mathbf{v}$, inclusive. We also denote the Euclidean norm by $\|\cdot\|$.

For any matrix $M \in \mathbb{R}^{m\times n}$, $M_{(i,\cdot)} \in \mathbb{R}^n$ denotes the i-th row of M, $i \in \{1, \cdots, m\}$, and $M_{(\cdot,j)} \in \mathbb{R}^m$ denotes the j-th column of M, $j \in \{1, \cdots, n\}$. $M^\top$ denotes the transpose of M. We introduce several matrix norms used in this paper:

- ℓ_0 norm: $\|M\|_{\ell_0}$ = number of nonzero rows of M
- "mixed" ℓ_1/ℓ_r" norm: $\|M\|_{\ell_1/\ell_r} = \sum_{i=1}^m \|M_{(i,\cdot)}\|_{\ell_r}$.
- (ℓ_q/ℓ_r) subordinate norm: $\|M\|_{(\ell_q,\ell_r)} = \max_{\beta\neq 0} \frac{\|M\beta\|_{\ell_r}}{\|\beta\|_{\ell_q}}$

For two matrices $A, B \in \mathbb{R}^{m\times n}$, the trace inner-product over $\mathbb{R}^{m\times n}$ is defined as $A \cdot B = \text{trace}\left(A^\top B\right)$.

When the context makes it clear, we will use 0 (or 1) to denote either a matrix, vector or scalar of zeros (or ones).

1.4 Thesis Overview

The thesis is organized as follows.

Chapters 2 and 3 studies the distributed operation of the electric power grid. In Chapter 2, we first review Quadratically Constrained Quadratic Programs (QCQPs), a class of optimization problems that occur frequently in power system analysis, and show that certain classes of non-convex QQCPs can be solved efficiently. In Chapter 3, we study feasibility problems for networks and develop two new algorithms that distribute computation among st network nodes whilst restricting information exchanges to take place only along network edges. We demonstrate that these algorithms are powerful heuristics for solving the power flow problem using IEEE test beds. Chapter 4, addresses the problem of grid cyber-physical security by developing a dynamic state estimator that is resilient against adversarial actions and robust to modeling errors and additive noise signals. We demonstrate the effectiveness of our estimator using simulations of an IEEE 14-bus system. Finally in Chapter 5, we conclude with a summary of our findings and suggestions for future research.

Chapter 2

Quadratically Constrained Quadratic Programs

A Quadratically Constrained Quadratic Program (QCQP) is an optimization problem of the following form:

$$\begin{aligned} \mu_1^* := \quad & \min \quad && f_0(x) \\ & \text{s.t.} \quad && f_i(x) \leq 0 \quad i = 1, \cdots, m \\ & && x \in \mathbb{R}^n \end{aligned} \tag{2.1}$$

where $f_i : \mathbb{R}^n \to \mathbb{R}$, $i = 0, \cdots, m$, are quadratic functions defined as:

$$f_i(x) := x^\top Q_i x + 2\, q_i^\top x + \gamma_i$$

with coefficients $Q_i \in \mathbb{S}^n$, $q_i \in \mathbb{R}^n$ and $\gamma_i \in \mathbb{R}$, $i = 0, \cdots, m$. Q_i is allowed to be a zero matrix, thus QCQPs include Linear Programs and Quadratic Programs. Notice that an equality constraint can be represented as two inequality constraints.

QCQP is a general class of optimization problem with a wide range of applications. It can model binary variables with the constraint $x^2 - x = 0$, allowing combinatorial problems such as the knapsack or max-cut problems to be cast as QCQPs [26]. QCQPs also appear in trust-region sub-problems [44] of Sequential Quadratic Programming methods in nonlinear programming [7].

The quadratic function f_i is convex if and only if $Q_i \succeq 0$. (2.1) is a convex optimization problem if $f_0, \cdots, f_m$ are convex, in which case it can be solved in polynomial time by Semidefinite Programming methods [9], or more efficiently by specialized Second-Order Cone Programming algorithms [36].

In general, QCQPs non-convex and therefore lack computationally tractable solution methods. The standard approach in such cases is to solve a relaxation, in which (2.1) is cast as a Semidefinite Program or Second-Order Cone Program. In certain cases, these relaxations are exact, i.e., the optimal value of the relaxation is equal to the true optimal value and an solution of (2.1) can be recovered from every solution of the relaxation. It turns out that these relaxations are exact for several classes of non-convex QCQPs.

This chapter studies non-convex QCQPs and their relaxations. Due to the quadratic relationship between voltage and power, QCQPs occur in electric power system analysis. The two most important problems in power flow analysis, the optimal power flow (OPF) problem and power flow problem can be formulated as QCQPs [8]. Several conditions under which the relaxations are exact will be provided and our motivation for studying QCQPs will become clear in Chapter 3 when QCQPs are encountered in the projection sub-problems.

2.1 Convex Relaxations

2.1.1 Semidefinite Programming Relaxation

The homogenized version of (2.1) is

$$\begin{aligned} \min \quad & x^\top Q_0 x + 2\ t\ q_0^\top x + t^2\ \gamma_0 \\ \text{s.t.} \quad & x^\top Q_i x + 2\ t\ q_i^\top x + t^2\ \gamma_i \leq 0 \quad i = 1, \cdots, m \\ & t^2 = 1 \\ & x \in \mathbb{R}^n, \quad t \in \mathbb{R} \end{aligned} \tag{2.1A}$$

If (x, t) solves (2.1A), then $\frac{1}{t}x$ solves (2.1) with the same objective function value.

We adopt the following matrix notations:

$$M_i = \begin{pmatrix} \gamma_i & q_i^\top \\ q_i & Q_i \end{pmatrix}, \quad i = 0, \cdots, m \quad \text{and} \quad M_{m+1} = \begin{pmatrix} 1 & \mathbf{0}^\top \\ \mathbf{0} & \mathbf{0} \end{pmatrix}$$

If we rewrite the quadratic functions as trace inner-products between two matrices, it is clear that (2.1A) admits the following lifted representation to the real symmetric space $\mathbb{S}^{n+1}$:

$$\begin{aligned} \min \quad & M_0 \cdot X \\ \text{s.t.} \quad & M_i \cdot X \leq 0 \quad i = 1, \cdots, m \\ & M_{m+1} \cdot X = 1 \\ & X = \begin{pmatrix} X_{00} & x^\top \\ x & xx^\top \end{pmatrix} \\ & X_{00} \in \mathbb{R}, \quad x \in \mathbb{R}^n \end{aligned}$$

Notice that the non-convex equality constraint

$$X = \begin{pmatrix} X_{00} & x^\top \\ x & xx^\top \end{pmatrix}$$

is equivalent to $X \succeq 0$ and $\text{rank}(X) = 1$. A usual approach to relax this constraint is to drop the constraint $\text{rank}(X) = 1$, thus obtaining the following Semidefinite Programming relaxation of (2.1A):

$$\begin{aligned} \mu_2^* = \quad \min \quad & M_0 \cdot X \\ \text{s.t.} \quad & M_i \cdot X \leq 0 \quad i = 1, \cdots, m \\ & M_{m+1} \cdot X = 1 \\ & X \in \mathbb{S}_+^{n+1} \end{aligned} \tag{2.2}$$

Unlike (2.1), (2.2) is a Semidefinite Program and it can be solved in polynomial time [9].

2.1.2 Lagrangian Dual Relaxation

It is useful to define the following notations:

$$\begin{aligned} Q_\lambda &:= Q_0 + \sum_{i=1}^{m} \lambda_i Q_i \\ q_\lambda &:= q_0 + \sum_{i=1}^{m} \lambda_i q_i \\ \gamma_\lambda &:= \gamma_0 + \sum_{i=1}^{m} \lambda_i \gamma_i \end{aligned}$$

for some $\lambda \in \mathbb{R}^n$.

The Lagrangian of (2.1) can be written as

$$L(x, \lambda) = x^\top Q_\lambda x + 2q_\lambda^\top x + \gamma_\lambda$$

To derive the Lagrange dual relaxation of (2.1),

$$\begin{aligned} \mu_1^* &= \min_{x \in \mathbb{R}^n} \max_{\lambda \in \mathbb{R}_+^m} L(x, \lambda) \\ &\geq \max_{\lambda \in \mathbb{R}_+^m} \min_{x \in \mathbb{R}^n} L(x, \lambda) \\ &=: \mu_3^* \end{aligned}$$

Applying Shor's relaxation scheme [45],

$$\begin{aligned} \mu_3^* \quad = \max \quad & \tau \\ \text{s.t.} \quad & M_0 - \tau\, M_{m+1} + \textstyle\sum_{i=1}^{m} \lambda_i\, M_i \succeq 0 \\ & \lambda_i \geq 0, \quad i = 1 \cdots, m \\ & \tau \in \mathbb{R} \end{aligned} \tag{2.3}$$

Notice that (2.3) is a Semidefinite Program. It is straightforward to verify that (2.3) is also the conic dual of (2.2), i.e., $\mu_3^* \leq \mu_2^*$.

2.2 Equivalence of Relaxations

Assumption 2.1. (2.1) *satisfies the Slater regularity condition, i.e. there exists* $\overline{x}$ *such that* $f_i(\overline{x}) < 0$, $i = 1, \cdots, m$. *It follows that* (2.2) *satisfies the Slater regularity condition as well.*

Proposition 2.1 ([51]). (2.3) *satisfies the Slater regularity condition when either*

(a) at least one of the m *constraints is ellipsoidal, or*

(b) the objective function is strictly convex.

Proof. For case (a), let us assume that the i-th constraint is ellipsoidal, i.e. $Q_i \succ 0$ and $\gamma_i + q_i^\top Q_i^{-1} q_i < 0$. If we let $\lambda_i > 0$ be sufficiently large and $\lambda_j = 1$, $j \neq i$, , we have

$$Q_0 + \sum_{i=1}^{m} \lambda_i Q_i \succ 0$$

Then we let $\tau < 0$ be sufficiently large in absolute value to obtain

$$M_0 - \tau\, M_{m+1} + \sum_{i=1}^{m} \lambda_i\, M_i \succ 0$$

For case (b), we have $Q_0 \succ 0$. Let $\lambda_1 = \cdots = \lambda_m = \varepsilon$ for some $\varepsilon > 0$ that is sufficiently small, and $\tau < 0$ be sufficiently large in absolute value such that

$$M_0 - \tau\, M_{m+1} + \varepsilon \sum_{i=1}^{m} M_i \succ 0$$

□

It is a well-known result in optimization that strong duality holds for a convex optimization problem under constraint qualification such as Slater regularity condition [51]. Hence, if either Assumption 2.1 or the hypothesis of Proposition 2.1 holds, then we have strong duality between (2.2) and (2.3), i.e., $\mu_2^* = \mu_3^*$.

2.3 Exactness of Relaxations

In general, the solutions of the relaxations (2.2) and (2.3) do not solve (2.1) but they can provide lower bounds, i.e., $\mu_3^* \leq \mu_2^* \leq \mu_1^*$. If a solution of (2.1) can be recovered from a solution of (2.2) or (2.3) in polynomial time, then (2.2) or (2.3) is said to be an exact relaxation of (2.1). Several authors have studied conditions under which the relaxation is exact. Here we provide a few of them.

Lemma 2.1 ([8]). *Assume that the feasible set of* (2.1) *is bounded. If* X^* *solves* (2.2) *and* rank $(X^*) \leq 1$, *then* $\mu_2^* = \mu_1^*$ *and* $\frac{1}{t^*}x^*$ *solves* (2.1), *where* $\begin{pmatrix} t^* \\ x^* \end{pmatrix}$ *uniquely solves* $X^* = \begin{pmatrix} t^* \\ x^* \end{pmatrix} \begin{pmatrix} t^* \\ x^* \end{pmatrix}^\top$.

Proof. Since the feasible sets of (2.1) and hence (2.2) are bounded, μ_1^* and μ_2^* are finite. Given any feasible solution x of (2.1),

$$X := \begin{pmatrix} 1 \\ x \end{pmatrix} \begin{pmatrix} 1 \\ x \end{pmatrix}^\top$$

is a feasible solution of (2.2). Hence (2.2) is feasible and $\mu_2^* \leq \mu_1^*$.

If $\operatorname{rank} X^* = 0$, then $X^* = \mathbf{0}$ and an optimal solution to (2.1) is $x^* = \mathbf{0}$, and therefore $\mu_2^* = \mu_1^*$.

If $\operatorname{rank} X^* = 1$, then X^* has the unique decomposition

$$X^* = \begin{pmatrix} t^* \\ x^* \end{pmatrix} \begin{pmatrix} t^* \\ x^* \end{pmatrix}^\top$$

and $\mu_2^* = \operatorname{trace}(M_0 \cdot X^*) = x^{*\top} Q_0 x^* + \dfrac{2}{t^*} q_i^\top x^* + (t^*)^2 \gamma_i = \mu_1^*$.

□

Theorem 2.1 ([31]). *Assume that there exists a sign vector* $\sigma \in \{-1, 1\}^{n+1}$ *such that*

$$[M_i]_{jk} \sigma_j \sigma_k \leq 0, \quad 0 \leq j < k \leq n, \quad 1 \leq i \leq m$$

Let

$$\hat{X} = \begin{pmatrix} \hat{X}_{00} & \hat{X}_{11} & \cdots & \hat{X}_{1n} \\ \hat{X}_{11} & \ddots & & \vdots \\ \vdots & & & \\ \hat{X}_{nn} & \cdots & & \hat{X}_{nn} \end{pmatrix}$$

be an optimal solution of (2.2). *Then*

$$\begin{pmatrix} \hat{x}_0 \\ \hat{x} \end{pmatrix} = \begin{pmatrix} 1 \\ \sigma_1\sqrt{\hat{X}_{11}} \\ \vdots \\ \sigma_n\sqrt{\hat{X}_{nn}} \end{pmatrix}$$

is an optimal solution of (2.1).

Proof. From our hypothesis, we observe that:

$$\hat{x}_0 = 1 \quad \text{and} \quad \hat{x}_j^2 = \hat{X}_{jj} \quad j = 1, \cdots, n$$

Since X is positive semidefinite,

$$(X_{jk})^2 \leq X_{jj}X_{kk}, \quad 0 \leq j < k \leq n$$

Hence, it follows that:

$$\begin{aligned} [M_i]_{jk}\hat{x}_j\hat{x}_k &= [M_i]_{jk}\sigma_j\sigma_k\sqrt{X_{jj}}\sqrt{X_{kk}} \\ &\leq [M_i]_{jk}X_{jk}, \qquad\qquad 0 \leq j < k \leq n, \quad 0 \leq i \leq m \end{aligned}$$

For $i = 0, \cdots, m$, we obtain

$$\begin{aligned}\begin{pmatrix}\hat{x}_0\\ \hat{x}\end{pmatrix}^{\top} M_i \begin{pmatrix}\hat{x}_0\\ \hat{x}\end{pmatrix} &= \sum_{j=0}^{n}\sum_{k=0}^{n}[M_i]_{jk}\hat{x}_j\hat{x}_k \\ &\leq \sum_{j=0}^{n}\sum_{k=0}^{n}[M_i]_{jk}X_{jk} \\ &= M_i \cdot X\end{aligned}$$

This implies that $\begin{pmatrix}\hat{x}_0\\ \hat{x}\end{pmatrix}$ is a feasible solution of (2.1A). Since (2.2) is a relaxation of (2.1A), $\begin{pmatrix}\hat{x}_0\\ \hat{x}\end{pmatrix}$ is also an optimal solution of (2.1A).

□

2.3.1 S-Procedure

Suppose $g_i : \mathbb{V} \to \mathbb{R}$, $i = 0, \cdots, m$, are $m+1$ real valued functionals defined on a vector space $\mathbb{V}$. Consider the minimization problem

$$\begin{array}{rcll}\mu_P^* & := & \min & g_0(x) \\ & & \text{s.t.} & g_i(x) \leq 0, \quad i = 1, \cdots, m \\ & & & x \in \mathbb{V}\end{array} \tag{P}$$

The Lagrangian of (P) is

$$L(x, \tau) = g_0(x) + \sum_{i=1}^{m} \tau_i \, g_i(x)$$

Consider the following two conditions

$$g_0(x) \geq 0 \quad \forall \quad x \in \mathbb{V}, \quad \text{s.t.} \quad g_i(x) \leq 0, \quad i = 1, \cdots, m \tag{S1}$$

$$\exists \ \tau_i \geq 0, \quad i = 1, \cdots, m, \quad \forall \quad x \in \mathbb{V} \quad \text{s.t.} \quad g_0(x) + \sum_{i=1}^{m} \tau_i \, g_i(x) \geq 0 \tag{S2}$$

It is straightforward to show that (S2) $\Rightarrow$ (S1):

$$\text{(S2)} \Longrightarrow \exists\ \tau_i \geq 0, \quad i = 1, \cdots, m, \quad \text{s.t.} \quad g_0(x) \geq -\sum_{i=1}^{m} \tau_i\ g_i(x), \quad \forall\ x \in \mathbb{V}$$

$$\Longrightarrow \text{(S1)}$$

In general, the converse may not be true. The S-Procedure for (P) is the method of verifying (S1) using (S2). This is helpful since (S2) is generally easier to verify than (S1). If (S1) $\Longrightarrow$ (S2), the S-Procedure is said to be *lossless.*

Theorem 2.2 (Theorem 3.1 [24]). *Connection of the losslessness of the S-Procedure with strong duality*

The S-Procedure for (P) *is lossless if and only if strong duality holds in* (P), *i.e.*

$$\max_{\tau \geq 0} \min_{x \in \mathbb{V}}\ g_0(x) + \sum_{i=1}^{m} \tau_i\ g_i(x)\ =\ \mu_P^*$$

Proof. The proof applies the S-Procedure for the inequality $g_0(x) - c \geq 0$ for some $c \in \mathbb{R}$, subject to the constraints $g_i(x) \leq 0$, $i = 1, \cdots, m$.

We begin with the proof of the sufficient condition. Choosing $c = \mu_P^*$, (S1) gives

$$g_0(x) \geq \mu_P^* \quad \forall \quad x \in \mathbb{V}, \quad \text{s.t.} \quad g_i(x) \leq 0, \quad i = 1, \cdots, m$$

Losslessness of the S-Procedure implies (S2):

$$\exists\ \tau_i \geq 0, \quad i = 1, \cdots, m, \quad \forall \quad x \in \mathbb{V}, \quad \text{s.t.} \quad g_0(x) + \sum_{i=1}^{m} \tau_i\ g_i(x)\ \geq\ \mu_P^*$$

Hence we obtain the inequality

$$\max_{\tau \geq 0} \min_{x \in \mathbb{V}}\ g_0(x) + \sum_{i=1}^{m} \tau_i\ g_i(x)\ \geq\ \mu_P^*$$

It is straightforward to show the reverse inequality. For any $\tau \geq 0$,

$$\begin{aligned} \min_{x \in \mathbb{V}} g_0(x) + \sum_{i=1}^{m} \tau_i \, g_i(x) \quad &\leq \quad \min_{x \in \mathbb{V}} \quad g_0(x) \qquad\qquad\qquad = \quad \mu_P^* \\ &\qquad\quad \text{s.t.} \quad g_i(x) \leq 0, \quad i = 1, \cdots, m \end{aligned}$$

Taking the maximum of both sides over all $\tau \geq 0$,

$$\max_{\tau \geq 0} \min_{x \in \mathbb{V}} g_0(x) + \sum_{i=1}^{m} \tau_i \, g_i(x) \;\leq\; \mu_P^*$$

Conversely, the application of strong duality and (S1) gives

$$\begin{aligned} \max_{\tau \geq 0} \min_{x \in \mathbb{V}} g_0(x) + \sum_{i=1}^{m} \tau_i \, g_i(x) \quad &= \quad \min_{x \in \mathbb{V}} \quad g_0(x) \\ &\qquad\quad \text{s.t.} \quad g_i(x) \leq 0, \quad i = 1, \cdots, m \\ &\geq \qquad\quad c \end{aligned}$$

Therefore there exists $\tau_i \geq 0$, $i = 1, \cdots, m$, such that $g_0(x) + \sum_{i=1}^{m} \tau_i \, g_i \geq c$ for all $x \in \mathbb{V}$, i.e.,

$$g_0(x) - c + \sum_{i=1}^{m} \tau_i \, g_i \geq 0$$

which satisfies (S2).

□

Definition 2.1. (P) *is said to satisfy the regularity condition if*

$$\exists \quad \overline{x} \in \mathbb{V}, \quad s.t. \quad g_j(\overline{x}) < 0, \quad j = 1, \cdots, m \tag{2.4}$$

Lemma 2.2 (Lemma 2.1.1 [18]). *For* (P), *define the mapping* $\varphi : \mathbb{V} \to \mathbb{R}^{m+1}$,

$$\varphi(x) = \begin{pmatrix} g_0(x) \\ g_1(x) \\ \vdots \\ g_m(x) \end{pmatrix}$$

If (P) *satisfies the regularity condition* (2.4) *and its joint image set* $\Theta := \{\varphi(x) \mid x \in \mathbb{V}\} \subseteq \mathbb{R}^{m+1}$ *is convex, then the S-Procedure for* (P) *is lossless.*

Proof. An equivalent characterization of (S1) is

$$\Theta \cap \Xi = \varnothing$$

where $\Xi = \{(\xi_0, \xi) \in \mathbb{R}^1 \times \mathbb{R}^m : \xi_0 < 0,\ \xi \leq \mathbf{0}\}$ is a convex cone. The separation theorem states that disjoint convex sets can be separated by a hyperplane, i.e., there exists a non-zero $(\widetilde{\lambda}_0,\ \widetilde{\lambda}) \in \mathbb{R}^1 \times \mathbb{R}^m$ such that

$$\widetilde{\lambda}_0\theta_0 + \sum_{i=1}^{m} \widetilde{\lambda}_i\theta_i \geq 0 \quad \forall \quad (\theta_0, \theta) \in \Theta \tag{2.5}$$

$$\widetilde{\lambda}_0\xi_0 + \sum_{i=1}^{m} \widetilde{\lambda}_i\xi_i \leq 0 \quad \forall \quad (\xi_0, \xi) \in \Xi \tag{2.6}$$

Since $(-1, \mathbf{0}) \in \Xi$, from (2.6) we get $\widetilde{\lambda}_0 \geq 0$. Using $(-\varepsilon, -\mathbf{e}_i) \in \Xi$, where $\mathbf{e}_i \in \mathbb{R}^m$ is the i-th unit vector and $\varepsilon > 0$, we get $\widetilde{\lambda}_i \geq 0$, $i = 1, \cdots, m$.

By the regularity assumption (2.4), there exists $\overline{x} \in \mathbb{V}$ such that $\theta_i = g_i(\overline{x}) < 0$, $i = 1, \cdots, m$. Thus (2.5) implies that $\widetilde{\lambda}_0 > 0$. If we multiply (2.5) by $\frac{1}{\widetilde{\lambda}_0}$, we obtain

$$g_0(x) + \sum_{i=1}^{m} \frac{\widetilde{\lambda}_i}{\widetilde{\lambda}_0} g_i(x) \geq 0 \quad \forall \quad x \in \mathbb{V}$$

which shows that (S2) holds with $\lambda_i = \frac{\widetilde{\lambda}_i}{\widetilde{\lambda}_0}$, $i = 1, \cdots, m$.

□

Remark 2.1. *Yakubovich's S-lemma (1971) proves the losslessness of the S-Procedure when* $m = 1$, *and* $g_0, g_1 : \mathbb{R}^n \to \mathbb{R}$ *are quadratic functions.*

We now turn our attention back to quadratic functions. In the rest of the section, we will consider the application of S-Procedure theorems on (2.1).

Lemma 2.3. *Define the mapping* $\varphi : \mathbb{R}^n \to \mathbb{R}^{m+1}$,

$$\varphi(x) = \begin{pmatrix} x^T Q_0 x \\ x^T Q_1 x \\ \vdots \\ x^T Q_m x \end{pmatrix}$$

If Q_i, $i = 0, \cdots, m$, *are real, diagonal matrices, then its joint image set* $\Theta = \{\varphi(x) \mid x \in \mathbb{R}^n\}$ *is convex.*

Proof. For any $\lambda \in [0, 1]$, and $x_1, x_2 \in \mathbb{R}^2$, we want to prove

$$\lambda\varphi(x_1) + (1 - \lambda)\varphi(x_2) \in \Theta = \{\varphi(x) \mid x \in \mathbb{R}^n\}$$

Since Q_i, $i = 0, \cdots, m$, are real, diagonal, we observe that $\varphi(\cdot)$ is linear with respect to x_i^2, $i = 1, \cdots, n$, hence $\varphi(\cdot)$ and its image set Θ are convex.

□

Lemma 2.4 (Theorem 2.1 [24]). *Consider* (2.1). *Assume that* Q_i, $i = 0, \cdots, m$ *are real, diagonal matrices and* (2.1) *satisfies the regularity condition. Then the S-Procedure for* (2.1) *is lossless.*

Proof. Let us first assume that f_i, $i = 0, \cdots, m$ are homogeneous quadratic functions. Since Q_i, $i = 0, \cdots, m$ are real, diagonal matrices, we may use Lemma 2.3 to deduce that the joint image set of (2.1) is convex and the application of Lemma 2.2 completes the proof.

Now let f_i, $i = 0, \cdots, m$ be general, not necessarily homogeneous, quadratic functions. We may assume the Slater point to be $\overline{x} = 0$, so that the regularity condition is equivalent to $f_i(0) = \gamma_i < 0$, $i = 1, \cdots, m$; if this is not the case, we replace $\overline{f}_i(x) = f_i(x + \overline{x})$.

Define homogeneous versions of our quadratic functions $F_i : \mathbb{R}^n \times \mathbb{R} \to \mathbb{R}$, $i =$

$0, \cdots, m$, where

$$F_i(x,\xi) = \begin{cases} \xi^2 f_i\left(\frac{1}{\xi}x\right) & \text{if } \xi \neq 0 \\ x^T Q_i x & \text{if } \xi = 0 \end{cases} \tag{2.7}$$

It is clear that

$$\begin{aligned} F_i(x,\xi) &= x^T Q_i x + 2\xi q_i^\top + \xi^2 \gamma_i \\ &= \begin{pmatrix} x \\ \xi \end{pmatrix}^\top \begin{pmatrix} Q_i & q_i^\top \\ q_i & \xi \end{pmatrix} \begin{pmatrix} x \\ \xi \end{pmatrix} \end{aligned}$$

Recall that (S1) of the S-Procedure states

$$f_0(x) \geq 0 \quad \forall \quad x \in \mathbb{R}^n, \quad \text{s.t.} \quad f_i(x) \leq 0, \quad i = 1, \cdots, m$$

An equivalent statement for our new functions is

$$F_0(x,\xi) \geq 0 \quad \forall \quad (x,\xi) \in \mathbb{R}^n \times \mathbb{R}, \quad \text{s.t.} \quad F_i(x,\xi) \leq 0, \quad i = 1, \cdots, m \qquad \text{(S1')}$$

We now proceed to prove (S1) $\Rightarrow$ (S1'). By contraposition, assume that there exists $(x,\xi) \in \mathbb{R}^n \times \mathbb{R}$ such that:

$$F_0(x,\xi) < 0 \quad \text{s.t.} \quad F_i(x,\xi) \leq 0, \quad i = 1, \cdots, m$$

If $\xi \neq 0$,

$$\begin{aligned} f_0\left(\tfrac{1}{\xi}x\right) &= \frac{1}{\xi^2} F_0(x,\xi) < 0 \\ f_i\left(\tfrac{1}{\xi}x\right) &= \frac{1}{\xi^2} F_i(x,\xi) \leq 0, \quad i = 1, \cdots, m \end{aligned}$$

which contradicts (S1).

If $\xi = 0$, the same result is obtained by observing the continuity and boundedness of $f_i, i = 0, \cdots, m$.

Furthermore, choosing $(x, \xi) = (\mathbf{0}, 1)$ gives

$$F_i(\mathbf{0}, 1) = f_i(\mathbf{0}) < 0, \quad i = 1, \cdots, m$$

Therefore the regularity condition (2.4) is satisfied by the new functions, so we can apply the homogeneous version of the theorem that has been proven above. We obtain

$$\exists\ \tau_i \geq 0, \quad i = 1, \cdots, m, \quad \text{s.t.} \quad F_0(x) + \sum_{i=1}^{m} \tau_i F_i(x) \geq 0, \quad \forall \quad x \in \mathbb{R}^n \qquad \text{(S2')}$$

which recovers (S2) if we let $\xi = 1$.

□

Theorem 2.3. *Assume that* (2.1) *satisfies the regularity condition* (2.4) *and* Q_j, $j = 0, \cdots, m$, *are diagonal. Then its Lagrangian relaxation* (2.3) *is tight.*

Proof. The proof follows from Lemmas 2.2, 2.3, and 2.4, and Theorem 2.2.

□

Chapter 3

Distributed Feasibility Algorithms for solving the Power Flow Problem

A canonical problem in distributed systems and networks is to simultaneously satisfy constraints between neighboring nodes. As the size of the problem grows, centralized algorithms become impractical and undesirable, due to the reliance on a central processor. On the other hand, distributed computing environments are on the rise due to advances in hardware and networking technologies, leading to renewed interest in distributed algorithms. Such algorithms have been studied in a wide variety of feasibility problems, the most widely known being that of average consensus, i.e., of calculating an average in a distributed manner [21, 40].

The shift towards distributed algorithms is also driven by various application domains. In electric power networks or communication networks, for example, privacy and security concerns discourage the collection and storage of information while the growth in the number of active components with sensing and computational capabilities has fueled interest in distributed operation schemes. Distributed approaches are also naturally suited for improving system reliability in the presence of faulty processes.

Given a network of interconnected nodes, each with its own value (such as a measurement, position or vote) and with constraints between each node and its neighbors, this chapter develops two novel algorithms for finding feasible assignments of values

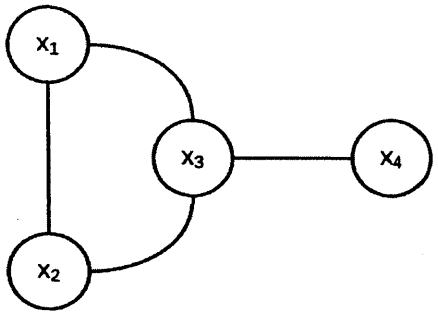
for all nodes when all constraints are convex. The algorithms, based on the Method of Alternating Projections (MAP) and the Projected Consensus algorithm, distribute computation among nodes and do not assume the existence of a communication infrastructure with topology different from the network. In other words, a node only needs to communicate with its immediate neighbors in the network.

The main application for our algorithms is the power flow problem, which is central to the operation of the electric power grid. The problem is formulated as a feasibility problem for finding an operating point that is consistent with the physical laws and within operational and safety limits of the grid. Due to the quadratic relations between voltage and power, the power flow problem is non-convex. If the optimization of a cost function is also sought, then the problem is equivalent to the Optimal Power Flow problem that was first introduced by [15]. In the event of a contingency (e.g. a generator or transmission line failure), the priority of the operator is the restoration of grid operation, while optimization and cost are secondary. For this reason, the focus of this chapter is on solving the power flow problem in a quick and reliable manner.

At first glance, applying convex feasibility algorithms on a non-convex problem would seem to contradict the convexity assumptions underlying these algorithms. But in fact, these algorithms turn out to be well-defined for general feasibility problems and often are powerful heuristics even for NP-hard non-convex problems in areas such as phase retrieval in image processing and synthesis problems in low-order control design, see, e.g. [2, 29]. Furthermore, the projection sub-problems encountered when solving the power flow problem can be formulated as Quadratically Constrained Quadratic Programs (QCQP) that can be solved efficiently.

3.1 Problem Statement

Consider a network of interconnected nodes that is described by the undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V} = \{1, \cdots, m\}$ represents the set of nodes and $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ represents the set of undirected edges between nodes. $N_i := \{j : (i,j) \in \mathcal{E}\}$ is defined



$$\begin{aligned} \text{find} \quad & x \\ \text{s.t.} \quad & f_1(x_1, x_2, x_3) \leq 0 \\ & f_2(x_1, x_2, x_3) \leq 0 \\ & f_3(x_1, x_2, x_3, x_4) \leq 0 \\ & f_4(x_3, x_4) \leq 0 \end{aligned}$$

Figure 3-1: Example feasibility problem with $\mathcal{V} = \{1, 2, 3, 4\}$

as the set of be neighbors of node i, excluding i itself, whereas $I_i := N_i \cup \{x_i\}$. Each node is assigned a value (such as a measurement, position or vote) and when it is clear from the context, $\mathcal{V}$, N_i and I_i will also be used to refer to the nodal values: $\mathcal{V} = \{x_1, \cdots, x_m\}$, $N_i := \{x_j : (i,j) \in \mathcal{E}\}$ and $I_i := N_i \cup \{i\}$.

For each node i, its value x_i and those of its neighbors are constrained to lie in the (local constraint) set $\mathcal{S}_i$. It is assumed that every nodal value x_i belongs to the same Euclidean space $\mathbb{E}$, thus $\mathcal{S}_i \subseteq \mathbb{E}^{|I_i|}$ and all nodal values can be aggregated into the value tuple $x = (x_1, \cdots, x_m) \in \mathbb{E}^m$. The feasible set of the problem is denoted as $\mathcal{S} \subseteq \mathbb{E}^m$.

The feasibility problem is to find an assignment of nodal values such that all constraint sets are satisfied simultaneously, for which an example is given in Figure 3-1. In the example, the nodal values x_1, x_2 and x_3 are constrained to lie in the set $\mathcal{S}_1 = \{(x_1, x_2, x_3) \in \mathbb{E}^3 : f_1(x_1, x_2, x_3) \leq 0\}$. Similar observations can be made for the the constraint sets $\mathcal{S}_2$, $\mathcal{S}_3$ and $\mathcal{S}_4$.

The goal of this thesis is to develop algorithms for solving the feasibility problem in a manner that i) distributes computation among nodes, ii) requires only (local) information exchanges along edges (i.e., with neighbors), and iii) keeps knowledge (e.g. domains, constraints) local to nodes.

3.2 Preliminary Material

3.2.1 Basic Results

For some Euclidean space $\mathbb{E}$, $\mathbb{E}^m$ consists of ordered m-tuples of elements of $\mathbb{E}$.

Definition 3.1. *Given a set $\mathcal{S} \subseteq \mathbb{E}$ and a vector $x \in \mathbb{E}$, the projection (if it exists) of x onto $\mathcal{S}$ is a point $p \in \mathcal{S}$ such that*

$$\|p - x\| = d(x, \mathcal{S}) := \inf_{s \in \mathcal{S}} \|x - s\|$$

If p is unique, then the projection onto $\mathcal{S}$ is the operator $P_{\mathcal{S}} : \mathbb{E} \to \mathcal{S}$ that maps x to its nearest point in $\mathcal{S}$. We write $P_{\mathcal{S}}(x) = p$. When the projection is not unique we consider $P_{\mathcal{S}}$ to be a set valued mapping, i.e. $P_{\mathcal{S}}(x) = \{p \in \mathcal{S} : \|p - x\| = d(x, \mathcal{S})\}$.

Definition 3.2. *Given a set $T \subseteq \mathbb{E}^{n+m}$, its projection onto the space of the first n coordinates is the set $U \subseteq \mathbb{E}^n$ that is defined as follows:*

$$(x_1, \cdots, x_n) \in U \iff \exists\ (y_1, \cdots, y_m) \in \mathbb{E}^m \quad \textit{such that} \quad (x_1, \cdots, x_n,\ y_1, \cdots, y_m) \in T$$

The projection onto the space of coordinates with indices $\{i_1, \cdots, i_n\} \subseteq \{1, \cdots, n+m\}$ is defined similarly.

3.2.2 Survey of Feasibility Algorithms

For this section, we assume that all sets belong to the same space, i.e., $\mathcal{S}_i \subseteq \mathbb{E}^m$.

The Method of Alternating Projections (MAP) was first proposed by Von Neumann [48] for finding the projection of a given point onto the intersection of two closed subspaces in a Hilbert space by iteratively projecting a point between the two sets. The method has since been rediscovered many times in the literature due to its simplicity and intuitive appeal. It extends in an obvious manner for finding points in the intersection $\mathcal{S}$ of multiple closed, convex sets $\mathcal{S}_1, \cdots, \mathcal{S}_m$ [11]. It is worth mentioning that the limit point in Bregman's scheme need not be the closest in the intersection to the starting point.

The MAP algorithm assumes that $P_{\mathcal{S}}$ is difficult to compute, whereas $P_{\mathcal{S}_i}$, $i = 1, \cdots, m$ are easy to obtain. Beginning with the initial value $x(0)$, the MAP algorithm solves a convex feasibility problem by generating a sequence of iterates $\{x^i(t)\}$, with $i = 0, \cdots, m$ (see Figure 3-2(a)). The superscript i indicates the cyclical projection onto the constraint sets $\mathcal{S}_1, \cdots, \mathcal{S}_m$, while the subscript j represents the entry index. The sequence is defined by the recursive formulas

$$\begin{aligned} x^0(t) &= x^m(t-1) \\ x^i(t) &= P_{\mathcal{S}_i}\left(x^{i-1}(t)\right), \quad i = 1, \cdots, m \end{aligned} \tag{3.1}$$

with initial value $x^m(0) = x(0)$.

As mentioned, (3.1) does not necessarily generate a limit point closest to $x(0)$. More recently, Boyle [10] and Dykstra [19] proposed a modification that allows MAP to generate a limit point that is closest in the intersection to $x(0)$. An extra sequence of increments $\{y^i(t)\}$ is generated, and the sequences are defined by the recursive formulas

$$\begin{aligned} &x^0(t) = x^m(t-1) \\ &x^i(t) = P_{\mathcal{S}_i}\left(x^{i-1}(t) - y^i(t-1)\right) \quad , \quad i = 1, \cdots, m \\ &y^i(t) = x^i - \left(x^{i-1}(t) - y^i(t-1)\right) \quad , \quad i = 1, \cdots, m \end{aligned} \tag{3.2}$$

with initial values $x^m(0) = x(0)$ and $y^i(t) = 0$, $i = 1, \cdots, m$. If $\mathcal{S}$ is non-empty, then the sequence of iterates in (3.2) will converge to $P_{\mathcal{S}}(x(0))$.

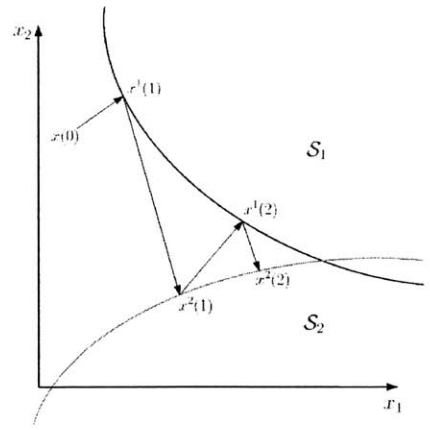
The appeal of the MAP algorithm for convex feasibility problems lies in the ease of the projection sub-problem. If a closed set in a Euclidean space is convex, then the projection of any point onto that set has a unique solution. Furthermore, computing the projection is tractable computationally using modern interior point methods provided the set is reasonably described [39]. These properties of the MAP algorithm makes it popular in a wide range of applications such as finding the correlation between stock returns [30] and solving the positive semidefinite matrix completion problem [9]. However, the MAP algorithm is not amenable for parallel or distributed

implementation.

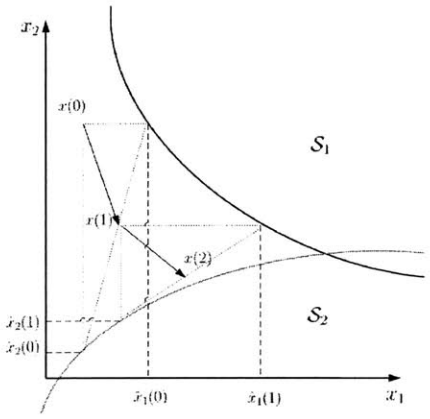
Using a different approach, the Projected Consensus algorithm [38] solves a convex feasibility problem by assigning an agent to each constraint set $\mathcal{S}_1, \cdots, \mathcal{S}_m$ and coordinating the actions of the agents to reach a consensus on a solution that satisfies all the constraints.

At time step t, agent i generates and stores an estimate $r^i(t)$ of x that is constrained to lie in $\mathcal{S}_i$ known only to agent i. Given a (possibly time-varying) communication network between agents, the collective objective of the agents is to cooperatively reach a consensus on a common vector x_* through a sequence of local estimate updates subject to the local constraint sets and local information exchanges with neighboring agents. The algorithm can be formulated as an iterative sequence defined by the recursive formula

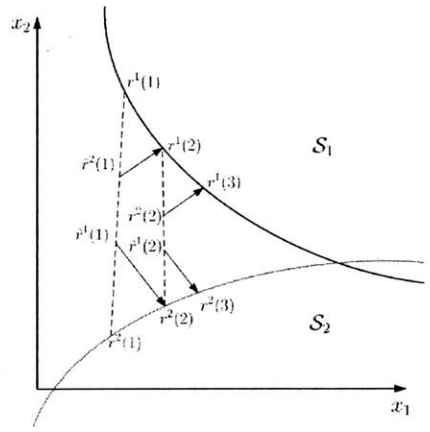
$$r^i(t+1) \;\; = P_{\mathcal{S}_i}\left(\sum_{j=1}^{m} a_j^i(t)\; r^j(t)\right), \quad i = 1, \cdots, m$$



(a) MAP algorithm (3.1)



(b) DCS algorithm (3.4)



(c) CC algorithm (3.6) (3.7)

Figure 3-2: Comparison of the MAP, DCS and CC algorithms for two closed, convex sets $\mathcal{S}_1$ and $\mathcal{S}_2$

3.2.3 Feasibility Algorithms in the Absence of Convexity

The attractive theories of the MAP and Projected Consensus algorithms and their extensive use for solving convex feasibility problems make it tempting to experiment with analogous heuristics for non-convex feasibility problems. In fact, these heuristics are quite popular in practice, in areas such as phase retrieval in image processing [2] and synthesis problems in low-order control design [29].

The analogous non-convex heuristics for the MAP and Projected Consensus algorithms may appear difficult to implement on non-convex problems. A key ingredient in the proofs of these algorithms was to use the non-expansivity of projections, allowing a rich fixed point theory to be applied. Such properties do not hold for general non-convex sets. Furthermore, the projection mapping for non-convex sets will no longer guarantee a unique solution and are hard to compute in general. As consequence, global convergence of either algorithm is no longer achievable. We must be content with a local theory.

Nevertheless, application of MAP for non-convex feasibility problems has been studied extensively and local convergence is guaranteed under assumptions of good geometric property of the sets [34, 35]. We state some of these results.

Definition 3.3. *A closed set $\mathcal{S} \subseteq \mathbb{E}$ is super-regular at $s \in \mathcal{S}$ if, for all $\delta > 0$, any two points x, y sufficiently near s with $y \in \mathcal{S}$ and any point $z \in P_{\mathcal{S}}(x)$ satisfy*

$$\langle y - z \ , \ x - y \rangle \leq \delta \, \|y - z\| \cdot \|x - z\|$$

Definition 3.4. *Two sets $\mathcal{S}, \mathcal{T} \subset \mathbb{E}$ have linearly regular intersection at $x^* \in S \cap T$ if there exist constants $\alpha, \beta > 0$ such that for all $x \in S \cap B(x^*, \alpha)$ and $y \in T \cap B(x^*, \alpha)$, and all $\rho \in (0, \alpha]$, we have*

$$B(0 \, , \, \beta \, \rho) \subseteq ((\mathcal{S} - x) \cap \rho B(0,1)) - ((\mathcal{T} - y) \cap B(0, \rho))$$

Theorem 3.1 ([35]). *Consider closed sets $\mathcal{S}, \mathcal{T} \subseteq \mathbb{E}$ and a point $x^* \in \mathcal{S} \cap \mathcal{T}$. Suppose $\mathcal{S}$ is super-regular at x^*. Furthermore, suppose that $\mathcal{S}$ and $\mathcal{T}$ have linearly regular in-*

tersection at x^*. Then for any initial point sufficiently close to x^*, the MAP algorithm converges to a point in $\mathcal{S} \cap \mathcal{T}$.

Proof. See Theorem 5.16 of [35].

□

The projection sub-problem for some fundamental non-convex sets is also relatively easy. Examples include the set of matrices of some fixed rank and any set defined by a single quadratic equation or inequality, which is analogous to the classical "trust region sub-problem". When the set is defined by one or more quadratic equations or inequalities, the projection sub-problem becomes a Quadratically Constrained Quadratic Program (QCQP). QCQPs are of particular interest because they occur frequently in power flow problems. Although they are non-convex and NP-hard in general, we show in Chapter 2 that a large class of QCQP is computationally tractable.

3.3 Distributed Constraint Satisfaction Algorithm

The appeal of the MAP algorithm for convex feasibility problems lies in its simplicity and ease of implementation, given a subroutine that solves the projection sub-problem efficiently. However, it is not amenable for parallel or distributed implementation. In this section, we propose an extension of the MAP algorithm that solves the feasibility problem for a network in a distributed manner.

The new algorithm is inspired by the coordinate descent method: at time step t, each node i in the network generates an estimate of its variable $x_i(t)$ and solves a projection sub-problem with respect to $x_i(t)$, while keeping other variables fixed. The sub-problem is an optimization problem with respect to a single variable and thus it can be computed more efficiently than the sub-problem considered in the MAP algorithm (3.1). Through a sequence of projections and message exchanges with their neighbors, the nodes update their estimates so that, eventually, all constraints are simultaneously satisfied (see Figure 3-2). We call our algorithm the Distributed Constraint Satisfaction (DCS) algorithm and illustrate it with the following example.

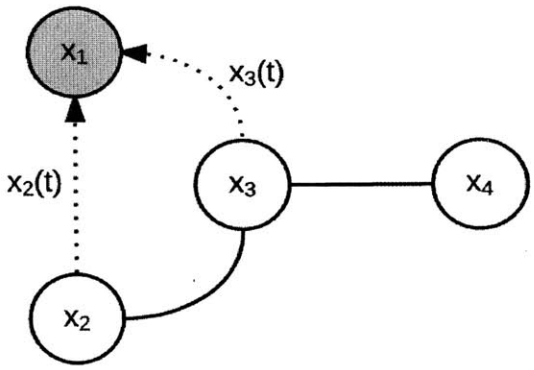


Figure 3-3: An iteration of the DCS algorithm for node 1. Nodes 2 and 3 broadcast their estimates to node 1.

Example 3.3.1. *Consider a feasibility problem for the network in Figure 3-1. At time step t, the neighbors of node 1 broadcast their estimates $x_2(t)$ and $x_3(t)$ to node 1, as depicted in Figure 3-3. To obtain a new estimate, node 1 solves the projection sub-problem*

$$\begin{array}{ll} \underset{x_1}{min} & \|x_1 - x_1(t)\|^2 \\ s.t. & (x_1\ ,\ x_2(t)\ ,\ x_3(t)) \in \mathcal{S}_1 \end{array} \tag{3.3}$$

We denote the feasible set of (3.3) *as $\mathcal{X}_1(t)$ and the solution to be $\widehat{x}_1(t) = P_{\mathcal{X}_1(t)}(x_1(t))$. Node 1 obtains a new estimate using the relation $x_1(t+1) = a_1(t)\widehat{x}_1(t)+(1-a_1(t))x_1(t)$ for some weight $a_1(t)$.*

The other nodes carry out the same operations simultaneously and $\mathcal{X}_2(t)$, $\mathcal{X}_3(t)$ and $\mathcal{X}_4(t)$ can be derived similarly. The process is repeated until all constraints are satisfied.

Summarizing Example 3.3.1, the DCS algorithm generates two sequences of iterates $\{x_i(t)\}$ and $\{\widehat{x}_i(t)\}$ that are defined by the recursive formulas[1]:

[1]We assume that $P_{\mathcal{X}_i(t)}(x_i(t))$ has a unique solution.

For $i = 1, \cdots, m$,

$$\begin{aligned} \widehat{x}_i(t) &= P_{\mathcal{X}_i(t)}\left(x_i(t)\right) \\ x_i(t+1) &= a_i(t)\ \widehat{x}_i(t) + (1 - a_i(t))x_i(t) \end{aligned} \tag{3.4}$$

where $a_i(t)$ are non-negative weights and each $\widehat{x}_i(t)$ is constrained to lie in the set $\mathcal{X}_i(t) \subseteq \mathbb{E}$, which depends on the values of i's neighbors and is known only to node i. In other words, $\mathcal{X}_i(t)$ is the projection of $\mathcal{S}_i$ onto the space of coordinate i, with the neighbors taking on the values $x_j(t)$, $j \in N(i)$. For example, if $i = 1$, then

$$\mathcal{X}_1(t) = \{x_1 \in \mathbb{E} \mid (x_1, x_2(t), \cdots, x_m(t)) \in \mathcal{S}_1\}$$

3.4 Constrained Consensus Algorithms

We will now develop an extension of the Projected Consensus Algorithm that solves the feasibility problem for a network. An agent is associated with a node in the network and its constraint set. Instead of assigning the same set of variables to every agent, each agent is only assigned the variables of the corresponding node and its neighbors. At time step t, each agent (node) i generates the estimate

$$r^i(t) = \left\{x_j^i(t) \ : \ j \in I_i\right\}$$

where $x_j^i(t)$ is the estimate of variable x_j generated by agent i at time step t. Through a sequence of projections and message exchanges with their neighbors, the agents update their estimates so that, eventually, all constraints are simultaneously satisfied (see Figure 3-2). We call this algorithm the Constrained Consensus (CC) algorithm and illustrate it with the following example.

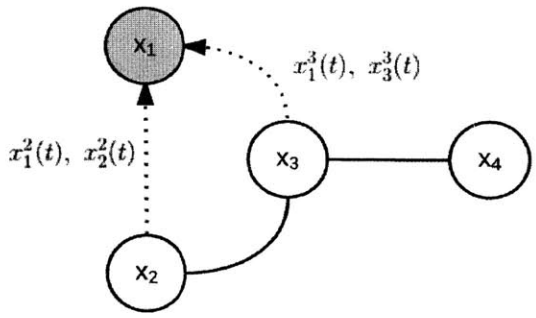


Figure 3-4: An iteration of the CC algorithm for agent 1. Agents 2 and 3 broadcast their estimates to agent 1.

Example 3.4.1. *Consider a feasibility problem for the network in Figure 3-1. At time step t, agent 1 stores the estimate*

$$r^{+1}(t) = \left\{x_j^1(t) \ : \ j \in I_i\right\} = \left\{x_1^1(t)\ ,\ x_2^1(t)\ ,\ x_3^1(t)\right\}$$

The neighbors of agent 1 broadcast their estimates to agent 1, as depicted in Figure 3-4. Agent 1 forms a convex combination of its estimate with the estimates received from other agents using the relation

$$\widehat{r}^{i}(t) = \left\{\widehat{x}_1^1(t)\ ,\ \widehat{x}_2^1(t)\ ,\ \widehat{x}_3^1(t)\right\}$$

where

$$\begin{aligned}
\widehat{x}_1^1(t) &= a_{1,1}^1(t)\ x_1^1(t) + a_{1,2}^1(t)\ x_1^2(t) + a_{1,3}^1(t)\ x_1^3(t) \\
\widehat{x}_2^1(t) &= a_{2,1}^1(t)\ x_2^1(t) + a_{2,2}^1(t)\ x_2^2(t) \\
\widehat{x}_3^1(t) &= a_{3,1}^1(t)\ x_2^1(t) + a_{3,3}^1(t)\ x_1^3(t)
\end{aligned}$$

To obtain its estimate for time step $t+1$, agent 1 solves the projection problem

$$\begin{aligned}
\underset{x_1,x_2,x_3}{min} \quad & \left\|x_1 - \widehat{x}_1^1(t)\right\|^2 + \left\|x_2 - \widehat{x}_2^1(t)\right\|^2 + \left\|x_3 - \widehat{x}_3^1(t)\right\|^2 \\
s.t. \quad & (x_1, x_2, x_3) \in \mathcal{S}_1
\end{aligned} \tag{3.5}$$

We denote the solution of (3.5) *as*

$$\begin{aligned} r_1(t+1) &= P_{\mathcal{S}_1}\left(\widehat{r}_1(t)\right) \\ &= \left\{\widehat{x}_1^1(t+1)\ ,\ \widehat{x}_2^1(t+1)\ ,\ \widehat{x}_3^1(t+1)\right\} \end{aligned}$$

The other agents carry out the same operations simultaneously, and the process is repeated until all constraints are satisfied.

To summarize Example 3.4.1, the CC algorithm first assigns an agent for each node in $\mathcal{V}$. Agent i keeps the set of local decision estimates $\{x_k\ :\ k \in I_i\}$, and at time step t agent i generates the estimate

$$r^i(t) = \left\{x_j^i(t)\ :\ j \in I_i\right\}$$

where $x_k^i(t)$ is the estimate of x_k held by agent i. In general, one or more agents may hold estimates of the same variable. $r^i(t)$ is a vector that is constrained to lie in the set $\mathcal{S}_i \subseteq \mathbb{E}^{|I_i|}$.

The collective goal of the agents is to find a feasible vector x^*, i.e.,

$$x^* = (x_1^*\ ,\ \cdots\ ,\ x_m^*) \in \mathcal{S}$$

through the sequence of local estimate updates and local information exchanges defined as follows[2]:

For $i = 1, \cdots, m$,

$$\widehat{x}_j^i(t) = \sum_{k \in I_i} a_{j,k}^i(t)\ x_j^k(t), \quad j \in I_i \tag{3.6}$$

$$r^i(t+1) = P_{\mathcal{S}_i}\left(\widehat{r}^i(t)\right) \tag{3.7}$$

where $\widehat{r}^i(t) = \left\{\widehat{x}_j^i(t)\ :\ j \in I_i\right\}$ and $a_{j,k}^i(t)$ are non-negative weights such that $0 \leq a_{j,k}^i(t) \leq 1$.

[2]We assume that $P_{\mathcal{R}_i}\left(\widehat{r}^i(t)\right)$ has a unique solution.

3.4.1 Convergence Analysis

The convergence analysis of the CC algorithm for the case where $\mathcal{S}_1, \cdots, \mathcal{S}_m$ are closed and convex and $\mathcal{S}$ is non-empty is inspired by the work of [38]. We will now state a few assumptions that will be required in our analysis.

Assumption 3.1 (Weights rule). *There exists a scalar η, with $0 < \eta < 1$, such that for all i,*

(a) $a^i_{i,i}(t) \geq \eta$

(b) $a^i_{j,k}(t) \geq \eta$ *if* $(j,k) \in \mathcal{E}$

(c) $a^i_{j,k}(t) = 0$ *if* $(j,k) \notin \mathcal{E}$, $i \neq j$ *and* $i \neq k$

Assumption 3.2 (Double stochasticity). *For all i, the weights $a^i_{j,k}(t)$ satisfy*

(a) $\sum_{k=1}^m a^i_{j,k}(t) = 1$ *if* $j \in N(i)$

(b) $\sum_{i=1}^m a^i_{j,k}(t) = 1$ *if* $(j,k) \in \mathcal{E}$

For full details of the analysis, the reader is referred to Appendix A.

3.5 Power Flow Problem: An Application

3.5.1 Problem Formulation

Consider an electric power grid with a set of buses $\mathcal{V} = \{1\ ,\ \cdots\ ,\ m\}$ and a set of lines $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$. Given the $m \times m$ bus admittance matrix $Y := G + \mathbf{i}B$, the power flow problem seeks to find a vector of complex bus voltages $V := V_d + \mathbf{i}V_q$ that is consistent with the power flow equations, is within operational limits and satisfies

complex power demands at every bus. It is formulated as

$$
\begin{aligned}
\text{find} \quad & V_{d,i},\ V_{q,i}, \quad \forall \quad i \in \mathcal{V} \\
\text{s.t.} \quad & P_{G,i}^{\min} \leq V_{d,i} \sum_{j \in I_i} (V_{d,j} G_{ij} - V_{q,j} B_{ij}) + V_{q,i} \sum_{j \in I_i} (V_{q,j} G_{ij} + V_{d,j} B_{ij}) + P_{D,i} \\
& \qquad \leq P_{G,i}^{\max} \quad \forall \quad i \in \mathcal{V} \\
& Q_{G,i}^{\min} \leq V_{q,i} \sum_{j \in I_i} (V_{d,j} G_{ij} - V_{q,j} B_{ij}) - V_{d,i} \sum_{j \in I_i} (V_{q,j} G_{ij} + V_{d,j} B_{ij}) + Q_{D,i} \\
& \qquad \leq Q_{G,i}^{\max} \quad \forall \quad i \in \mathcal{V} \\
& \left(V_i^{\min}\right)^2 \leq V_{d,i}^2 + V_{q,i}^2 \leq (V_i^{\max})^2 \quad \forall \quad i \in \mathcal{V}
\end{aligned}
$$

where generator limits $P_{G,i}^{\min}$, $P_{G,i}^{\max}$, $Q_{G,i}^{\min}$, real and reactive power demands $P_{D,i}$ and $Q_{D,i}$, and $Q_{G,i}^{\max}$ and voltage magnitude limits $V_i^{\min}$ and $V_i^{\max}$ are assumed to be known. The reader is referred to Appendix B for a detailed derivation of the power flow problem and its parameters.

3.5.2 Implementation

It is clear that the power flow problem is a feasibility problem for a network, where the nodes are represented by buses and the constraints are local to the nodes and their neighbors. We shall now show that the distributed feasibility algorithms developed in Sections 3.3 and 3.4 are well-defined for the power flow problem because the projection sub-problems can be formulated as Quadratically Constrained Quadratic Programs (QCQPs) that can be solved efficiently.

Each bus i has the local variable

$$x_i = \left(V_{d,i}, V_{q,i}\right) \in \mathbb{R}^2$$

such that the state of the transmission system is given by

$$x = \left(x_1, \cdots, x_m\right) \in \mathbb{R}^{2m}$$

DCS algorithm

At time step t, bus i generates the estimate

$$x_i(t) = \Big(V_{d,i}(t), V_{q,i}(t)\Big)$$

and solves the sub-problem

$$\begin{aligned}
\min \quad & \left\|\widehat{V}_{d,i} - V_{d,i}(t)\right\|^2 + \left\|\widehat{V}_{q,i} - V_{q,i}(t)\right\|^2 \\
\text{s.t.} \quad & P_{G,i}^{\min} \le \widehat{V}_{d,i} \sum_{j\in N_i} \left(V_{d,j}(t)\ G_{ij} - V_{q,j}(t)\ B_{ij}\right) + \widehat{V}_{q,i} \sum_{j\in N_i} \left(V_{q,j}(t)\ G_{ij} + V_{d,j}(t)\ B_{ij}\right) \\
& \qquad + \widehat{V}_{d,i}\left(\widehat{V}_{d,i}\ G_{ii} - \widehat{V}_{q,i}\ B_{ii}\right) + \widehat{V}_{q,i}\left(\widehat{V}_{q,i}\ G_{ii} + \widehat{V}_{d,i}\ B_{ii}\right) + P_{D,i} \le P_{G,i}^{\max} \\
& Q_{G,i}^{\min} \le \widehat{V}_{q,i} \sum_{j\in N_i} \left(V_{d,j}(t)\ G_{ij} - V_{q,j}(t)\ B_{ij}\right) - \widehat{V}_{d,i} \sum_{j\in N_i} \left(V_{q,j}(t)\ G_{ij} + V_{d,j}(t)\ B_{ij}\right) \\
& \qquad + \widehat{V}_{q,i}\left(\widehat{V}_{d,i}\ G_{ii} - \widehat{V}_{q,i}\ B_{ii}\right) - \widehat{V}_{d,i}\left(\widehat{V}_{q,i}\ G_{ii} + \widehat{V}_{d,i}\ B_{ii}\right) + Q_{D,i} \le Q_{G,i}^{\max} \\
& \left(V_i^{\min}\right)^2 \le \widehat{V}_{d,i}^2 + \widehat{V}_{q,i}^2 \le \left(V_i^{\max}\right)^2
\end{aligned} \tag{3.8}$$

The optimization variables are indicated by a "hat" (namely $\widehat{V}_{d,i}$ and $\widehat{V}_{q,i}$), while the other variables (e.g. $V_{d,j}(t)$ and $V_{q,j}(t)$) are given. Notice that (3.8) is a Quadratically Constrained Quadratic Program and the equivalent quadratic matrices Q_i (see (2.1)) are diagonal. Assuming that (3.8) satisfies the regularity condition, we can apply Theorem 2.3 to deduce that the convex relaxation of (3.8) will be exact.

CC algorithm

Each bus is assigned an agent. At time step t, agent i generates the estimate

$$r^i(t) = \left\{x_j^i(t)\ :\ j \in I_i\right\}$$

and solves the sub-problem

$$
\begin{aligned}
\min \quad & \sum_{j\in I_i} \left\| \widehat{V}_{d,j} - V_{d,j}(t) \right\|^2 + \sum_{j\in I_i} \left\| \widehat{V}_{q,j} - V_{q,j}(t) \right\|^2 \\
\text{s.t.} \quad & P_{G,i}^{\min} \le \widehat{V}_{d,i} \sum_{j\in I_i} \left(\widehat{V}_{d,j} G_{ij} - \widehat{V}_{q,j} B_{ij} \right) + \widehat{V}_{q,i} \sum_{j\in I_i} \left(\widehat{V}_{q,j}(t) G_{ij} + \widehat{V}_{d,j}(t) B_{ij} \right) + P_{D,i} \le P_{G,i}^{\max} \\
& Q_{G,i}^{\min} \le \widehat{V}_{q,i} \sum_{j\in I_i} \left(\widehat{V}_{d,j} G_{ij} - \widehat{V}_{q,j} B_{ij} \right) - \widehat{V}_{d,i} \sum_{j\in I_i} \left(\widehat{V}_{q,j} G_{ij} + \widehat{V}_{d,j} B_{ij} \right) + Q_{D,i} \le Q_{G,i}^{\max} \\
& \left(V_i^{\min}\right)^2 \le \widehat{V}_{d,i}^2 + \widehat{V}_{q,i}^2 \le \left(V_i^{\max}\right)^2
\end{aligned}
\tag{3.9}
$$

The optimization variables are indicated by a "hat" (e.g. $\widehat{V}_{d,i}$ and $\widehat{V}_{q,i}$), while the other variables (e.g. $V_{d,j}(t)$ and $V_{q,j}(t)$) are given. Notice that (3.9) is a Quadratically Constrained Quadratic Program (QCQP). Although there are no guarantees for the exactness of the convex relaxations of (3.9), in practice they are found to be exact by verifying the conditions of Lemma 2.1.

3.5.3 Numerical Results

We validate the CC and DCS algorithms developed in Sections 3.4 and 3.3 using IEEE test beds from the Power Systems Test Case Archives [17] and MATPOWER archives by [55].

The algorithms are implemented in MATLAB and a MATLAB interface to CVX provided by [27] and [28] is used to solve the projection sub-problems. In all cases, the algorithms are initialized with a flat profile (i.e., $V_{d,i} = 1$ and $V_{q,i} = 0$ for $i = 1, \cdots, m$) and the criteria for declaring convergence is that norm of update differences are less than 10^{-3}, with the further requirement that constraints must be satisfied within a precision of 10^{-3}. For comparison, the same problem is also solved with the MAP algorithm (3.1).

The results of a serial implementation of the CC, DCS and MAP algorithms on an Intel Core i7-3720QM 2.60 GHz Processor are summarized in Table 3.5.3. In our serial implementation, a "cycle" of the above algorithms completes when all nodes have consecutively solved their corresponding projection sub-problems.

The DCS algorithm converges in the shortest time all test cases, largely due to the ease of solving the projection sub-problems (involving the least number of variables). The CC algorithm has an easier projection sub-problem compared to the MAP algorithm, but appears to perform worse than the MAP algorithm. In practice, the CC and DCS algorithms will be executed in parallel across processors at each node, thus they can show even greater performance improvements over the MAP algorithm.

Number of buses	Algorithms		
	MAP	DCS	CC
5	59, 6.0s	111, 0.9s	280, 20.6s
9	25, 5.4s	79, 1.7s	73, 10.8s
14	216, 76.2s	155, 8.3s	345, 120.4s
30	418, 310.8s	396, 10.6s	446, 305.0s

Table 3.1: Comparison of algorithms for solving the power flow problem. (Number of cycles, Total CPU time)

Chapter 4

Secure Estimation for Cyber-Physical Systems

Cyber-physical are computer-based systems that monitor and control physical processes using embedded sensors, actuators, control processing units and communication devices. They characterize many of the critical infrastructure that sustain our modern society, such as electric power grids, oil and natural gas distribution, water treatment and transportation systems. The disruption of their operation can have disastrous consequences on public health and the economy.

The operation of a cyber-physical system can be modeled as a dynamical system that is supported by numerous feedback mechanisms. These mechanisms rely heavily on state estimation algorithms to work correctly and an entire field of research has been dedicated to improving these algorithms. As more cyber-physical systems are connected to the internet for remote monitoring and control, they become vulnerable to attacks on their communication channels, while their large scales make it challenging to secure every system component. Failure to check these attacks may corrupt state estimates and lead to physical consequences in the forms of faults and failures.

This chapter considers the problem of estimating the states of a noisy and uncertain cyber-physical system that is subject to data injection attacks [14, 37, 42] on its actuators and sensors. By leveraging principles of robust optimization, a novel robust and resilient state estimator that can be formulated as a convex optimization problem

is proposed. The effectiveness of our estimator is demonstrated in simulations of an IEEE 14-bus system.

4.1 Problem Statement

A noisy and uncertain cyber-physical system that is under attack can be modeled by the following linear, time invariant (LTI) dynamical system:

$$\begin{aligned} x_{k+1} &= \widetilde{A}\, x_k + \widetilde{B}\, (u_k + d_k) + w_k \\ y_k &= \widetilde{C}\, x_k + \widetilde{D}\, (u_k + d_k) + e_k + v_k \end{aligned} \tag{4.1}$$

where $x_k \in \mathbb{R}^n$ is the state vector at time k, $u_k \in \mathbb{R}^m$ is a known input vector and $y_k \in \mathbb{R}^p$ is the measurement vector, $w_k \in \mathbb{R}^n$ and $v_k \in \mathbb{R}^p$ are process and measurement noise signals. The data injection attacks carried out by the adversary are generalized by the attack signals $d_k \in \mathbb{R}^m$ and $e_k \in \mathbb{R}^p$ that are injected into the actuators and sensors, respectively. The system parameters $\widetilde{A} := A+\delta A$, $\widetilde{B} := B+\delta B$, $\widetilde{C} := C + \delta C$ and $\widetilde{D} := D + \delta D$ each consists of a known part (A, B, C and D) as well as an unknown part (δA, δB, δC and δD) that represents (possibly time-varying) modeling errors. We shall henceforth refer to the modeling errors and noise signals as uncertainties, and the attack signals as attacks (on actuators and sensors).

We will assume in this paper that all pairs $(\widetilde{A}, \widetilde{C})$ are observable and that the known inputs u_k are independent of x_0 (i.e., we consider the closed loop dynamics in which the dependence of u_k on x_0 is already incorporated into the system). In addition, adversary attacks a fixed subset of the sensors and actuators. Note that if sensor $i \in \{1, \cdots, p\}$ is not attacked then necessarily $e_k^{(i)} = 0$ for all time steps k; otherwise $e_k^{(i)}$ can take any value, i.e., the attack signals are arbitrary and unpredictable. The same observation holds for the attacks on actuators d_k.

The objective of this paper is *robust and resilient estimation*: given T corrupted measurements $y_0, y_1, \cdots, y_{T-1}$, we wish to obtain estimates for the states $x_0, \cdots, x_{T-1}$ that are 1) robust to uncertainties, and 2) resilient to attacks.

4.2 Preliminary Material

4.2.1 Known System with Sensor Attacks Only

We begin with the following simplified system:

$$\begin{aligned} x_{k+1} &= A\, x_k \\ y_k &= C\, x_k + e_k \end{aligned} \tag{4.2}$$

The goal of the estimator is to reconstruct the initial state x_0 of the plant from the corrupted measurements $y_0, \cdots, y_{T-1}$. Since A is known, the remaining states $x_1, \cdots, x_{T-1}$, can be reconstructed from x_0 using (4.2) and therefore it is sufficient for the estimator to reconstruct x_0.

The system (4.2) can be written compactly as

$$\mathrm{Y} = \Phi(x_0) + \mathrm{E}$$

where $\mathrm{Y} := \begin{bmatrix} y_0 & , & \cdots & , & y_{T-1} \end{bmatrix} \in \mathbb{R}^{p \times T}$, $\mathrm{E} := \begin{bmatrix} e_0 & , & \cdots & , & e_{T-1} \end{bmatrix} \in \mathbb{R}^{p \times T}$ and Φ is a linear map defined by $\Phi : \mathbb{R}^n \to \mathbb{R}^{p \times T}$, $\Phi(x) = \begin{bmatrix} Cx & CAx & \cdots & CA^{T-1}x \end{bmatrix}$.

The optimal estimator of (4.2) is given by [22] as

$$x_0 = \arg\min_{x_0 \in \mathbb{R}^n} \|\mathrm{E}\|_{\ell_0} = \arg\min_{x_0 \in \mathbb{R}^n} \|\mathrm{Y} - \Phi(x_0)\|_{\ell_0} \tag{4.3}$$

It has been shown that, if (A, C) is observable, then the maximum number of attacked sensors (such that x_0 can be reconstructed exactly) is $\lceil \frac{p}{2} - 1 \rceil$. Additionally, the maximum number of correctable errors cannot increase beyond a window size of $T = n$ measurements (a consequence of Cayley-Hamilton theorem).

However, since (4.3) is intractable (NP-hard), we consider a convex relaxation of the optimal estimator using a "mixed" ℓ_1/ℓ_r norm that is also used in the compressed sensing literature [20], i.e., the relaxed estimator minimizes the ℓ_1/ℓ_r norm of E:

$$\hat{x}_0 = \arg\min_{x_0 \in \mathbb{R}^n} \|\mathrm{E}\|_{\ell_1/\ell_r} = \arg\min_{x_0 \in \mathbb{R}^n} \|\mathrm{Y} - \Phi(x_0)\|_{\ell_1/\ell_r} \tag{4.4}$$

The "hat" on $\hat{x}_0$ denotes that the relaxed estimator (4.4) generates an estimate of x_0, whereas the optimal estimator (4.3) recovers the exact x_0. The relaxed estimator (4.4) has been demonstrated to generate estimates that are close to the exact solutions in [22].

4.2.2 Known System with Actuator and Sensor Attacks

Next, we consider the following system

$$\begin{aligned} x_{k+1} &= Ax_k + B\left(u_k + d_k\right) \\ y_k &= Cx_k + D\left(u_k + d_k\right) + e_k \end{aligned} \tag{4.5}$$

(4.5) can be written compactly as

$$\mathrm{Y} = \Phi(x_0) + \Theta(\mathrm{U}) + \Theta(\mathrm{D}) + \mathrm{E}$$

where $\mathrm{Y} := \begin{bmatrix} y_0 & , & \cdots & , & y_{T-1} \end{bmatrix} \in \mathbb{R}^{p\times T}$, and $\mathrm{U} \in \mathbb{R}^{m\times T}$, $\mathrm{D} \in \mathbb{R}^{m\times T}$ and $\mathrm{E} \in \mathbb{R}^{p\times T}$ are defined similarly. $\Phi : \mathbb{R}^n \to \mathbb{R}^{p\times T}$ and $\Theta : \mathbb{R}^{m\times T} \to \mathbb{R}^{p\times T}$ are linear maps defined by

$$\begin{aligned} \Phi(x) &= \begin{bmatrix} Cx & , & CAx & , & \cdots & , & CA^{T-1}x \end{bmatrix} \\ \Theta(\mathrm{U}) &= \begin{bmatrix} Du_0 & , & CBu_0 + Du_1 & , & \cdots & , & C\sum_{i=0}^{T-2} A^{T-2-i}Bu_i + Du_{T-1} \end{bmatrix} \\ \Theta(\mathrm{D}) &= \begin{bmatrix} Dd_0 & , & CBd_0 + Dd_1 & , & \cdots & , & C\sum_{i=0}^{T-2} A^{T-2-i}Bd_i + Dd_{T-1} \end{bmatrix} \end{aligned}$$

For (4.5), the optimal estimator is given by [22] as

$$(x_0, \mathrm{D}) = \underset{\substack{x_0\in\mathbb{R}^n \\ \mathrm{D}\in\mathbb{R}^{m\times T}}}{\arg\min} \ \|\mathrm{Y} - \Phi(x_0) - \Theta(\mathrm{U}) - \Theta(\mathrm{D})\|_{\ell_0} + \|\mathrm{D}\|_{\ell_0} \tag{4.6}$$

In contrast to (4.3), the optimal estimator in (4.6) has to generate the initial state x_0 as well as the actuator attacks D so that the remaining states $x_1, \cdots, x_{T-1}$ can be recovered using (4.5).

Similar to (4.4), the following convex relaxation of (4.6) is considered:

$$(\hat{x}_0, \hat{\mathrm{D}}) = \underset{\substack{x_0 \in \mathbb{R}^n \\ \mathrm{D} \in \mathbb{R}^{m \times T}}}{\arg\min} \|\mathrm{Y} - \Phi(x_0) - \Theta(\mathrm{U}) - \Theta(\mathrm{D})\|_{\ell_1/\ell_r} + \lambda \|\mathrm{D}\|_{\ell_1/\ell_r} \tag{4.7}$$

where λ is a tuning parameter. Since the system parameters are known, the estimates of the remaining states $\hat{x}_1, \cdots, \hat{x}_{T-1}$ can be obtained using $\hat{x}_0$, estimates of the actuator attacks $\hat{\mathrm{D}} := \left[\hat{d}_0, \cdots, \hat{d}_{T-1}\right]$ and (4.5).

4.2.3 Equivalence of Robust Regression and ℓ_q-Regularization

A useful theorem that we shall make use of in our design of a robust estimator is the equivalence of robust regression and ℓ_q-regularization for the (ℓ_q, ℓ_r) subordinate norm.

Theorem 4.1 (Equivalence of Robust Regression and ℓ_q-Regularization [6, Corollary 1]). *Let $\delta\Psi$ be an uncertain matrix belonging to the uncertainty set $\mathcal{U}_{(\ell_q,\ell_r)} = \{\delta\Psi : \|\delta\Psi\|_{(\ell_q,\ell_r)} \leq \rho\}$. If $q, r \in [1, \infty]$ then for for some matrix Ψ and vectors $\mathbf{y}$, $\boldsymbol{\beta}$, we have*

$$\max_{\delta\Psi \in \mathcal{U}_{(\ell_q,\ell_r)}} \|\mathbf{y} - (\Psi + \delta\Psi)\boldsymbol{\beta}\|_{\ell_r} = \|\mathbf{y} - \Psi\boldsymbol{\beta}\|_{\ell_r} + \rho\|\boldsymbol{\beta}\|_{\ell_q}.$$

Proof. See [6, Corollary 1].

□

It is worth noting that there are theorems similar to Theorem 4.1 for the Schatten and Frobenius norms [3, 6, 50].

4.3 Robust and Resilient State Estimation

Now we are ready to consider the system in (4.1), restated below

$$\begin{aligned} x_{k+1} &= \widetilde{A}\, x_k + \widetilde{B}\, (u_k + d_k) + w_k \\ y_k &= \widetilde{C}\, x_k + \widetilde{D}\, (u_k + d_k) + e_k + v_k \end{aligned} \tag{4.1}$$

(4.1) can be compactly written as

$$\mathrm{Y} = \widetilde{\Phi}(x_0) + \widetilde{\Theta}(\mathrm{U}) + \widetilde{\Theta}(\mathrm{D}) + \widetilde{\Upsilon}(\mathrm{W}, \mathrm{V}) + \mathrm{E} \tag{4.8}$$

where $\mathrm{Y} := \begin{bmatrix} y_0 & \cdots & y_{T-1} \end{bmatrix} \in \mathbb{R}^{p\times T}$, and $\mathrm{D} \in \mathbb{R}^{m\times T}$, $\mathrm{U} \in \mathbb{R}^{p\times T}$, $\mathrm{W} \in \mathbb{R}^{n\times T}$, $\mathrm{V} \in \mathbb{R}^{p\times T}$ and $\mathrm{E} \in \mathbb{R}^{p\times n}$ are defined similarly. $\widetilde{\Phi}$, $\widetilde{\Theta}$ and $\widetilde{\Upsilon}$ are linear maps $\widetilde{\Phi} : \mathbb{R}^n \to \mathbb{R}^{p\times T}$, $\widetilde{\Theta} : \mathbb{R}^{m\times T} \to \mathbb{R}^{p\times T}$ and $\widetilde{\Upsilon} : \mathbb{R}^{n\times T} \times \mathbb{R}^{p\times T} \to \mathbb{R}^{p\times T}$ defined as

$$\begin{aligned} \widetilde{\Phi}(x) &= \begin{bmatrix} \widetilde{C}x & , & \widetilde{C}\widetilde{A}x & , & \cdots & , & \widetilde{C}\widetilde{A}^{T-1}x \end{bmatrix} \\ \widetilde{\Theta}(\mathrm{U}) &= \begin{bmatrix} \widetilde{D}u_0 & , & \widetilde{C}\widetilde{B}u_0 + \widetilde{D}u_1 & , & \cdots & , & \widetilde{C}\sum_{i=0}^{T-2}\widetilde{A}^{T-2-i}\widetilde{B}u_i + \widetilde{D}u_{T-1} \end{bmatrix} \\ \widetilde{\Theta}(\mathrm{D}) &= \begin{bmatrix} \widetilde{D}d_0 & , & \widetilde{C}\widetilde{B}d_0 + \widetilde{D}d_1 & , & \cdots & , & \widetilde{C}\sum_{i=0}^{T-2}\widetilde{A}^{T-2-i}\widetilde{B}d_i + \widetilde{D}d_{T-1} \end{bmatrix} \\ \widetilde{\Upsilon}(\mathrm{W}, \mathrm{V}) &= \begin{bmatrix} v_0 & , & \widetilde{C}w_0 + v_1 & , & \cdots & , & \widetilde{C}\sum_{i=0}^{T-2}\widetilde{A}^{T-2-i}w_i + v_{T-1} \end{bmatrix}. \end{aligned} \tag{4.9}$$

In light of the uncertain parameters in (4.1), we consider the robustification of the estimator in (4.7) by using the compact representation in (4.8), i.e.,

$$\begin{aligned} (\hat{x}_0, \hat{\mathrm{D}}) &= \underset{\substack{x_0\in\mathbb{R}^n \\ \mathrm{D}\in\mathbb{R}^{m\times T}}}{\arg\min} \ \max_{\delta\Psi\in\mathcal{U}_{(\ell_q,\ell_r)}} \ \|\mathrm{E}\|_{\ell_1/\ell_r} + \lambda\|\mathrm{D}\|_{\ell_1/\ell_r} \\ &= \underset{\substack{x_0\in\mathbb{R}^n \\ \mathrm{D}\in\mathbb{R}^{m\times T}}}{\arg\min} \ \max_{\delta\Psi\in\mathcal{U}_{(\ell_q,\ell_r)}} \ \left\|\mathrm{Y} - \widetilde{\Phi}(x_0) - \widetilde{\Theta}(\mathrm{U}) - \widetilde{\Theta}(\mathrm{D}) - \widetilde{\Upsilon}(\mathrm{W}, \mathrm{V})\right\|_{\ell_1/\ell_r} + \lambda\|\mathrm{D}\|_{\ell_1/\ell_r} \end{aligned} \tag{4.10}$$

for some tuning parameter λ and some uncertain $\delta\Psi$ belonging to the uncertainty set $\mathcal{U}_{(\ell_q,\ell_r)}$. In Section 4.3.1, we will provide two formulas for $\delta\Psi$ and $\mathcal{U}_{(\ell_q,\ell_r)}$ that will lead to tractable formulations of (4.10). It is noteworthy that by substituting E into

the objective function, we have avoided equality constraints that are known in the robust optimization community to oftentimes cause infeasibility.

Even with the estimates $(\hat{x}_0, \hat{\mathrm{D}})$, we cannot obtain $\hat{x}_1, \cdots, \hat{x}_{T-1}$ using (4.1) because the system parameters $\widetilde{A}, \widetilde{B}$ and noise signals w_k are unknown. In Section 4.3.2, we develop a robust estimator for the states $x_1, \cdots, x_{T-1}$ using $(\hat{x}_0, \hat{\mathrm{D}})$.

4.3.1 Robust and Resilient Estimation of x_0 and Actuator Attacks

Row-wise Uncertainty Sets

Notice that we can use the definition of the ℓ_1/ℓ_r norm to rewrite (4.10) as the sum of rows:

$$(\hat{x}_0, \hat{\mathrm{D}}) = \underset{\substack{x_0 \in \mathbb{R}^n \\ \mathrm{D} \in \mathbb{R}^{m \times T}}}{\arg\min} \max_{\delta\Psi \in \mathcal{U}_{(\ell_q, \ell_r)}} \sum_{i=1}^{p} \left\| \left(\mathrm{Y} - \widetilde{\Phi}(x_0) - \widetilde{\Theta}(\mathrm{U}) - \widetilde{\Theta}(\mathrm{D}) - \widetilde{\Upsilon}(\mathrm{W}, \mathrm{V}) \right)_{(i,\cdot)} \right\|_{\ell_r} + \lambda \|\mathrm{D}\|_{\ell_1/\ell_r} \tag{4.11}$$

We first consider the sub-problem for the i-th row of E, i.e.,

$$\max_{\delta\Psi_i \in \mathcal{U}_{i,(\ell_q, \ell_r)}} \|\mathrm{E}\|_{\ell_r} = \max_{\delta\Psi_i \in \mathcal{U}_{i,(\ell_q, \ell_r)}} \left\| \left(\mathrm{Y} - \widetilde{\Phi}(x_0) - \widetilde{\Theta}(\mathrm{U}) - \widetilde{\Theta}(\mathrm{D}) - \widetilde{\Upsilon}(\mathrm{W}, \mathrm{V}) \right)_{(i,\cdot)} \right\|_{\ell_r} \tag{4.12}$$

for some uncertain matrix $\delta\Psi_i$ belonging to the uncertainty set $\mathcal{U}_{i,(\ell_q, \ell_r)}$ that we will now define.

It is helpful to consider another compact representation of (4.1):

$$\mathbf{y} = \widetilde{\mathcal{O}} x_0 + \widetilde{\mathcal{J}}_u(\mathbf{u} + \mathbf{d}) + \widetilde{\mathcal{J}}_w \mathbf{w} + \mathbf{e} + \mathbf{v} \tag{4.13}$$

where $\mathbf{y} := \mathrm{vec}(\mathrm{Y})$, $\mathbf{u} := \mathrm{vec}(\mathrm{U})$, $\mathbf{d} := \mathrm{vec}(\mathrm{D})$, $\mathbf{e} := \mathrm{vec}(\mathrm{E})$, $\mathbf{w} := \mathrm{vec}(\mathrm{W})$ and

$\mathbf{v} := \mathrm{vec}(\mathrm{V})$, as well as the following observability and invertibility matrices

$$\widetilde{\mathcal{O}} = \begin{bmatrix} \widetilde{C}^\top & (\widetilde{C}\widetilde{A})^\top & (\widetilde{C}\widetilde{A}^2)^\top & \dots & (\widetilde{C}\widetilde{A}^{T-1})^\top \end{bmatrix}^\top,$$

$$\widetilde{\mathcal{J}}_u = \begin{bmatrix} \widetilde{D} & 0 & 0 & \dots & 0 \\ \widetilde{C}\widetilde{B} & \widetilde{D} & 0 & \dots & 0 \\ \widetilde{C}\widetilde{A}\widetilde{B} & \widetilde{C}\widetilde{B} & \widetilde{D} & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \widetilde{C}\widetilde{A}^{T-2}\widetilde{B} & \widetilde{C}\widetilde{A}^{T-3}\widetilde{B} & \widetilde{C}\widetilde{A}^{T-4}\widetilde{B} & \dots & \widetilde{D} \end{bmatrix}, \quad \widetilde{\mathcal{J}}_w = \begin{bmatrix} 0 & 0 & 0 & \dots & 0 \\ \widetilde{C} & 0 & 0 & \dots & 0 \\ \widetilde{C}\widetilde{A} & \widetilde{C} & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \widetilde{C}\widetilde{A}^{T-2} & \widetilde{C}\widetilde{A}^{T-3} & \widetilde{C}\widetilde{A}^{T-4} & \dots & 0 \end{bmatrix}$$

The matrices $\mathcal{O}$ and $\mathcal{J}_u$ are defined in a similar fashion to (4.9) with the nominal system matrices A, B, C and D. We also define $\delta\mathcal{O} := \widetilde{\mathcal{O}} - \mathcal{O}$ and $\delta\mathcal{J}_u := \widetilde{\mathcal{J}}_u - \mathcal{J}_u$.

Definition 4.1 (Row-wise uncertainty sets). *Let*

$$\delta\Psi_i := \begin{bmatrix} (\delta\mathcal{O})_i & (\delta\mathcal{J}_u)_i & (\delta\mathcal{J}_u)_i & (\widetilde{\mathcal{J}}_w)_i\mathbf{w} + (\mathbf{v})_i \end{bmatrix} \tag{4.14}$$

be an uncertain matrix belonging to the uncertainty set $\mathcal{U}_{i,(\ell_q,\ell_r)} = \{\delta\Psi_i : \|\delta\Psi_i\|_{(\ell_q,\ell_r)} \leq \rho_i\}$ *with* $(M)_i$ *denoting the sub-matrix of* M *consisting of only the* $(i+jp)$*-th rows of* M *for* $j = 0, \cdots, T-1$. *(e.g.,* $(\widetilde{\mathcal{O}})_i := \begin{bmatrix} \widetilde{C}_{(i,\cdot)} & (\widetilde{C}\widetilde{A})_{(i,\cdot)} & (\widetilde{C}\widetilde{A}^2)_{(i,\cdot)} & \dots & (\widetilde{C}\widetilde{A}^{T-1})_{(i,\cdot)} \end{bmatrix}^\top$*).*

Lemma 4.1. *Let* $\delta\Psi_i$ *and* $\mathcal{U}_{i,(\ell_q,\ell_r)}$ *be defined according to Definition 4.1. In addition, we define*

$$\Psi_i := \begin{bmatrix} (\mathcal{O})_i & (\mathcal{J}_u)_i & (\mathcal{J}_u)_i & \mathbf{0}_{T\times 1} \end{bmatrix}$$
$$\boldsymbol{\beta} := \begin{bmatrix} x_0^\top & \mathbf{u}^\top & \mathbf{d}^\top & 1 \end{bmatrix}^\top$$

Then, for any $q, r \in [1, \infty]$, (4.12) *is equivalent to*

$$\max_{\delta\Psi_i \in \mathcal{U}_{i,(\ell_q,\ell_r)}} \left\| \left(\mathrm{Y} - \widetilde{\Phi}(x_0) - \widetilde{\Theta}(\mathrm{U}) - \widetilde{\Theta}(\mathrm{D}) - \widetilde{\Upsilon}(\mathrm{W}, \mathrm{V}) \right)_{(i,\cdot)} \right\|_{\ell_r} = \|\mathrm{Y}_{(i,\cdot)} - \Psi_i\boldsymbol{\beta}\|_{\ell_r} + \rho_i\|\boldsymbol{\beta}\|_{\ell_q}$$

Proof. We can rewrite $\mathrm{E}_{(i,\cdot)}$ as follows

$$\mathrm{E}_{(i,\cdot)} = \left(\mathrm{Y} - \widetilde{\Phi}(x_0) - \widetilde{\Theta}(\mathrm{U}) - \widetilde{\Theta}(\mathrm{D}) - \widetilde{\Upsilon}(\mathrm{W}, \mathrm{V}) \right)_{(i,\cdot)} = \mathrm{Y}_{(i,\cdot)} - (\Psi_i + \delta\Psi_i)\boldsymbol{\beta}.$$

The result follows by application of Theorem 4.1 on (4.12).

□

Assumption 4.1 (Uncoupled uncertainty sets). *Let $\delta\Psi_i$ and $\mathcal{U}_{i,(\ell_q,\ell_r)}$ be defined according to Definition 4.1. We assume that the uncertainty sets $\mathcal{U}_{i,(\ell_q,\ell_r)}$, $i = 1, \cdots, p$, are uncoupled.*

Now we are ready to develop a robust estimator for (4.10).

Proposition 4.1 (Robust Estimation of x_0 with ℓ_1/ℓ_r relaxation and ℓ_q-regularization). *Let Assumption 4.1 hold, and let $\delta\Psi \in \mathcal{U}_{(\ell_q,\ell_r)}$ represent $\delta\Psi_i \in \mathcal{U}_{i,(\ell_q,\ell_r)}$ for $i = 1, \cdots, p$. Then, for any $q, r \in [1, \infty)$, the robust estimator is equivalent to the following constrained optimization problem*

$$
\begin{aligned}
(\hat{x}_0, \hat{\mathrm{D}}) = \quad & \underset{\substack{x_0 \in \mathbb{R}^n \\ \mathrm{D} \in \mathbb{R}^{m \times T} \\ \boldsymbol{\beta} \in \mathbb{R}^{n+2mT+1}}}{\arg\min} \quad \rho \|\boldsymbol{\beta}\|_{\ell_q} + \|\mathrm{Y} - \Phi(x_0) - \Theta(\mathrm{U}) - \Theta(\mathrm{D})\|_{\ell_1/\ell_r} + \lambda \|\mathrm{D}\|_{\ell_1/\ell_r} \\
& \text{s.t.} \quad \boldsymbol{\beta}_{1:n} = x_0 \\
& \qquad \boldsymbol{\beta}_{n+1:n+mT} = \mathrm{vec}(\mathrm{U}) \\
& \qquad \boldsymbol{\beta}_{n+mT+1:n+2mT} = \mathrm{vec}(\mathrm{D}) \\
& \qquad \boldsymbol{\beta}_{n+2mT+1} = 1
\end{aligned}
\tag{4.15}
$$

with $\rho := \sum_{i=1}^{p} \rho_i$. $\lambda > 0$ is a tuning parameter that controls the relative weight between the penalty on errors corresponding to attacks on sensors and actuators.

Proof. This proposition follows the repeated application of Lemma 4.1 and noticing that $Y_{(i,\cdot)} - \Psi_i \boldsymbol{\beta} = (\mathrm{Y} - \Phi(x_0) - \Theta(\mathrm{U}) - \Theta(\mathrm{D}))_{(i,\cdot)}$.

□

We next consider the case $\ell_q = \ell_1$, which significantly simplifies the robust estimator.

Corollary 4.1 (Robust Estimation of x_0 with ℓ_1/ℓ_r relaxation and ℓ_1-regularization). *Let Assumption 4.1 hold, and let $\delta\Psi \in \mathcal{U}_{(\ell_1,\ell_r)}$ denote $\delta\Psi_i \in \mathcal{U}_{i,(\ell_1,\ell_r)}$, $i = 1, \cdots, p$.*

Then, for any $r \in [1, \infty]$,

$$(\hat{x}_0, \hat{\mathrm{D}}) = \underset{\substack{x_0 \in \mathbb{R}^n \\ \mathrm{D} \in \mathbb{R}^{m \times T}}}{\arg\min}\ \rho \|x_0\|_{\ell_1} + \|\mathrm{Y} - \Phi(x_0) - \Theta(\mathrm{U}) - \Theta(\mathrm{D})\|_{\ell_1/\ell_r} + \rho \|\mathrm{D}\|_{\ell_1/\ell_1} + \lambda \|\mathrm{D}\|_{\ell_1/\ell_r}$$

with $\lambda > 0$ *as a tuning parameter that controls the relative weight between the penalty on errors corresponding to attacks on sensors and actuators while* $\rho = \sum_{i=1}^{p} \rho_i$, *where* ρ_i *is a robust parameter for the uncertainty set* $\mathcal{U}_{i,(\ell_q,\ell_r)}$ *(see Definition 4.1).*

Proof. Noting that

$$\begin{aligned} \|\boldsymbol{\beta}\|_{\ell_1} &= \|x_0\|_{\ell_1} + \|\mathbf{u}\|_{\ell_1} + \|\mathbf{d}\|_{\ell_1} + 1, \\ \|\mathbf{d}\|_{\ell_1} &= \|\mathrm{D}\|_{\ell_1/\ell_1}, \text{ and} \\ Y_{(i,\cdot)} - \Psi_i \boldsymbol{\beta} &= (\mathrm{Y} - \Phi(x_0) - \Theta(\mathrm{U}) - \Theta(\mathrm{D}))_{(i,\cdot)} \end{aligned}$$

Application of Lemma 4.1 on (4.12) gives

$$\begin{aligned} &\max_{\delta\Psi_i \in \mathcal{U}_{i,(\ell_q,\ell_r)}} \left\| \left(\mathrm{Y} - \widetilde{\Phi}(x_0) - \widetilde{\Theta}(\mathrm{U}) - \widetilde{\Theta}(\mathrm{D}) - \widetilde{\Upsilon}(\mathrm{W}, \mathrm{V}) \right)_{(i,\cdot)} \right\|_{\ell_r} \\ &\qquad = \left\| (\mathrm{Y} - \Phi(x_0) - \Theta(\mathrm{U}) - \Theta(\mathrm{D}))_{(i,\cdot)} \right\|_{\ell_r} + \rho_i (\|x_0\|_{\ell_1} + \|\mathbf{u}\|_{\ell_1} + \|\mathbf{d}\|_{\ell_1} + 1) \end{aligned}$$

Thus (4.11) becomes

$$\begin{aligned} (\hat{x}_0, \hat{\mathrm{D}}) &= \underset{\substack{x_0 \in \mathbb{R}^n \\ \mathrm{D} \in \mathbb{R}^{m \times T}}}{\arg\min} \sum_{i=1}^{p} \left\| (\mathrm{Y} - \Phi(x_0) - \Theta(\mathrm{U}) - \Theta(\mathrm{D}))_{(i,\cdot)} \right\|_{\ell_r} + \sum_{i=1}^{p} \rho_i (\|x_0\|_{\ell_1} + \|\mathbf{u}\|_{\ell_1} + \|\mathrm{D}\|_{\ell_1/\ell_1} + 1) \\ &\qquad + \lambda \|\mathrm{D}\|_{\ell_1/\ell_r} \\ &= \underset{\substack{x_0 \in \mathbb{R}^n \\ \mathrm{D} \in \mathbb{R}^{m \times T}}}{\arg\min}\ \|\mathrm{Y} - \Phi(x_0) - \Theta(\mathrm{U}) - \Theta(\mathrm{D})\|_{\ell_1/\ell_r} + \sum_{i=1}^{p} \rho_i \|x_0\|_{\ell_1} + \sum_{i=1}^{p} \rho_i \|\mathrm{D}\|_{\ell_1/\ell_1} + \lambda \|\mathrm{D}\|_{\ell_1/\ell_r} \end{aligned}$$

since we have assumed that $\mathbf{u}$ is independent of x_0.

□

Coupled Uncertainty Set

We now consider the case $\ell_q = \ell_r = \ell_1$. We will show that an alternative robust counterpart can be found that accommodates a coupled uncertainty set. This is in contrast to the uncoupled row-wise uncertainty sets in Proposition 4.1 and Corollary 4.1.

Definition 4.2 (Coupled uncertainty set). *Let* $\delta\Psi := \begin{bmatrix} \delta\mathcal{O} & \delta\mathcal{J}_u & \delta\mathcal{J}_u & \left(\widetilde{\mathcal{J}}_w \mathbf{w} + \mathbf{v}\right) \end{bmatrix}$ *be an uncertain matrix belonging to the uncertainty set* $\mathcal{U}_{(\ell_1,\ell_1)} = \{\delta\Psi : \|\delta\Psi\|_{(\ell_1,\ell_1)} \leq \rho\}$.

Proposition 4.2 (Robust Estimation of x_0 with ℓ_1/ℓ_1 relaxation and ℓ_1-regularization). *Let the uncertain matrix* $\delta\Psi$ *and its corresponding uncertainty set* $\mathcal{U}_{(\ell_1,\ell_1)}$ *be defined according to Definition 4.2. Then,*

$$(\hat{x}_0, \hat{\mathrm{D}}) = \underset{\substack{x_0 \in \mathbb{R}^n \\ \mathrm{D} \in \mathbb{R}^{m \times T}}}{\arg\min} \|\mathrm{Y} - \Phi(x_0) - \Theta(\mathrm{U}) - \Theta(\mathrm{D})\|_{\ell_1/\ell_1} + \rho\|x_0\|_{\ell_1} + (\rho + \lambda)\|\mathrm{D}\|_{\ell_1/\ell_1}$$

with $\lambda > 0$ *as a tuning parameter.*

Proof. From the definition of the mixed ℓ_1/ℓ_1-norm, we have

$$\begin{aligned} &\left\|\mathrm{Y} - \widetilde{\Phi}(x_0) - \widetilde{\Theta}(\mathrm{D}) - \widetilde{\Theta}(\mathrm{U}) - \widetilde{\Upsilon}(\mathrm{W}, \mathrm{V})\right\|_{\ell_1/\ell_1} \\ &= \left\|\mathbf{y} - \widetilde{\mathcal{O}} x_0 - \widetilde{\mathcal{J}}_u(\mathbf{u} + \mathbf{d}) - \widetilde{\mathcal{J}}_w \mathbf{w} - \mathbf{v}\right\|_{\ell_1} \\ &= \|\mathbf{y} - \mathcal{J}_u \mathbf{u} - \mathcal{J}_u \mathbf{d} - (\Psi + \delta\Psi)\,\boldsymbol{\kappa}\|_{\ell_1} \end{aligned} \tag{4.16}$$

where $\Psi := \begin{bmatrix} \mathcal{O} & 0 & 0 & 0 \end{bmatrix}$ and $\boldsymbol{\kappa} := \begin{bmatrix} x_0^\top & \mathbf{u}^\top & \mathbf{d}^\top & 1 \end{bmatrix}^\top$. Noting that

$$\begin{aligned} &\Psi\,\boldsymbol{\kappa} = \mathcal{O} x_0, \\ &\|\boldsymbol{\kappa}\|_{\ell_1} = \|x_0\|_{\ell_1} + \|\mathbf{u}\|_{\ell_1} + \|\mathbf{d}\|_{\ell_1} + 1, \quad \text{and} \\ &\|\mathbf{d}\|_{\ell_1} = \|\mathrm{D}\|_{\ell_1/\ell_1} \end{aligned}$$

From (4.10) and the application of Theorem 4.1,

$$
\begin{aligned}
(\hat{x}_0, \hat{\mathrm{D}}) &= \underset{\substack{x_0 \in \mathbb{R}^n \\ \mathrm{D} \in \mathbb{R}^{m \times T}}}{\arg\min} \max_{\delta\Psi \in \mathcal{U}_{(\ell_1,\ell_1)}} \left\| \mathrm{Y} - \widetilde{\Phi}(x_0) - \widetilde{\Theta}(\mathrm{U}) - \widetilde{\Theta}(\mathrm{D}) - \widetilde{\Upsilon}(\mathrm{W}, \mathrm{V}) \right\|_{\ell_1/\ell_1} + \lambda \|\mathrm{D}\|_{\ell_1/\ell_1} \\
&= \underset{\substack{x_0 \in \mathbb{R}^n \\ \mathrm{D} \in \mathbb{R}^{m \times T}}}{\arg\min} \max_{\delta\Psi \in \mathcal{U}_{(\ell_1,\ell_1)}} \|(\mathbf{y} - \mathcal{J}_u \mathbf{u} - \mathcal{J}_u \mathbf{d}) - (\Psi + \delta\Psi)\,\boldsymbol{\kappa}\|_{\ell_1} + \lambda \|\mathrm{D}\|_{\ell_1/\ell_1} \\
&= \underset{\substack{x_0 \in \mathbb{R}^n \\ \mathbf{d} \in \mathbb{R}^{mT}}}{\arg\min} \|\mathbf{y} - \mathcal{J}_u \mathbf{u} - \mathcal{J}_u \mathbf{d} - \Psi\,\boldsymbol{\kappa}\|_{\ell_1} + \rho \|\boldsymbol{\kappa}\|_{\ell_1} + \lambda \|\mathrm{D}\|_{\ell_1/\ell_1} \\
&= \underset{\substack{x_0 \in \mathbb{R}^n \\ \mathbf{d} \in \mathbb{R}^{mT}}}{\arg\min} \|\mathbf{y} - \mathcal{J}_u \mathbf{u} - \mathcal{J}_u \mathbf{d} - \mathcal{O} x_0\|_{\ell_1} + \rho \left(\|x_0\|_{\ell_1} + \|\mathbf{u}\|_{\ell_1} + \|\mathrm{D}\|_{\ell_1/\ell_1} + 1 \right) + \lambda \|\mathrm{D}\|_{\ell_1/\ell_1} \\
&= \underset{\substack{x_0 \in \mathbb{R}^n \\ \mathbf{d} \in \mathbb{R}^{mT}}}{\arg\min} \|\mathbf{y} - \mathcal{J}_u \mathbf{u} - \mathcal{J}_u \mathbf{d} - \mathcal{O} x_0\|_{\ell_1} + \rho \|x_0\|_{\ell_1} + (\rho + \lambda) \|\mathrm{D}\|_{\ell_1/\ell_1}
\end{aligned}
$$

since we have assumed that $\mathbf{u}$ is independent of x_0. The result follows from the relation

$$
\|\mathrm{Y} - \Phi(x_0) - \Theta(\mathrm{U}) - \Theta(\mathrm{D})\|_{\ell_1/\ell_1} = \|\mathbf{y} - \mathcal{J}_u \mathbf{u} - \mathcal{J}_u \mathbf{d} - \mathcal{O} x_0\|_{\ell_1}
$$

□

Summary

We have now gained a key insight that, with an appropriate choice of an uncertainty set from Definitions 4.1 or 4.2, a robustification of (4.7) is equivalent to a regularization procedure. In addition, we restrict ourselves to the cases where $\ell_q = \ell_1$.

Summarizing the results of Corollary 4.1 and Proposition 4.2, our robust estimator is given by

$$
(\hat{x}_{0,\mathrm{rob}}^{\ell_1/\ell_r}, \hat{\mathrm{D}}_{\mathrm{rob}}) = \underset{\substack{x_0 \in \mathbb{R}^n \\ \mathrm{D} \in \mathbb{R}^{m \times T}}}{\arg\min} \|\mathrm{Y} - \Phi(x_0) - \Theta(\mathrm{U}) - \Theta(\mathrm{D})\|_{\ell_1/\ell_r} + \rho \|x_0\|_{\ell_1} + \rho \|\mathrm{D}\|_{\ell_1/\ell_1} + \lambda \|\mathrm{D}\|_{\ell_1/\ell_r} \tag{4.17}
$$

where ρ is a parameter that controls the amount of robustification (a greater ρ indicates a more conservative estimator) and λ is a tuning parameter.

In contrast, the nominal estimator in (4.7) does not consider modeling errors and noise signals. It is given by

$$(\hat{x}_{0,\text{nom}}^{\ell_1/\ell_r}, \hat{\mathrm{D}}_{\text{nom}}) = \underset{\substack{x_0 \in \mathbb{R}^n \\ \mathrm{D} \in \mathbb{R}^{m \times T}}}{\arg\min} \|\mathrm{Y} - \Phi(x_0) - \Theta(\mathrm{U}) - \Theta(\mathrm{D})\|_{\ell_1/\ell_r} + \lambda \|\mathrm{D}\|_{\ell_1/\ell_r} \tag{4.18}$$

Remark 4.1. *In practice, it is difficult to construct ρ because the modeling errors and noise signals cannot be accurately predicted. In addition, there is no clear strategy for selecting the ideal values for λ and ℓ_r. Therefore it is natural to use a statistical approach such as cross-validation with data sets to obtain ρ, λ and ℓ_r, which will be discussed in detail in Section 4.4.1.*

4.3.2 Robust Estimation of the Remaining States

In the previous section, we have developed a robust and resilient estimator for obtaining $(\hat{x}_0, \hat{\mathrm{D}})$. However, we cannot obtain $\hat{x}_1, \cdots, \hat{x}_{T-1}$ using (4.1) because we do not know the parameters $\widetilde{A}, \widetilde{B}$ and noise signals w_k. In this section, we will develop a robust estimator for the states $x_1, \cdots, x_{T-1}$ using $(\hat{x}_0, \hat{\mathrm{D}})$.

The problem can be formulated as: given $(\hat{x}_0, \hat{\mathrm{D}})$, we wish to obtain estimates of the states $X := \begin{bmatrix} x_1^\top & x_2^\top & \cdots & x_{T-1}^\top \end{bmatrix}^\top$ that are robust to modeling errors δA, δB and noise signals w_k. First, note that (4.1) can be compactly written as

$$X = \widetilde{\mathcal{P}} x_0 + \widetilde{\mathcal{K}}_u(\mathbf{u} + \mathbf{d}) + \widetilde{\mathcal{K}}_w \mathbf{w} \tag{4.19}$$

where the state transition and input matrices are given by

$$\widetilde{\mathcal{P}} = \begin{bmatrix} (\widetilde{A})^\top & (\widetilde{A}^2)^\top & \cdots & (\widetilde{A}^{T-1})^\top \end{bmatrix}^\top,$$

$$\widetilde{\mathcal{K}}_u = \begin{bmatrix} \widetilde{B} & 0 & 0 & \cdots & 0 & 0 \\ \widetilde{A}\widetilde{B} & \widetilde{B} & 0 & \cdots & 0 & 0 \\ \widetilde{A}^2\widetilde{B} & \widetilde{A}\widetilde{B} & \widetilde{B} & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ \widetilde{A}^{T-2}\widetilde{B} & \widetilde{A}^{T-3}\widetilde{B} & \widetilde{A}^{T-4}\widetilde{B} & \cdots & \widetilde{B} & 0 \end{bmatrix}, \quad \widetilde{\mathcal{K}}_w = \begin{bmatrix} I & 0 & 0 & \cdots & 0 & 0 \\ \widetilde{A} & I & 0 & \cdots & 0 & 0 \\ \widetilde{A}^2 & \widetilde{A} & I & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ \widetilde{A}^{T-2} & \widetilde{A}^{T-3} & \widetilde{A}^{T-4} & \cdots & I & 0 \end{bmatrix} \tag{4.20}$$

The matrices $\mathcal{P}$, $\mathcal{K}_u$ and $\mathcal{K}_w$ are similarly defined with A and B instead of $\widetilde{A}$ and $\widetilde{B}$. In addition, we define $\delta\mathcal{P} := \widetilde{\mathcal{P}} - \mathcal{P}$ and $\delta\mathcal{K}_u := \widetilde{\mathcal{K}}_u - \mathcal{K}_u$.

Definition 4.3. *Let* $\delta\Omega := \begin{bmatrix} \delta\mathcal{P} & \delta\mathcal{K}_u & \widetilde{\mathcal{K}}_w\mathbf{w} \end{bmatrix}$ *be an uncertain matrix belonging to the uncertainty set* $\mathcal{U}_{(\ell_q,\ell_r)} = \{\delta\Omega : \|\delta\Omega\|_{(\ell_q,\ell_r)} \leq \tilde{\rho}\}$.

Proposition 4.3 (Robust Estimation of State Sequence). *Let* $\delta\Omega$ *and* $\mathcal{U}_{(\ell_q,\ell_r)}$ *be defined according to Definition 4.3. Then, given* $x_0 = \hat{x}_0$ *and* $\hat{\mathbf{d}} = \mathrm{vec}\left(\hat{\mathrm{D}}\right)$ *and for some* $q, r \in [1, \infty]$, *the robust estimate of* X *is given by*

$$\begin{aligned}\hat{X} &= \underset{X\in\mathbf{R}^{n(T-1)}}{\arg\min} \max_{\delta\Omega\in\mathcal{U}_{(\ell_q,\ell_r)}} \|X - \widetilde{\mathcal{P}}\hat{x}_0 - \widetilde{\mathcal{K}}_u(\mathbf{u}+\hat{\mathbf{d}}) - \widetilde{\mathcal{K}}_w\mathbf{w}\|_{\ell_r} \\ &= \mathcal{P}\hat{x}_0 + \mathcal{K}_u(\mathbf{u}+\hat{\mathbf{d}}).\end{aligned}$$

Proof. From (4.19) and the definitions in (4.20),

$$X - \widetilde{\mathcal{P}}\hat{x}_0 - \widetilde{\mathcal{K}}_u(\mathbf{u}-\hat{\mathbf{d}}) - \widetilde{\mathcal{K}}_w\mathbf{w} = X - (\Omega + \delta\Omega)\boldsymbol{\gamma},$$

where $\Omega := \begin{bmatrix} \mathcal{P} & \mathcal{K}_u & 0 \end{bmatrix}$, $\delta\Omega := \begin{bmatrix} \delta\mathcal{P} & \widetilde{\mathcal{K}}_u & \widetilde{\mathcal{K}}_w\mathbf{w} \end{bmatrix}$ and $\boldsymbol{\gamma} = \begin{bmatrix} \hat{x}_0^\top & \mathbf{u}^\top + \hat{\mathbf{d}}^\top & 1 \end{bmatrix}^\top$. Then, by Theorem 4.1, we have

$$\begin{aligned}\hat{X} &= \underset{X\in\mathbf{R}^{n(T-1)}}{\arg\min} \max_{\delta\Omega\in\mathcal{U}_{(\ell_q,\ell_r)}} \|X - (\Omega+\delta\Omega)\boldsymbol{\gamma}\|_{\ell_r} \\ &= \underset{X\in\mathbf{R}^{n(T-1)}}{\arg\min} \|X - \Omega\boldsymbol{\gamma}\|_{\ell_r} + \tilde{\rho}\|\boldsymbol{\gamma}\|_{\ell_q} \\ &= \underset{X\in\mathbf{R}^{n(T-1)}}{\arg\min} \|X - \mathcal{P}\hat{x}_0 - \mathcal{K}_u(\mathbf{u}+\hat{\mathbf{d}})\|_{\ell_r} + \tilde{\rho}\|\boldsymbol{\gamma}\|_{\ell_q} \\ &= \mathcal{P}\hat{x}_0 + \mathcal{K}_u(\mathbf{u}+\hat{\mathbf{d}})\end{aligned}$$

since $\boldsymbol{\gamma}$ is known and $\|z\|_{\ell_r} = 0$ if and only if $z = 0$. □

4.4 Numerical Simulations

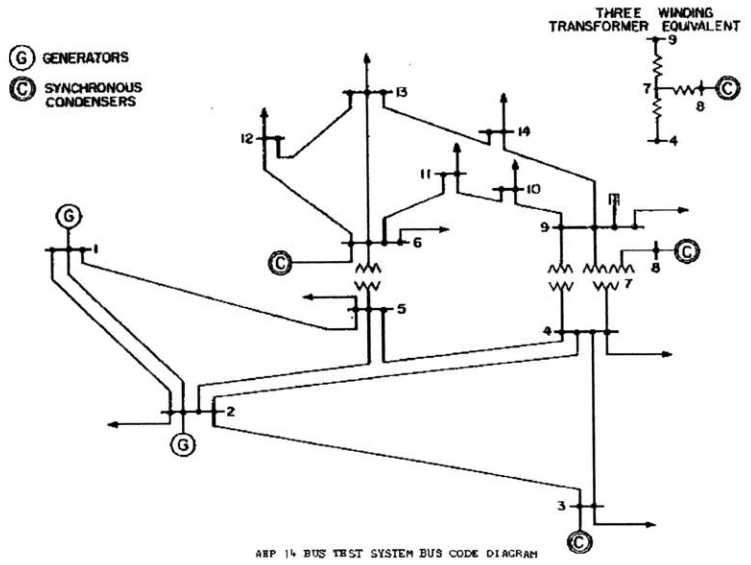


Figure 4-1: IEEE 14-bus system [17]

In this section, we demonstrate the effectiveness of our robust and resilient estimator (4.17) using an IEEE 14-bus system [17]. The system, depicted in Figure 4-1, comprises of 5 synchronous generators and 14 buses. It is represented by 10 states comprising of the rotor angles and frequencies of each generator. The dynamics of the system can be represented by the following uncertain and noisy continuous time LTI model:

$$\begin{aligned} \dot{x}(t) &= \widetilde{A}_c\, x(t) + \widetilde{B}_c\, (u(t) + d(t)) + w(t) \\ y(t) &= \widetilde{C}_c\, x(t) + \widetilde{D}_c\, (u(t) + d(t)) + e(t) + v(t) \end{aligned} \tag{4.21}$$

where the matrices $\widetilde{A}_c$, $\widetilde{B}_c$ and $\widetilde{C}_c$ are defined in Appendix B and $\widetilde{D}_c = 0$. The other variables denote the same quantities as in (4.1).

To obtain a discrete time model, we discretize (4.21) with a sampling interval of $\Delta T = 0.05s$. Similar to [43], $p = 35$ sensors is deployed to measure the real power injections at every bus, the real power flows along every branch and the rotor angle of generator 1 and the sensor measuring the rotor angle of generator 1 is also protected

from attacks.

In the following sections, we will describe the cross validation procedure used to determine the hyperparameters of our robust and resilient estimator, as well as the simulations used to compare the performance of our robust and resilient estimator (4.17) with the nominal estimator (4.18). The estimators are implemented in MATLAB and a MATLAB interface to CVX [27, 28] is used to solve the optimization problems. In all our simulations, the initial state $x(0) = x_0$ and uncertainties (modeling errors and noise signals) are drawn from independent and identically distributed Gaussian distributions.

4.4.1 Cross Validation Procedure for Selection of Hyperparameters

In practice, it is difficult to predict the modeling errors and noise signals for the purpose of constructing our uncertainty sets. Thus, it is natural to use a statistical learning procedure known as *cross-validation* to determine the hyperparameters of our robust and resilient estimator – namely, given some training data, we want to select i) the tuning parameter λ, ii) the robustification level ρ, and iii) the estimation approach among our robust and resilient ℓ_1/ℓ_1, ℓ_1/ℓ_2 and ℓ_1/ℓ_∞ estimators.

To this end, 200 sets of data (given by the tuple $(x_0, y_0, y_1, \ldots, y_{T-1})$ are generated with a window size of $T = 15$) using a nominal system model with modeling errors and attack signals drawn from i.i.d. Gaussian distributions, initial states x_0 drawn from the standard Gaussian distribution, different sets of attacked sensors K of cardinality $q_s = 3$ and different sets of attacked actuators L of cardinality $q_a = 1$. Subsequently, the data is randomly partitioned into three sets: allocate 50% for *training*, 25% for *validation* and 25% for *testing*. The procedure of cross-validation for both the nominal and robust resilient estimators is conducted in the following phases:

Training: For each approach (ℓ_1/ℓ_1, ℓ_1/ℓ_2 and ℓ_1/ℓ_∞), find the best values of λ and ρ using the *training* set.

Validation: Using the *validation* set, select the best approach among the ℓ_1/ℓ_1,

ℓ_1/ℓ_2 and ℓ_1/ℓ_∞ estimators with λ and ρ that were determined in the training phase.

Testing: Determine how well the resilient estimator (nominal and robust) can predict the values of x_0 in the *testing* set.

When the above process is repeated 20 times, average reductions of 16.92% and 11.68% in the mean and standard deviation, respectively, of the state estimation errors are observed. Furthermore, when the intensities of the model errors and noise signals are increased by about 2.5 times, a similar cross-validation study shows decreases of 14.06% and 41.43% in the mean and standard deviation, respectively, of the state estimation errors.

4.4.2 Varying intensities of modeling uncertainty and noise signals

To observe the effects of uncertainties (modeling errors and noise signals) on the performances of our estimators, their intensities are varied while parameters are kept constant. For different intensity levels, the simulations are repeated 100 times with different sets of attacked sensors K of cardinality $q_s = 3$, different sets of attacked actuators L of cardinality $q_a = 1$.

The procedure is repeated for different robust and resilient estimators (ℓ_1/ℓ_1, ℓ_1/ℓ_2 and ℓ_1/ℓ_∞) and compared with the nominal estimator. From Figure 4-2, it is clear that the nominal estimator has the best performance when the uncertainty intensity is small. As the uncertainty intensity is increased, the situation is reversed, and in addition a larger ρ (i.e., more conservative robust and resilient estimator) leads to improved estimates.

4.4.3 Varying number of sensor and actuator attacks

In this task, the performances of the robust and resilient estimator and the nominal estimator are compared for different number of attacked sensors and actuators. The

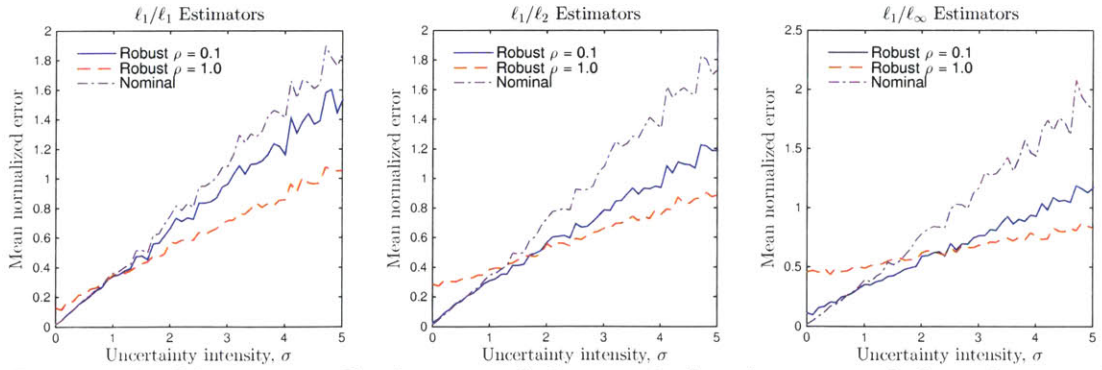
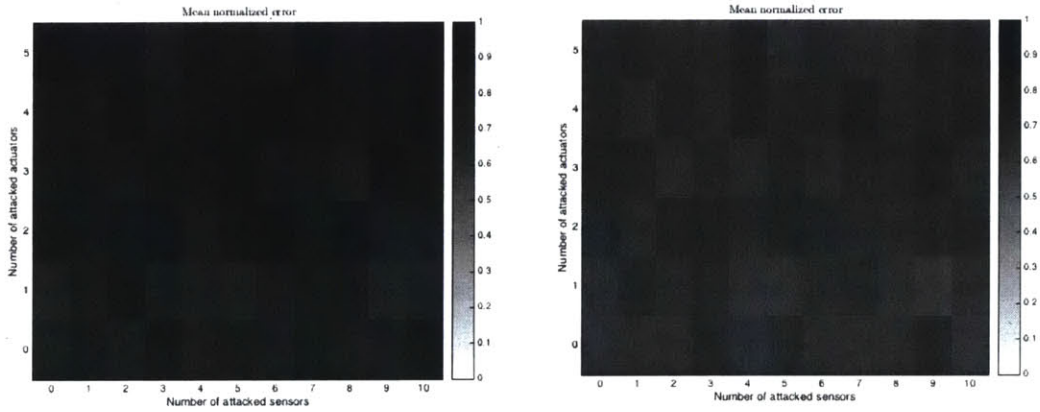


Figure 4-2: Mean normalized errors of the nominal estimator and the robust and resilient ℓ_1/ℓ_1, ℓ_1/ℓ_2 and ℓ_1/ℓ_∞ estimators (with $\lambda = 0.2$ and different values for ρ) simulated on the IEEE 14-bus system.

results of 100 simulations, summarized in 4-3, indicate that the robust and resilient estimator performs consistently better (with a mean normalized error between 0.6072 and 0.7009) than the nominal estimator (with a mean normalized error between 0.7326 and 0.9360).



(a) Nominal ℓ_1/ℓ_∞ estimator

(b) Robust and resilient ℓ_1/ℓ_∞ estimator

Figure 4-3: Mean normalized errors of the nominal and robust ℓ_1/ℓ_∞ estimators (with $\lambda = 0.2$ and $\rho = 0.1$) simulated on the IEEE 14-bus system. A darker shade indicates a higher relative mean normalized error.

4.4.4 Estimation of the Initial State for Different Attack and Uncertainty Scenarios

Next, we compared the performances of the robust and resilient estimator and the nominal estimator for various scenarios. The results of 500 simulations are summarized in Figure 4-4.

In the first scenario, where the modeling errors and noise signals are absent ("*attack only*"), it can be observed that the nominal estimator performs the best, thus validating the results of 4.4.2. In the second scenario, where the attack signals are absent ("*uncertainty only*"), a significant improvement in performance of the robust and resilient estimator over the robust estimator can be observed. The same observations can be drawn from the third scenario ("*uncertainty and attack*"), where modeling errors, noise signals and attacks are considered.

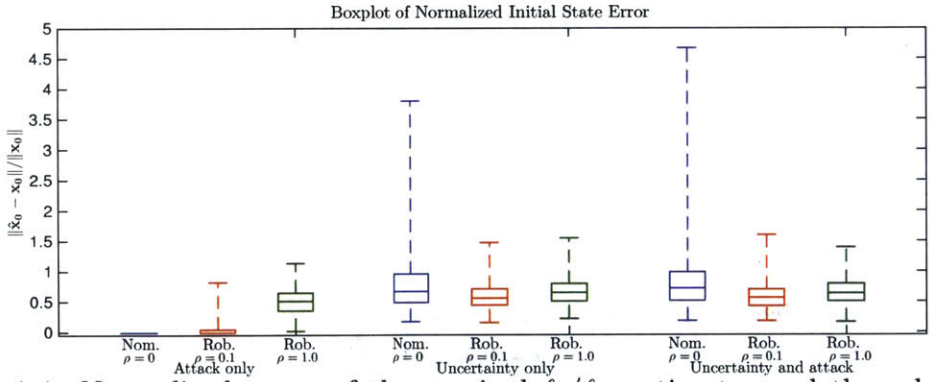


Figure 4-4: Normalized errors of the nominal ℓ_1/ℓ_∞ estimator and the robust and resilient ℓ_1/ℓ_∞ estimator (with $\lambda = 0.2$ and different values for ρ) under different scenarios simulated on the IEEE 14-bus system. The dashed lines represent the support of the data, while the box represents the mean and standard deviation of the normalized errors.

4.4.5 Estimation of the State Trajectory for Different Attack and Uncertainty Scenarios

Lastly, the robust estimation of the state trajectory, developed in Section 4.3.2, is validated using the same scenarios that are considered in Section 4.4.4. The results of 100 simulations are summarized in Figure 4-5. As expected, the nominal estimator

performs best in the "*attack only*" scenario, but fares worse than the robust and resilient estimators in the other scenarios.

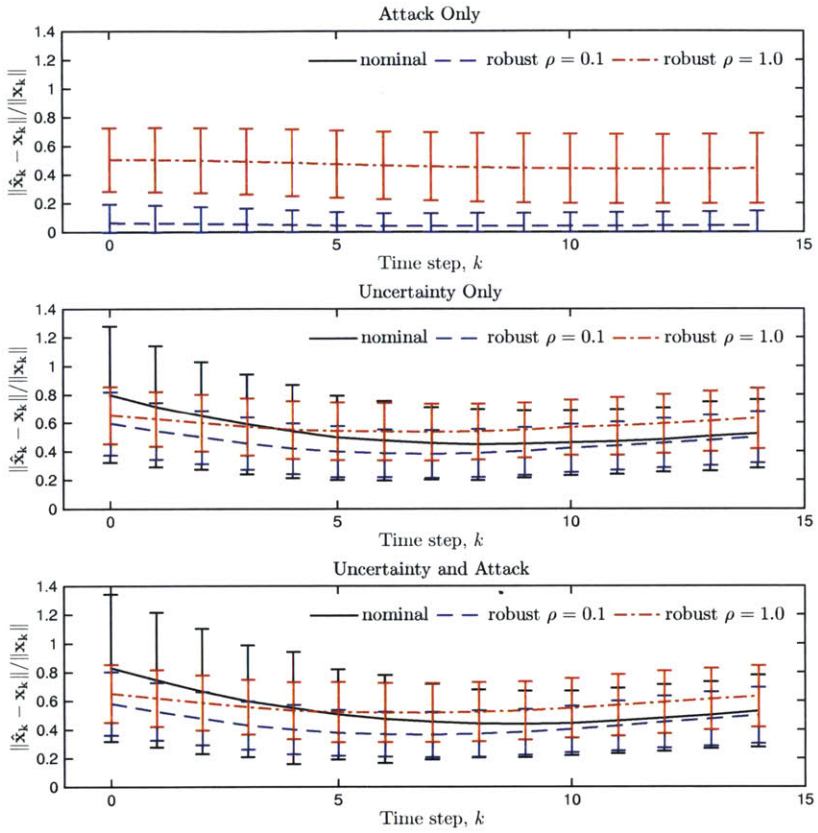


Figure 4-5: Normalized errors of the nominal ℓ_1/ℓ_∞ estimator and the robust and resilient ℓ_1/ℓ_∞ estimator (with $\lambda = 0.2$ and different values for ρ) under different scenarios for the IEEE 14-bus system. The curves represent mean values and the error bars represent standard deviations.

Chapter 5

Conclusion

Distributed Feasibility Algorithms for Networks

In the first part of this thesis, we studied the distributed operation of the electric power grid using the power flow problem. Two novel distributed algorithms are developed for finding feasible assignments of values in a network when all constraints are convex. Our algorithms distribute computation among the nodes of the network and require only local information exchanges. Although the power flow problem is non-convex, our algorithms are demonstrated to be effective heuristics using meaningful power flow scenarios and are shown to perform well in comparison to existing algorithms.

A number of issues, mainly related to convergence analysis, deserve future attention. Also, the choice of weights in the CC and DCS algorithms and their impact on the algorithms' rates of convergence remain open problems.

Secure Estimation for Cyber-Physical Systems

In the second part of this thesis, we studied the cyber-physical security of the electric power grid by considering the problem of state estimation of a noisy and uncertain cyber-physical system that is subjected to data injection attacks. A novel state estimation algorithm that is resilient to adversarial actions and robust to modeling errors and additive noise signals is developed. By leveraging principles of robust optimiza-

tion, the estimator is formulated as a convex optimization problem. The use of cross validation is also advocated for determining the hyperparameters of our estimator and its effectiveness is demonstrated using simulations of an IEEE 14-bus system.

For future research, it will be interesting to adapt the robust and resilient estimator developed in this thesis in a feedback loop for control and compared with a H_∞ controller. The robust and resilient estimator can also be improved by developing uncertainty sets that are tailored for specific applications, by taking into account structural vulnerabilities of the particular cyber-physical system.

Appendix A

Convergence Analysis of the CC algorithm

In this section, we analyze the convergence of the CC algorithm for the case where $\mathcal{S}_1, \cdots, \mathcal{S}_m$ are closed and convex and $\mathcal{S}$ is non-empty. We will first introduce several notations for representational convenience, then derive two lemmas, both of which will then be used to establish the convergence properties of the CC algorithm.

For the m-tuple $x = (x_1\ ,\ \cdots\ ,\ x_m)$ and subset of indices $I \subseteq V$, we let $[x]_I$ denote that the subset of decision variables $\{x_i : i \in I\}$. Similarly for the set $\mathcal{S} \subseteq \mathbb{E}^m$, we denote $[\mathcal{S}]_I$ denote that projection of $\mathcal{S}$ onto the space of the coordinates $i \in I$.

Observe that the update rules (3.6) and (3.7) can be rewritten as

$$\widehat{r}^i(t) = \left\{ \sum_{k \in I_i} a^i_{j,k}(t)\ x^k_j(t)\ \ :\ \ j \in I_i \right\} \tag{A.1}$$

$$r^i(t+1) = \widehat{r}^i(t) + e^i(t) \tag{A.2}$$

where $e^i(t)$ represents the error due to projection given by

$$e^i(t) = P_{\mathcal{S}_i}\left(\widehat{r}^i(t)\right) - \widehat{r}^i(t) \tag{A.3}$$

The evolution dynamics of the estimates $r^i(t)$ for each agent is decomposed into

a sum of a linear term $\widehat{r}^i(t)$ and a non-linear term $e^i(t)$. The linear term captures the effects of mixing the agent estimates, while the non-linear term captures the effects of the projection operation.

Lemma A.1. *Let $\mathcal{S}$ be a non-empty closed convex set in $\mathbb{E}$. Then for any $x \in \mathbb{E}$, we have*

$$\|P_{\mathcal{S}}(x) - s\|^2 \leq \|x - s\|^2 - \|P_{\mathcal{S}}(x) - x\|^2 \quad \forall \quad s \in \mathcal{S}$$

Lemma A.2. *Assume that $\mathcal{S}_1, \cdots, \mathcal{S}_m$ are closed and convex, and $\mathcal{S}$ is non-empty. In addition, let Assumptions 3.1 and 3.2 hold.*

(a) For all $x^* \in \mathcal{S}$ and all t, we have

(i) $\left\|r^i(t+1) - [x^*]_{I_i}\right\|^2 \leq \left\|\widehat{r}^i(t) - [x^*]_{I_i}\right\|^2 - \left\|e^i(t)\right\|^2$
for all i

(ii) $\displaystyle\sum_{i=1}^{m} \left\|\widehat{r}^i(t) - [x^*]_{I_i}\right\|^2 \leq \sum_{i=1}^{m} \left\|r^i(t) - [x^*]_{I_i}\right\|^2$

(b) For all $x^* \in \mathcal{S}$, the sequences $\left\{\displaystyle\sum_{i=1}^{m} \left\|\widehat{r}^i(t) - [x^*]_{I_i}\right\|^2\right\}$ *and* $\left\{\displaystyle\sum_{i=1}^{m} \left\|r^i(t) - [x^*]_{I_i}\right\|^2\right\}$ *are monotonically non-increasing with t.*

(c) The errors $e^i(t)$ converge to zero as $t \to \infty$.

Proof. (a) (i)

For any $x^* \in \mathcal{S}$ and i, we have $[x^*]_{I_i} \in [\mathcal{S}]_{I_i}$ and $[\mathcal{S}]_{I_i} \subseteq \mathcal{S}_i$. It follows that $[x^*]_{I_i} \in \mathcal{S}_i$ for all i. The application of Lemma A.1 yields

$$\left\|P_{\mathcal{S}_i}\left(\widehat{r}^i(t)\right) - [x^*]_{I_i}\right\|^2 \leq \left\|\widehat{r}^i(t) - [x^*]_{I_i}\right\|^2 - \left\|P_{\mathcal{S}_i}\left(\widehat{r}^i(t)\right) - \widehat{r}^i(t)\right\|^2$$

Substituting for the update relation (3.6) and error expression (A.3), we obtain (a)(i).

(a) (ii)

By Assumption 3.2(a) we have $\sum_{k=1}^{m} a_{j,k}^i = \sum_{k \in I_i} a_{j,k}^i = 1$, thus

$$\widehat{r}^i(t) - [x^*]_{I_i} = \left\{ \sum_{k \in I_i} a_{j,k}^i(t) \left((x_j^k(t) - x_j^*) \right) \ : \ j \in I_i \right\}$$

Taking the squared norm of $\widehat{r}^i(t) - [x^*]_{I_i}$,

$$\begin{aligned} \left\| \widehat{r}^i(t) - [x^*]_{I_i} \right\|^2 &= \sum_{j \in I_i} \left\| \sum_{k \in I_i} a_{j,k}^i(t) \left(x_j^k(t) - x_j^* \right) \right\|^2 \\ &\leq \sum_{j \in I_i} \sum_{k \in I_i} a_{j,k}^i(t) \left\| x_j^k(t) - x_j^* \right\|^2 \end{aligned}$$

where the second inequality holds because the inner sum $\sum_{k \in I_i} a_{j,k}^i(t) \left(x_j^k(t) - x_j^* \right)$ is a convex combination of $x_j^k(t) - x_j^*$ and the norm operator $\| \cdot \|$ is a convex function. By summing the preceding relations over i,

$$\begin{aligned} \sum_{i=1}^{m} \left\| \widehat{r}^i(t) - [x^*]_{I_i} \right\|^2 &\leq \sum_{j \in I_i} \sum_{k \in I_i} \left(\sum_{i=1}^{m} a_{j,k}^i(t) \right) \left\| x_j^k(t) - x_j^* \right\|^2 \\ &= \sum_{j \in I_i} \sum_{k \in I_i} \left\| x_j^k(t) - x_j^* \right\|^2 \text{ by Assumption 3.2(a)} \\ &= \sum_{k \in I_i} \left\| r^k(t) - [x^*]_{I_k} \right\|^2 \end{aligned}$$

(b)

Combining parts (a)(i) and (a)(ii),

$$\begin{aligned} \sum_{i=1}^{m} \left\| \widehat{r}^i(t+1) - [x^*]_{I_i} \right\|^2 &\leq \sum_{i=1}^{m} \left\| r^i(t+1) - [x^*]_{I_i} \right\|^2 \\ &\leq \sum_{i=1}^{m} \left\| \widehat{r}^i(t) - [x^*]_{I_i} \right\|^2 - \sum_{i=1}^{m} \left\| e^i(t) \right\|^2 \\ &\leq \sum_{i=1}^{m} \left\| \widehat{r}^i(t) - [x^*]_{I_i} \right\|^2 \end{aligned}$$

which proves that the sequence $\left\{\sum_{i=1}^{m} \left\|\widehat{r}^i(t) - [x^*]_{I_i}\right\|^2\right\}$ is monotonically non-increasing with t. Similarly,

$$\begin{aligned}\sum_{i=1}^{m} \left\|r^i(t+1) - [x^*]_{I_i}\right\|^2 &\leq \sum_{i=1}^{m} \left\|\widehat{r}^i(t) - [x^*]_{I_i}\right\|^2 - \sum_{i=1}^{m} \left\|e^i(t)\right\|^2 \\ &\leq \sum_{i=1}^{m} \left\|\widehat{r}^i(t) - [x^*]_{I_i}\right\|^2 \\ &\leq \sum_{i=1}^{m} \left\|r^i(t) - [x^*]_{I_i}\right\|^2\end{aligned}$$

which proves that the sequence $\left\{\sum_{i=1}^{m} \left\|r^i(t) - [x^*]_{I_i}\right\|^2\right\}$ is monotonically non-increasing with t.

(c)

Summing (a)(i) over i,

$$\sum_{i=1}^{m} \left\|e^i(t)\right\|^2 \leq \sum_{i=1}^{m} \left\|\widehat{r}^i(t) - [x^*]_{I_i}\right\|^2 - \sum_{i=1}^{m} \left\|r^i(t+1) - [x^*]_{I_i}\right\|^2$$

Combining the above relation with (a) (ii),

$$\begin{aligned}\sum_{i=1}^{m} \left\|e^i(t)\right\|^2 &\leq \sum_{i=1}^{m} \left\|r^i(t) - [x^*]_{I_i}\right\|^2 - \sum_{i=1}^{m} \left\|r^i(t+1) - [x^*]_{I_i}\right\|^2 \\ &\leq \sum_{i=1}^{m} \left\|r^i(t) - [x^*]_{I_i}\right\|^2 - \sum_{i=1}^{m} \left\|r^i(t+1) - [x^*]_{I_i}\right\|^2\end{aligned}$$

Summing the above relation over $t = 0, \cdots, s$, for any $s > 0$ and expressing the result as a telescoping series yields

$$\begin{aligned}\sum_{t=0}^{s} \sum_{i=1}^{m} \left\|e^i(t)\right\|^2 &\leq \sum_{i=1}^{m} \left\|r^i(0) - [x^*]_{I_i}\right\|^2 - \sum_{i=1}^{m} \left\|r^i(s+1) - [x^*]_{I_i}\right\|^2 \\ &\leq \sum_{i=1}^{m} \left\|r^i(0) - [x^*]_{I_i}\right\|^2\end{aligned}$$

By letting $s \to \infty$,

$$\sum_{t=0}^{\infty} \sum_{i=1}^{m} \left\| e^i(t) \right\|^2 \leq \sum_{i=1}^{m} \left\| r^i(0) - [x^*]_{I_i} \right\|^2$$

which implies $\lim_{t\to\infty} \|e^i(t)\| = 0$ for all i.

□

For the rest of the section, let us assume that $\mathcal{G}$ is a complete graph, i.e. every pair of distinct vertices is connected by a unique edge. We proceed to show that the proposed algorithm becomes equivalent to the Distributed Projected Consensus algorithm and the results of [38] can be applied.

Assumption A.1. *$\mathcal{G}$ is complete and for all i and k, the weights $a^i_{j,k}(t)$, $j \in \mathcal{V}$, are equal i.e.*

$$a^i_{1,k}(t) = a^i_{2,k}(t) = \cdots = a^i_{m,k}(t)$$

If $\mathcal{G}$ is complete, then $I_i = \mathcal{V}$ for all i, i.e. all agents have the same set of variables. We can define the auxiliary sequence $\{y(t)\}$, where $y(t)$ is given by

$$y(t) = \frac{1}{m} \sum_{i=1}^{m} \widehat{r}^i(t) \tag{A.4}$$

Lemma A.3. *Assume that $\mathcal{S}_1, \cdots, \mathcal{S}_m$ are closed and convex, and $\mathcal{S}$ is non-empty. In addition, let Assumptions 3.1, 3.2 and A.1 hold. Then for all i,*

$$\lim_{t\to\infty} \left\| r^i(t) - y(t) \right\| = 0 \quad \textit{and} \quad \lim_{t\to\infty} \left\| \widehat{r}^i(t) - y(t) \right\| = 0$$

Proof. By Assumption A.1, we can define new weights $\widetilde{a}^i_k(t)$ where, for all i and k,

$$\widetilde{a}^i_k(t) = a^i_{j,k}(t), \quad j = 1, \cdots, m \tag{A.5}$$

such that our update rules (3.6) and (3.7) become

$$\widehat{r}^i(t) = \sum_{k=1}^{m} \widetilde{a}_k^i r^k(t) \tag{A.6}$$

$$r^i(t+1) = P_{\mathcal{S}_i}\left(\widehat{r}^i(t)\right) \tag{A.7}$$

We can verify that (A.5), (A.6) and (A.3) satisfy Assumptions 2, 3, 4 and 5 of [38]. The proof follows by the application of Lemma 4 in [38].

□

Proposition A.1. *Assume that $\mathcal{S}_1, \cdots, \mathcal{S}_m$ are closed and convex, and $\mathcal{S}$ is non-empty. In addition, let Assumptions 3.1, 3.2 and A.1 hold. Then for some $x^* \in \mathcal{S}$, we have*

$$\lim_{t\to\infty} \left\|r^i(t) - x^*\right\| = 0 \quad \textit{and} \quad \lim_{t\to\infty} \left\|\widehat{r}^i(t) - x^*\right\| = 0$$

Proof. The proof follows from the application of (A.5), Lemma A.3, and Proposition 2 of [38].

□

Appendix B

Electric Power System Analysis

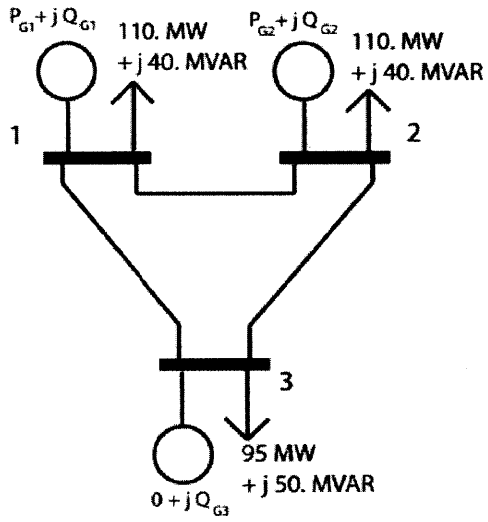


Figure B-1: 3-bus electric power grid [33]

Electric power grids can be thought of as electric circuits of nation, or even continent-wide, dimensions. The multivariate versions of Kirchoff's and Ohm's laws apply, which are overviewed using a matrix-vector notation. As electric power grids are alternating current circuits, all electric quantities involved are complex valued.

In power engineering nomenclature, a bus is a connection point or node in the electric power grid. It connects various electrical elements such as transmission lines, transformers, generating units and loads. Buses, which may have generating units

and loads connected to them, can inject or remove power from the grid. A bus is called a *generator bus* if it has one or more generating units connected to it and a *load bus* otherwise.

We restrict our analysis to a simplified one-line diagram of the electric power grid under steady-state, synchronized operation with only buses, lines, generating units and loads as shown in Figure B-1. All other electrical elements are assumed to have been absorbed into the line, generating unit, or load models and are not shown explicitly. Further, quantities are measured in the per unit (p.u.) system, which are assumed to be properly normalized. The p.u. system enables uniform three-phase analysis over the different voltage levels present in the electric power grid [25]. Thus, the grid can be represented abstractly by nodes (buses) and edges (lines) in a connected network.

B.1 Equivalent Π Circuit Model

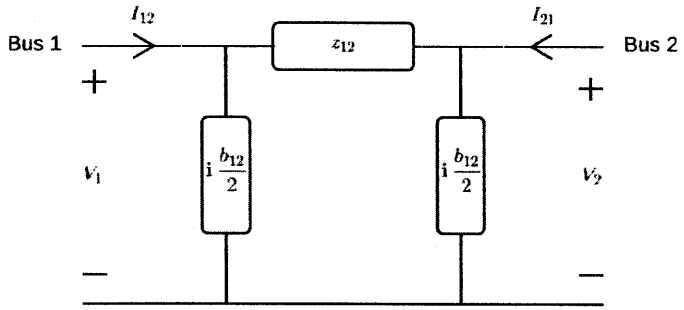


Figure B-2: Equivalent Π circuit

Consider a system that comprises of two buses, 1 and 2, connected by a line (also known as a branch), which may represent a transmission line or even a transformer. A line between two nodes is represented by the equivalent Π circuit model [4] depicted in Figure B-2.

The model entails the line series impedance z_{12} and a line charging susceptance

b_{12}. The line series impedance consists of a resistive part r_{12} and a reactive (inductive) part $x_{12} > 0$, so that $z_{12} = r_{12} + \mathbf{i}x_{12}$. The line series impedance $y_{12} := \frac{1}{y_{12}} = g_{12} + \mathbf{i}b_{12}$ is often used in place of the impedance.

Let V_1 and V_2 denote the complex voltages at buses 1 and 2 respectively, and I_{12} the complex current flow from bus 1 to bus 2. Invoking Ohm's and Kirchoff's laws on the circuit of Figure B-2, we obtain:

$$I_{12} = \mathbf{i}\frac{b_{12}}{2}V_1 + y_{12}\left(V_1 - V_2\right)$$

The reverse current flow I_{21} is expressed symmetrically. Note that unless b_{12} is zero, it holds that $I_{12} \neq I_{21}$.

Building on the two-bus system, consider next a network consisting of a set of buses $\mathcal{V} = \{1\ ,\ \cdots\ ,\ m\}$ and a set of lines $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$. By Kirchoff's current law, the complex current injected at bus i into the network must equal the sum of currents on the lines incident to bus i, i.e.,

$$\begin{aligned} I_i &= \sum_{j \in N(i)} I_{ij} \\ &= \sum_{j \in N(i)} \left\{ \mathbf{i}\frac{b_{ij}}{2}V_i + y_{ij}\left(V_i - V_j\right) \right\} \end{aligned}$$

where $N(i)$ is the set of buses incident to bus i, excluding i itself. Collecting complex bus voltages in the vector $V \in \mathbb{C}^m$ and complex bus currents in the vector $I \in \mathbb{C}^m$, we obtain the Multivariate Ohm's Law:

$$I = Y\,V \tag{B.1}$$

where $Y \in \mathbb{C}^{m \times m}$ is the bus admittance matrix given by

$$Y_{ij} = \begin{cases} \sum_{k \in N(i)} \left(\mathbf{i}\frac{b_{ik}}{2} + y_{ik} \right) & \text{if } i = j \\ -y_{ij} & \text{if } j \in N(i) \\ 0 & \text{otherwise} \end{cases}$$

Y is sparse and symmetric, but not necessarily Hermitian. We use a rectangular representation for the bus admittance matrix $Y := G + \mathbf{i}B$, where G is the bus conductance matrix and B is the bus susceptance matrix.

B.2 Power Flow Equations

A major implication of (B.1) is the control of power flows. We denote $S_{G,i} := P_{G,i} + \mathbf{i}Q_{G,i}$ to be the complex power produced at generator bus i and $S_{D,i} := P_{D,i} + \mathbf{i}Q_{D,i}$ to be the complex power demanded by bus i. It is convenient to define $S_i := S_{G,i} - S_{D,i}$ to be the complex power injected at bus i to the rest of the network.

Typically, the bus voltages and bus admittances are expressed in Cartesian coordinates, i.e. $V_i := V_{d,i} + \mathbf{i}V_{q,i}$ and $Y_{ij} = G_{ij} + \mathbf{i}B_{ij}$. From the definition of power and application of the Multivariate Ohm's Law (B.1), the complex power injected at bus i into the network is

$$\begin{aligned} S_i &= V_i I_i^H \\ &= V_i \left(\sum_{j=1}^{m} Y_{ij} V_j \right)^H && \text{(B.2)} \\ &= (V_{d,i} + \mathbf{i}V_{q,i}) \sum_{j=1}^{m} \left((G_{ij} - \mathbf{i}B_{ij}) (V_{d,i} - \mathbf{i}V_{q,i}) \right) && \text{(B.3)} \end{aligned}$$

If we resolve (B.3) into real and imaginary parts, we obtain the real and reactive

powers injected at bus i into the network:

$$P_i = V_{d,i} \sum_{j=1}^{m} (V_{d,j} G_{ij} - V_{q,j} B_{ij}) + V_{q,i} \sum_{j=1}^{m} (V_{q,j} G_{ij} + V_{d,j} B_{ij}) \tag{B.4}$$

$$Q_i = V_{q,i} \sum_{j=1}^{m} (V_{d,j} G_{ij} - V_{q,j} B_{ij}) - V_{d,i} \sum_{j=1}^{m} (V_{q,j} G_{ij} + V_{d,j} B_{ij}) \tag{B.5}$$

We can also express the bus voltages in polar coordinates, i.e., $V_i := |V_i| e^{\mathrm{i}\theta_i}$. If we resolve (B.2) into real and imaginary parts, we obtain alternative formulas for the real and reactive power injected at bus i into the network:

$$P_i = |V_i| \sum_{j=1}^{m} (|V_j| G_{ij} \cos\theta_{ij} + |V_j| B_{ij} \sin\theta_{ij}) \tag{B.6}$$

$$Q_i = |V_i| \sum_{j=1}^{m} (|V_j| G_{ij} \sin\theta_{ij} - |V_j| B_{ij} \cos\theta_{ij}) \tag{B.7}$$

where $\theta_{ij} := \theta_i - \theta_j$. It is common to linearize (B.6) and (B.7) to obtain the DC power flow equations. The DC model hinges on the following assumptions:

- The power network is purely inductive, so the conductance part of the bus admittance matrix is zero, i.e. $G = \mathbf{0}$.

- The voltage phase differences across directly connected buses are small, thus $\theta_{ij} \approx 0$ for every pair of neighboring buses (i, j) and the trigonometric functions in (B.6) and (B.7) are approximated by $\sin\theta_{ij} \approx \theta_i - \theta_j$ and $\cos\theta_{ij} \approx 1$.

- The magnitude of bus voltages is approximately one p.u., i.e. $|V_i| \approx 1$ for all i.

The DC model simplifies to:

$$\begin{aligned} P_i &= \sum_{j=1}^{m} B_{ij} (\theta_i - \theta_j) \\ &:= P_{G,i} - P_{D,i} \end{aligned} \tag{B.8}$$

B.3 Power Flow Problem

An important problem in electric power grid operation is the problem of power flow, where we want to solve for the steady-state powers and voltages of an electric power grid that are consistent with the power flow equations and within operational limits. The formulation of the power flow problem is based on operational limits of the grid as well as the power flow equations and circuit model described in the preceding sections.

Given a transmission network with a set of buses $\mathcal{V} = \{1\ ,\ \cdots\ ,\ m\}$ and a set of transmission lines $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$, the power flow problem is formulated as:

$$
\begin{aligned}
\text{find} \quad & V_{d,i},\ V_{q,i}, \quad \forall \quad i \in \mathcal{V} \\
\text{s.t.} \quad & P_{G,i}^{\min} \leq V_{d,i} \sum_{j=1}^{m} (V_{d,j} G_{ij} - V_{q,j} B_{ij}) + V_{q,i} \sum_{j=1}^{m} (V_{q,j} G_{ij} + V_{d,j} B_{ij}) + P_{D,i} \\
& \qquad\qquad \leq P_{G,i}^{\max} \quad \forall \quad i \in \mathcal{V} \\
& Q_{G,i}^{\min} \leq V_{q,i} \sum_{j=1}^{m} (V_{d,j} G_{ij} - V_{q,j} B_{ij}) - V_{d,i} \sum_{j=1}^{m} (V_{q,j} G_{ij} + V_{d,j} B_{ij}) + Q_{D,i} \\
& \qquad\qquad \leq Q_{G,i}^{\max} \quad \forall \quad i \in \mathcal{V} \\
& \left(V_i^{\min}\right)^2 \leq V_{d,i}^2 + V_{q,i}^2 \leq \left(V_i^{\max}\right)^2 \quad \forall \quad i \in \mathcal{V}
\end{aligned}
$$

where generator limits $P_{G,i}^{\min}$, $P_{G,i}^{\max}$, $Q_{G,i}^{\min}$ and $Q_{G,i}^{\max}$ and voltage magnitude limits $V_i^{\min}$ and $V_i^{\max}$ are assumed to be known. For load buses, we have $P_{G,i}^{\min} = P_{G,i}^{\max} = Q_{G,i}^{\min} = Q_{G,i}^{\max} = 0$.

The power flow problem seeks to find a set of complex bus voltages, $V_{d,i},\ V_{q,i},\ \forall\, i \in \mathcal{V}$, that is consistent with the power equations and operational limits and satisfies complex power demands at every bus. From the solution, real and reactive power generation can be recovered using the power flow equations (B.4) and (B.5).

B.4 Structure-Preserving Power Network Model

Consider a simplified model of an electric power grid consisting of n generating units $\{g_1, \cdots, g_n\}$, their associated n generator buses $\{b_1, \cdots, b_n\}$, and m load buses $\{b_{n+1}, \cdots, b_{n+m}\}$. The interconnection structure of the grid is encoded by a connected admittance-weighted graph. The generators form the vertex set of the graph and the edges represent lines between buses or internal connections between generator buses and their corresponding generating units

The Laplacian matrix associated with the admittance-weighted graph is the symmetric matrix:

$$\mathcal{L} = \begin{bmatrix} \mathcal{L}_{gg} & \mathcal{L}_{gl} \\ \mathcal{L}_{lg} & \mathcal{L}_{ll} \end{bmatrix} \in \mathbb{R}^{(2n+m)\times(2n+m)}$$

where the first n entries are associated with the generating units and the last $n+m$ entries are associated with the buses. $\mathcal{L}_{gg} \in \mathbb{R}^{n\times n}$ is diagonal, $\mathcal{L}_{ll} \in \mathbb{R}^{(n+m)\times(n+m)}$ is invertible and $\mathcal{L}_{lg} = \mathcal{L}_{gl}^T$.

Given the transient reactances z_i of the generating units,

$$\mathcal{L}_{gg} = \text{diag}\left(\frac{1}{z_1}, \cdots, \frac{1}{z_n}\right) \in \mathbb{R}^{n\times n}, \quad \mathcal{L}_{lg} = \begin{bmatrix} -\mathcal{L}_{gg} \\ \mathbf{0} \end{bmatrix} \in \mathbb{R}^{(n+m)\times n}, \quad \mathcal{L}_{gl} = \mathcal{L}_{lg}^\top$$

And given the bus susceptances B_{kj}, the element in the k-th row and j-th column of $\mathcal{L}_{ll}$ is can be derived using:

$$[\mathcal{L}_{ll}]_{kj} = \begin{cases} -B_{kj} & \text{if } j \neq k \\ \sum_{l\neq k} B_{kl} & \text{if } j = k \text{ and } k \text{ is a load bus} \\ \sum_{l\neq k} B_{kl} + \frac{1}{z_k} & \text{if } j = k \text{ and } k \text{ is a generator bus} \end{cases}$$

for $k = 1, \cdots, n+m$ and $j = 1, \cdots, n+m$.

A classical mathematical model to describe the behavior of the electric power grid in transient stability studies is the structure-preserving power network model [32, 42]. In this work, we consider the linearized small signal version of the structure-

preserving model, which consists of the linearized swing equations and the DC power flow equations (B.8) [43]. In addition, we use a Kron-reduced representation of the electric power grid to reduce the number of states. The model is given by the linear continuous-time descriptor system

$$\dot{x}(t) = \widetilde{A}_c \; x(t) + \widetilde{B}_c \; P(t)$$

where the state $x(t) = \begin{bmatrix} \delta^\top & \omega^\top \end{bmatrix}^\top \in \mathbb{R}^{2n}$ consists of the rotor angles $\delta \in \mathbb{R}^n$ and the frequencies $\omega \in \mathbb{R}^n$. The input term

$$P(t) = \begin{bmatrix} P_\omega(t) \\ P_\theta(t) \end{bmatrix} \in \mathbb{R}^{m+2n}$$

is due to known changes in mechanical input power to the generators $P_\omega(t)$ and power injections at the buses $P_\theta(t)$. Furthermore, the descriptor matrices are:

$$\widetilde{A}_c = \begin{bmatrix} 0 & I \\ M^{-1}\left(-\mathcal{L}_{gg} + \mathcal{L}_{gl}\mathcal{L}_{ll}^{-1}\mathcal{L}_{lg}\right) & -M^{-1}D \end{bmatrix} \in \mathbb{R}^{2n \times 2n}$$

$$\widetilde{B}_c = \begin{bmatrix} 0 & 0 \\ M^{-1} & M^{-1}\mathcal{L}_{gl}\mathcal{L}_{ll}^{-1} \end{bmatrix} \in \mathbb{R}^{2n \times (m+2n)}$$

where $M = \operatorname{diag}(M_1 \, , \, \cdots \, , \, M_n)$ and $D = \operatorname{diag}(D_1 \, , \, \cdots \, , \, D_n)$ are the diagonal matrices of the generating units' inertias and damping constants respectively.

Bibliography

[1] Nabil Adam. Workshop on future directions in cyber-physical systems security. In *Report on workshop organized by Department of Homeland Security (DHS)*, 2010.

[2] Heinz H. Bauschke, Patrick L. Combettes, and D. Russell Luke. Phase retrieval, error reduction algorithm, and fienup variants: a view from convex optimization. *J. Opt. Soc. Am. A*, 19(7):1334–1345, Jul 2002.

[3] Aharon Ben-Tal, Laurent El Ghaoui, and Arkadi Nemirovski. *Robust optimization.* Princeton University Press, 2009.

[4] Arthur R. Bergen and Vijay Vittal. *Power systems analysis.* Upper Saddle River, N.J. : Prentice Hall, c2000., 2000.

[5] Dimitris Bertsimas, David B Brown, and Constantine Caramanis. Theory and applications of robust optimization. *SIAM review*, 53(3):464–501, 2011.

[6] Dimitris Bertsimas and Martin S Copenhaver. Characterization of the equivalence of robustification and regularization in linear, median, and matrix regression. *arXiv preprint arXiv:1411.6160*, 2014.

[7] J. F. Bonnans. *Numerical optimization : theoretical and practical aspects.* Universitext. Berlin ; New York : Springer, c2006., 2006.

[8] S. Bose. Quadratically constrained quadratic programs on acyclic graphs with application to power flow. *Control of Network Systems, IEEE Transactions on*, PP(99):1–1, 2015.

[9] Stephen Boyd and Lieven Vandenberghe. *Convex Optimization.* Cambridge University Press, New York, NY, USA, 2004.

[10] James P. Boyle and Richard L. Dykstra. A method for finding projections onto the intersection of convex sets in hilbert spaces. In Richard Dykstra, Tim Robertson, and FarrollT. Wright, editors, *Advances in Order Restricted Statistical Inference*, volume 37 of *Lecture Notes in Statistics*, pages 28–47. Springer New York, 1986.

[11] L. M. Bregman. The method of successive projection for finding a common point of convex sets. *Soviet Math. Dokl.*, 162:688–692, 1965.

[12] G. Calafiore and L. El Ghaoui. Ellipsoidal bounds for uncertain linear equations and dynamical systems. *Automatica*, 40:773–787, 2004.

[13] A.A. Cárdenas, S. Amin, and S. Sastry. Research challenges for the security of control systems. In *Proceedings of the 3rd Conference on Hot Topics in Security*, HOTSEC'08, pages 6:1–6:6, 2008.

[14] A.A. Cárdenas, S. Amin, and S. Sastry. Secure control: Towards survivable cyber-physical systems. In *International Conference on Distributed Computing Systems Workshops*, pages 495–500, June 2008.

[15] J. Carpentier. Contribution to the economic dispatch problem. *Bulletin de la Societe Francoise des Electriciens*, 13(4):431–447, 1962. in French.

[16] Gregory Chockler, Murat Demirbas, Seth Gilbert, Calvin Newport, and Tina Nolte. Consensus and collision detectors in wireless ad hoc networks. In *Proceedings of the Twenty-fourth Annual ACM Symposium on Principles of Distributed Computing*, PODC '05, pages 197–206, New York, NY, USA, 2005. ACM.

[17] R. Christie. Power Systems Test Case Archive, University of Washington, Electrical Engineering. Online: http://www.ee.washington.edu/research/pstca/, 2000.

[18] Yichuan Ding. On efficient semidefinite relaxations for quadratically constrained quadratic programming. Master's thesis, University of Waterloo, 2007.

[19] Richard L. Dykstra. An algorithm for restricted least squares regression. *Journal of the American Statistical Association*, 78(384):837–842, December 1983.

[20] Yonina C. Eldar and Helmut Bölcskei. Block-sparsity: Coherence and efficient recovery. *CoRR*, abs/0812.0329, 2008.

[21] T. Erseghe, D. Zennaro, E. Dall'Anese, and L. Vangelista. Fast consensus by the alternating direction multipliers method. *Signal Processing, IEEE Transactions on*, 59(11):5523–5537, Nov 2011.

[22] H. Fawzi, P. Tabuada, and S. Diggavi. Secure estimation and control for cyber-physical systems under adversarial attacks. *Automatic Control, IEEE Transactions on*, 59(6):1454–1467, June 2014.

[23] Apostolos G Fertis. *A robust optimization approach to statistical estimation problems.* PhD thesis, Massachusetts Institute of Technology, 2009.

[24] AL Fradkov and VA Yakubovich. The s-procedure and duality relations in non-convex problems of quadratic programming. *Vestn. LGU, Ser. Mat., Mekh., Astron*, (1):101–109, 1979.

[25] G. Giannakis, V. Kekatos, N. Gatsis, Seung-Jun Kim, Hao Zhu, and B. Wollenberg. Monitoring and optimization for power grids: A signal processing perspective. *Signal Processing Magazine, IEEE*, 30(5):107–128, Sept 2013.

[26] Michel X. Goemans and David P. Williamson. Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming. *J. ACM*, 42(6):1115–1145, November 1995.

[27] Michael Grant and Stephen Boyd. Graph implementations for nonsmooth convex programs. In V. Blondel, S. Boyd, and H. Kimura, editors, *Recent Advances in Learning and Control*, Lecture Notes in Control and Information Sciences, pages 95–110. Springer-Verlag Limited, 2008.

[28] Michael Grant and Stephen Boyd. CVX: Matlab software for disciplined convex programming, version 2.1. http://cvxr.com/cvx, March 2014.

[29] Karolos M. Grigoriadis and Eric B. Beran. *Alternating Projection Algorithms for Linear Matrix Inequalities Problems with Rank Constraints*, chapter 13. Society for Industrial and Applied Mathematics, 2000.

[30] Nicholas J. Higham. Computing the nearest correlation matrixâĂŤa problem from finance. *IMA Journal of Numerical Analysis*, 22(3):329–343, 2002.

[31] Sunyoung Kim and Masakazu Kojima. Exact solutions of some nonconvex quadratic optimization problems via sdp and socp relaxations. *Computational Optimization and Applications*, 26(2):143–154, 2003.

[32] P. Kundur, Neal J. Balu, and Mark G. Lauby. *Power system stability and control.* EPRI power system engineering series. New York : McGraw-Hill, c1994., 1994.

[33] B.C. Lesieutre, D.K. Molzahn, A.R. Borden, and C.L. DeMarco. Examining the limits of the application of semidefinite programming to power flow problems. In *Communication, Control, and Computing (Allerton), 2011 49th Annual Allerton Conference on*, pages 1492–1499, Sept 2011.

[34] Adrian S. Lewis and Jérôme Malick. Alternating projections on manifolds. *Mathematics of Operations Research*, 33(1):216–234, 2008.

[35] A.S. Lewis, D.R. Luke, and J. Malick. Local linear convergence for alternating and averaged nonconvex projections. *Foundations of Computational Mathematics*, 9(4):485–513, 2009.

[36] Miguel Sousa Lobo, Lieven Vandenberghe, Stephen Boyd, and Hervé Lebret. Applications of second-order cone programming. *Linear Algebra and its Applications*, 284(1-3):193 – 228, 1998. International Linear Algebra Society (ILAS) Symposium on Fast Algorithms for Control, Signals and Image Processing.

[37] Y. Mo and B. Sinopoli. False data injection attacks in control systems. In *Workshop on Secure Control Systems*, 2010.

[38] A. Nedič, A. Ozdaglar, and P.A. Parrilo. Constrained consensus and optimization in multi-agent networks. *Automatic Control, IEEE Transactions on*, 55(4):922–938, April 2010.

[39] Y. Nesterov and A. Nemirovskii. *Interior-Point Polynomial Algorithms in Convex Programming.* Society for Industrial and Applied Mathematics, 1994.

[40] R. Olfati-Saber, J.A. Fax, and R.M. Murray. Consensus and cooperation in networked multi-agent systems. *Proceedings of the IEEE*, 95(1):215–233, Jan 2007.

[41] M. Pajic, J. Weimer, N. Bezzo, P. Tabuada, O. Sokolsky, I. Lee, and G. Pappas. Robustness of attack-resilient state estimators. In *ACM/IEEE International Conference on Cyber-Physical Systems (ICCPS)*, pages 163–174, April 2014.

[42] F. Pasqualetti, F. Dörfler, and F. Bullo. Attack detection and identification in cyber-physical systems. *IEEE Transactions on Automatic Control*, 58(11):2715–2729, Nov 2013.

[43] Fabio Pasqualetti, Florian DÃűrfler, and Francesco Bullo. Cyber-physical attacks in power networks: Models, fundamental limitations and monitor design. In *CDC-ECE*, pages 2195–2201. IEEE, 2011.

[44] M.J.D. Powell and Y. Yuan. A trust region algorithm for equality constrained optimization. *Mathematical Programming*, 49(1-3):189–211, 1990.

[45] N. Z. Shor. Quadratic optimization problems. *Soviet Journal of Circuits and Systems Sciences*, 25(6):1–11, 1987.

[46] Dan Simon. *Optimal state estimation : Kalman, H [infinity] and nonlinear approaches.* Hoboken, N.J. : Wiley-Interscience, c2006., 2006.

[47] H. Tang, F.R. Yu, M. Huang, and Z. Li. Distributed consensus-based security mechanisms in cognitive radio mobile ad hoc networks. *Communications, IET*, 6(8):974–983, May 2012.

[48] J. Von Neumann. *Functional Operators.* Annals of mathematics studies. Princeton University Press, 1950.

[49] Lihua Xie, Yeng Chai Soh, and Carlos E de Souza. Robust Kalman filtering for uncertain discrete-time systems. *IEEE Transactions on Automatic Control*, 39(6):1310–1314, 1994.

[50] Huan Xu, Constantine Caramanis, and Shie Mannor. Robust regression and Lasso. In *Advances in Neural Information Processing Systems*, pages 1801–1808, 2009.

[51] Yinyu Ye and Shuzhong Zhang. New results on quadratic minimization. *SIAM Journal on Optimization*, 14(1):245–267, 2003.

[52] Makoto Yokoo. *Electronic commerce. [electronic resource] : theory and practice.* Studies in computational intelligence: v. 10. Berlin : Springer, c2008., 2008.

[53] S.Z. Yong, M. Zhu, and E. Frazzoli. Resilient state estimation against switching attacks on stochastic cyber-physical systems. In *IEEE Conference on Decision and Control (CDC)*, 2015. submitted.

[54] M. Zhu and S. Martínez. On distributed constrained formation control in operator-vehicle adversarial networks. *Automatica*, 49(12):3571–3582, 2013.

[55] R.D. Zimmerman, C.E. Murillo-Saànchez, and R.J. Thomas. Matpower: Steady-state operations, planning, and analysis tools for power systems research and education. *Power Systems, IEEE Transactions on*, 26(1):12–19, Feb 2011.