KTH Electrical Engineering

Toward Cyber-Secure and Resilient Networked Control Systems

ANDRÉ TEIXEIRA

Doctoral Thesis
Stockholm, Sweden 2014

KTH Royal Institute of Technology
School of Electrical Engineering
Department of Automatic Control
SE-100 44 Stockholm
SWEDEN

TRITA-EE 2014:055
ISSN 1653-5146
ISBN 978-91-7595-319-9

Akademisk avhandling som med tillstånd av Kungliga Tekniska högskolan framläggges till offentlig granskning för avläggande av teknologie doktorsexamen i Reglerteknik fredagen den 07:e november 2014, klockan 14:00, i sal F3, Kungliga Tekniska högskolan, Lindstedtsvägen 26, Stockholm.

Tryck: Universitetsservice US AB

Abstract

Resilience is the ability to maintain acceptable levels of operation in the presence of abnormal conditions. It is an essential property in industrial control systems, which are the backbone of several critical infrastructures. The trend towards using pervasive information technology systems, such as the Internet, results in control systems becoming increasingly vulnerable to cyber threats. Traditional cyber security does not consider the interdependencies between the physical components and the cyber systems. On the other hand, control-theoretic approaches typically deal with independent disturbances and faults, thus they are not tailored to handle cyber threats. Theory and tools to analyze and build control system resilience are, therefore, lacking and in need to be developed. This thesis contributes towards a framework for analyzing and building resilient control systems.

First, a conceptual model for networked control systems with malicious adversaries is introduced. In this model, the adversary aims at disrupting the system behavior while remaining undetected by an anomaly detector The adversary is constrained in terms of the available model knowledge, disclosure resources, and disruption capabilities. These resources may correspond to the anomaly detector's algorithm, sniffers of private data, and spoofers of control commands, respectively.

Second, we address security and resilience under the perspective of risk management, where the notion of risk is defined in terms of a threat's scenario, impact, and likelihood. Quantitative tools to analyze risk are proposed. They take into account both the likelihood and impact of threats. Attack scenarios with high impact are identified using the proposed tools, e.g., zero-dynamics attacks are analyzed in detail. The problem of revealing attacks is also addressed. Their stealthiness is characterized, and how to detect them by modifying the system's structure is also described.

As our third contribution, we propose distributed fault detection and isolation schemes to detect physical and cyber threats on interconnected second-order linear systems. A distributed scheme based on unknown input observers is designed to jointly detect and isolate threats that may occur on the network edges or nodes. Additionally, we propose a distributed scheme based on local models and measurements that is resilient to changes outside the local subsystem. The complexity of the proposed methods is decreased by reducing the number of monitoring nodes and by characterizing the minimum amount of model information and measurements needed to achieve fault detection and isolation.

Finally, we tackle the problem of distributed reconfiguration under sensor and actuator faults. In particular, we consider a control system with redundant sensors and actuators cooperating to recover from the removal of individual nodes. The proposed scheme minimizes a quadratic cost while satisfying a model-matching condition, which maintains the nominal closed-loop behavior after faults. Stability of the closed-loop system under the proposed scheme is analyzed.

Populär sammanfattning

Ett resilient system har förmågan att återhämta sig efter en kraftig och oväntad störning. Resiliens är en viktig egenskap hos industriella styrsystem som utgör en viktig komponent i många kritiska infrastrukturer, såsom processindustri och elkraftnät. Trenden att använda storskaliga IT-system, såsom Internet, inom styrsystem resulterar i en ökad sårbarhet för cyberhot. Traditionell IT-säkerhet tar inte hänsyn till den speciella koppling mellan fysikaliska komponenter och IT-system som finns inom styrsystem. Å andra sidan så brukar traditionell reglerteknik fokusera på att hantera naturliga fel och inte cybersårbarheter. Teori och verktyg för resilienta och cybersäkra styrsystem saknas därför och behöver utvecklas. Denna avhandling bidrar till att ta fram ett ramverk för att analysera och konstruera just sådana styrsystem.

Först så tar vi fram en representativ abstrakt modell för nätverkade styrsystem som består av fyra komponenter: den fysikaliska processen med sensorer och ställdon, kommunikationsnätet, det digitala styrsystemet och en feldetektor. Sedan införs en konceptuell modell för attacker gentemot det nätverkade styrsystemet. I modellen så beskrivs attacker som försöker undgå att skapa alarm i feldetektorn men ändå stör den fysikaliska processen. Dessutom så utgår modellen ifrån att den som utför attacken har begränsade resurser i fråga om modellkännedom och kommunikationskanaler.

Det beskrivna ramverket används sedan för att studera resilens gentemot attackerna genom en riskanalys, där risk definieras utifrån ett hots scenario, konsekvenser och sannolikhet. Kvantitativa metoder för att uppskatta attackernas konsekvenser och sannolikheter tas fram, och speciellt visas hur hot med hög risk kan identifieras och motverkas. Resultaten i avhandlingen illustreras med ett flertal numeriska och praktiska exempel.

Acknowledgements

Research is a social activity! This thought stems from the great interactions with the several contributors to this thesis, whom I would like to acknowledge.

As a student, the initial contact with research comes from advisors. I am fortunate to have both Henrik Sandberg and Karl H. Johansson taking that role. Their guidance and scientific insights have been a continuous source of inspiration throughout my journey in academic research.

An integral part of research is to exchange and discuss ideas. In this regard, I am truly grateful for the fantastic visiting periods spent abroad. I am thankful to Pangun Park, Saurabh Amin, Annarita Giani, Kameshwar Poolla, and Shankar Sastry for the time I spent at UC Berkeley in different occasions, and the great discussions we had. A kind thanks to James Anderson and Antonis Papachristodoulou for the wonderful moments at the University of Oxford.

I would like to thank all my collaborators and co-authors for the fruitful discussions and struggles we shared. I must say, I am indebted for all the promising ideas we have come across and struggled with... the ones that worked and the ones that did not, they all were a valuable source of knowledge and inspiration. A special thanks to Saurabh Amin, James Anderson, José Araújo, György Dán, Farhad Farokhi, Euhanna Ghadimi, Mikael Johansson, Morten Juelsgaard, Cedric Langbort, Antonis Papachristodoulou, Daniel Pérez, Iman Shames, and Kin Cheong Sou.

I am particularly thankful for the fantastic environment at the Department of Automatic Control at KTH. It is truly amazing how many exciting discussions and collaborations started during coffee breaks, or on the way to lunch. Thank you to José Araújo for motivating the beginning of my journey abroad and for his treasured friendship. Many thanks to my current and former colleagues, in particular Assad Alam, Mariette Annergren, Themistoklis Charalambous, Phoebus Chen, Burak Demirel, Farhad Farokhi, António Gonga, Euhanna Ghadimi, Per Hägg, Christian Larsson, Piergiuseppe Di Marco, Pangun Park, Demia Della Penda, Chithrupa Ramesh, Iman Shames, Pablo Soldati, Kin Cheong Sou, Christopher Sturk, Jana Tumova, and Jim Weimer, for all the support and helpful discussions.

A special thanks to José Araújo, Pedro Lima, Chithrupa Ramesh, and Patricio Valenzuela for carefully reviewing parts of this thesis.

The work leading to this thesis has been financially supported by the EU FP7 project VIKING, the Swedish Research Council, the Swedish Foundation for Strategic Research, and the Swedish Governmental Agency for Innovation Systems. Their support is greatly acknowledged.

None of this would have been possible without the support from my family and friends. A great thanks to the Família Tuga, for making Stockholm a warmer and brighter city. Special thanks to my parents Manuel and Maria Eugénia and my brother Daniel for their support and nurturing throughout my life. My dearest love to my wife Cátia, for her tremendous support and unwavering energy.

André

Contents

Notation

$\mathbb{R}$	Set of real numbers
$\mathbb{C}$	Set of complex numbers
$\mathbb{Z}$	Set of integer numbers
$\mathbb{R}^{m \times n}$	Set of matrices with m rows, n columns, and entries in $\mathbb{R}$
$\emptyset$	Empty set
$\lvert\mathcal{V}\rvert$	Cardinality of the set $\mathcal{V}$
$x \in \mathbb{R}^{n_x}$	Real-valued column vector of dimension n_x
x_i	The i-th entry of the vector x
t	Continuous-time instant, real-valued
k	Discrete-time instant, integer-valued
$x(t)$	Continuous-time vector variable
x_k	Discrete-time vector variable
$x_{(i),k}$	The i-th entry of the discrete-time vector x_k
$\lVert x\rVert_p$	The p-norm of vector x, for $p \geq 1$
$\mathbf{x}_{[0,\ N]}$	Discrete-time signal of x_k from $k = 0$ to $k = N$
$\mathbf{x}_{(i),\ [0,\ N]}$	Discrete-time signal of $x_{(i),k}$ from $k = 0$ to $k = N$
$\operatorname{Im}(A)$	Range-space of matrix A
$\operatorname{Ker}(A)$	Null-space of matrix A
$\dim(\mathcal{Z})$	Dimension of the subspace $\mathcal{Z}$
$A \otimes B$	Kronecker product of matrices A and B
$A \succ 0$	Positive definite matrix A
$A \succeq 0$	Positive semi-definite matrix A
$\operatorname{vec}(A)$	Vectorization of matrix A
$\operatorname{tr}(A)$	Trace of matrix A
$A^\top$	Transpose of matrix A
A^H	Hermitian conjugate of $A \in \mathbb{C}^{n \times m}$
$\lVert A\rVert_F$	Frobenius norm of matrix A, $\lVert A\rVert_F = \sqrt{\operatorname{tr}\left(A^H A\right)}$
$A^\dagger$	Moore-Penrose pseudo-inverse of A
$\Re(x)$	Real part of the complex number $x \in \mathbb{C}$

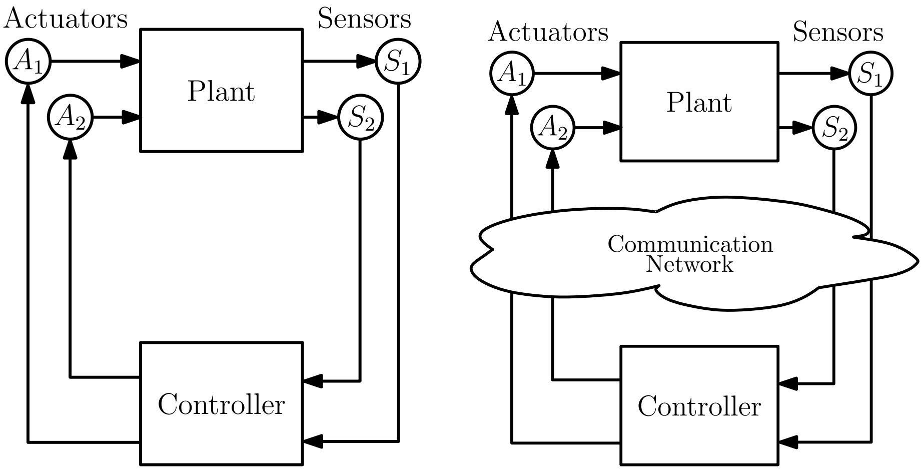
Chapter 1

Introduction

Feedback control is essential in modern societies, being a core component of electronic devices, vehicles, industrial plants, and large-scale critical infrastructures such as the electric power network. The ubiquitous use of automatic control is very much due to the technological developments in computation, actuation, and sensing, together with a strong theoretical development of the field over the recent decades (Åström and Kumar, 2014). The simplest instance of a feedback control system consists of two blocks, as illustrated in Figure 1.1a: a physical plant, with sensors measuring its relevant variables and actuators driving its behavior, and a controller that computes the control signal to be applied to the plant. Such a representation accurately captures the essence of control systems until the 1960s, when feedback controllers were comprised of mechanical or analog electronic devices with reliable sensor-to-controller and controller-to-actuator links.

The technological development during the digital era since the 1960s has led to the increased use of digital controllers and communication networks in many control applications, effectively transforming them into networked control systems (NCS), as depicted in Figure 1.1b (Samad *et al.*, 2007). The digital revolution led to several opportunities to increase the overall efficiency of control systems, as well as their successful use in many domains (Samad and Annaswamy, 2011). Using information technology (IT) infrastructures, digital controllers, sensors, and actuators from the low-level control layer could now be integrated with high-level supervisory layers, giving birth to supervisory control and data acquisition (SCADA) systems. As illustrated in Figure 1.2, the lower layers of SCADA systems consist of sensors and actuators interfaced with programmable logic computers (PLC) at local stations, or with remote terminal units (RTU) imbued with extended communication capabilities at remote locations. Measurement data are collected by RTUs and PLCs and transmitted to the higher layers of the SCADA system through heterogeneous communication networks. Low-level control may be implemented in the PLCs or RTUs, which receive supervisory commands and set-points from the higher levels.

In addition to facilitating communication between different hierarchical layers, SCADA systems provide also other functionalities, such as human-machine inter-



(a) Control system with ideal communication links.

(b) Control system with a communication network.

Figure 1.1: Schematic of control systems with ideal and imperfect communication links.

faces (HMI), workstations, historian databases, and integration with corporate IT systems. These components have become an integral part of modern industrial control systems (ICS), enabling an efficient and flexible operation of the physical system. A typical ICS architecture is depicted in Figure 1.2.

Novel challenges surface with the tighter integration of IT in control systems and the use of pervasive technologies, such as the Internet and wireless communication. As the use of these technologies increase, their effects become more noticeable in the closed-loop system behavior. To tackle problems such as packet losses and delays, among many others, new theoretical foundations have been established over the past years (Baillieul and Antsaklis, 2007; Hespanha *et al.*, 2007). More recently, a new concern has come into focus: that of security and resilience of control systems against malicious adversaries (Samad *et al.*, 2007; Rieger *et al.*, 2009; Åström and Kumar, 2014).

1.1 Threats Against Industrial Control Systems

There exist several threats to ICS, both physical and cyber, be they unintentional or malicious. A key feature of a resilient control system is its ability to maintain state awareness and acceptable performance under unexpected events (Rieger *et al.*, 2009). Failing to achieve such properties can have dire consequences, as illustrated by the U.S.-Canada Northeastern blackout in 2003, which resulted in severe economical losses (U.S.-Canada PSOTF, 2004). Although the blackout was triggered

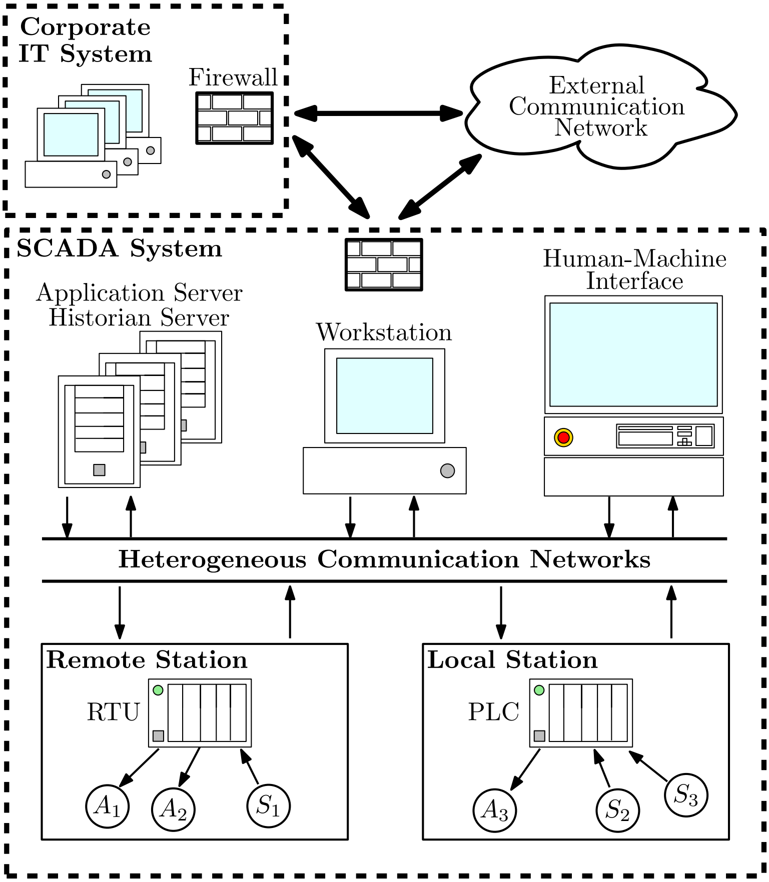


Figure 1.2: Schematic of a typical ICS architecture with the SCADA and corporate IT systems. The corporate IT system is connected to the supervisory layer of the SCADA system through firewalls. At the SCADA system's supervisory level, historian databases and software application servers enable the efficient operation of the ICS. The workstation and HMI are used to configure and monitor the low-level components, respectively. In the lower layer, local stations have programable logic controllers (PLC), typically with wired communication capabilities. The PLC receives measurements from sensors (S_i) and controls the physical system through actuators (A_i). A similar description applies to the remote station, where PLCs are replaced with remote terminal units (RTU) with extended communication capabilities, e.g., wireless communication or wired Internet access. The layers within the SCADA system are connected through heterogeneous communication networks, using wired and wireless communications. The ICS may be connected through firewalls to external networks and systems, such as other ICS and remote stations. (The figure is adapted from U.S. GAO (2004).)

by natural events and malfunctioning monitoring algorithms, one may envision comparable consequences resulting from deliberate threats against the system. To demonstrate the possible impact of cyber threats on control systems, the Idaho National Lab conducted the Aurora project, where a staged cyber attack on a diesel generator was performed (Meserve, 2007). As a consequence of the cyber attack, the mechanical vibrations of the generator substantially increased and resulted in physical damage to the machine.

Security against cyber threats is a classical concern for IT systems (Bishop, 2002). Therefore, the security concern is expected to carry over to ICS, as they increasingly rely on IT infrastructures. However, cyber security of ICS has not been a major concern during the past couples of decades (Samad *et al.*, 2007), for which Krutz (2006) points a few reasons: Legacy SCADA systems were somewhat isolated from external communication networks and were based on custom proprietary hardware and software, which conferred them a reasonable level of "security by obscurity". Additionally, security has been perceived as a low-priority domain from an economical perspective, given the reduced number of ICS-related security incidents reported over the last decades (U.S. GAO, 2004).

Recently, the awareness and concern over security of ICS has been increased. Modern SCADA systems have moved towards the use of standard communication technologies, to enable access to remote devices and to facilitate a smooth interface between devices from different vendors. Consequently, the number of possible attack points for malicious cyber agents to exploit have greatly increased. Another common practice is the use of standard hardware and software platforms to decrease costs and improve flexibility. New vulnerabilities of these standard platforms may be discovered over their life-cycle, which greatly increases the risk of cyber threats to a large number of SCADA systems. In fact, the number of reported ICS-related security incidents has significantly increased over the recent years, as depicted in Figure 1.3.

The best practices and techniques from IT security are a sound first approach to increase the security and resilience of ICS. However, traditional IT security does not consider the interdependencies between the physical components and the cyber domain. A holistic approach is required to effectively handle the complex coupling between the physical process and the IT infrastructure.

1.2 Motivational Examples

In the following, several examples are used to illustrate the importance of resilience in control systems.

Malware tailored against ICS

Staged cyber attacks have succeeded in physically damaging generators in test facilities (Meserve, 2007). Despite being a mock threat staged in a contained environment, it was one of the first "proofs of concept" for cyber attacks on ICS.

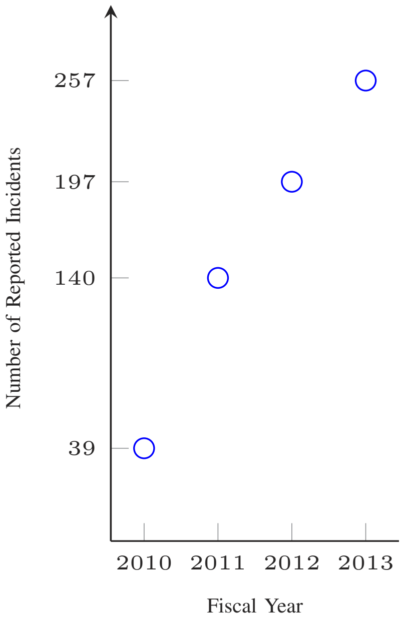


Figure 1.3: Number of cyber security incidents in industrial control systems voluntarily reported to ICS-CERT over the fiscal years 2010 through 2013 (ICS, 2013).

Another evidence came from the search engine Shodan, created in 2009, which finds electronic devices connected to the Internet (Matherly, 2009), such as routers, printers, computers, and PLCs. Several ICS devices were found and located using Shodan (Shefte *et al.*, 2012), which raised concerns regarding the exposure of ICS to the Internet and external threats (ICS-CERT, 2010). Since then, several advanced threats targeting ICS were reported. Discovered in 2010, the Stuxnet malware was designed to infiltrate SCADA systems with specific hardware and software components (Falliere *et al.*, 2011). The alleged aim of Stuxnet was to physically damage heavy machinery like steam turbines and gas centrifuges present in process plants by interfering with low-level actuators (Rid, 2011). The malware Duqu and Flame were found in 2011 and 2012, respectively (Symantec, 2011, 2012). Some components of these malware appears to be based on Stuxnet's source code and were aimed at espionage attacks, in an attempt to obtain sensitive information for facilitating future attacks. In 2013, Symantec discovered and monitored the actions of a cyber espionage group named Dragonfly (Symantec, 2014), which targeted mainly organizations within the energy sector and ICS software producers.

Out of all the malware threatening control systems, the one that sparked most amazement and concern was Stuxnet, not only because it was the first publicly known malware targeting ICS, but also due to its great complexity and functionalities. In the example below, we revisit some of the details regarding Stuxnet.

Example 1.1
Stuxnet was discovered in 2010 and has been closely examined since then (Falliere *et al.*, 2011). It is the first known malware tailored to compromise PLC software and it has raised several concerns due to its astonishing capabilities:

- four zero-day exploits (flaws previously unknown to the software developers);
- Windows rootkits (software to grant the malware with privileged rights and hide its existence from detection software);
- first infection through USB drive;
- infected devices can spread the malware through local networks;
- peer-to-peer communication between infected devices;
- self-update capabilities using the Internet and peer-to-peer communications;
- remains dormant and continues spreading until a specific PLC software is found;
- first known PLC rootkit;
- ability to modify PLC software and hide the modified code.

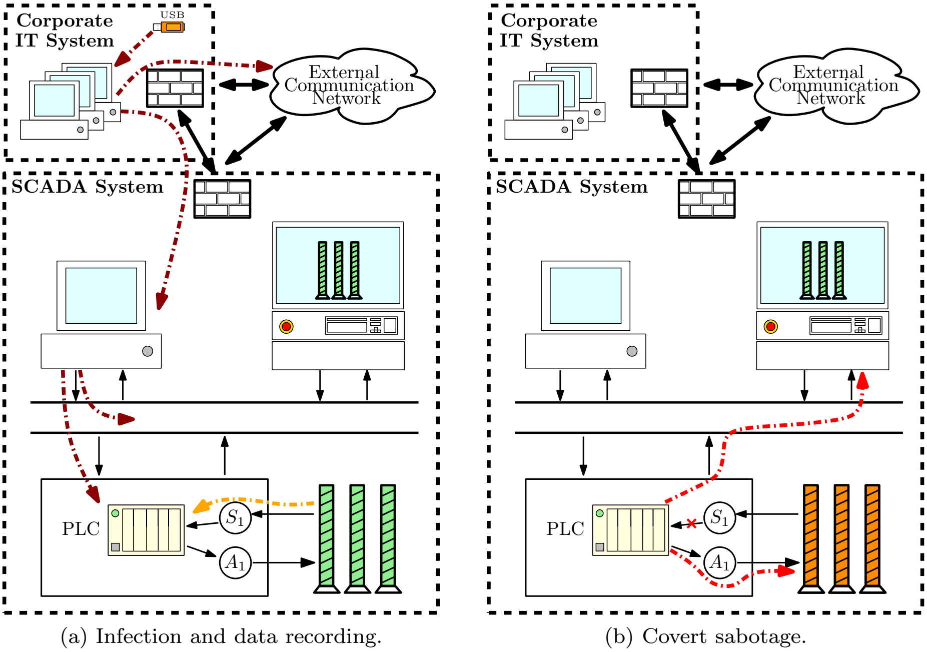
Further analysis of Stuxnet shed light on its main goal and operation, from which plausible attack scenarios can be constructed. In particular, the attack scenario described in Figure 1.4 has allegedly occurred in reality (Kushner, 2013). This scenario illustrates the complex behavior of Stuxnet and the potential damage it could have.

As concluded by Falliere *et al.* (2011) after a detailed analysis of the malware's capabilities and behavior, Stuxnet contains several interesting features: a resourceful and knowledgeable adversary, who aims at covertly disrupting the physical system. These features will be considered in several attack scenarios throughout this thesis.

False-data injection attack against power systems

Power transmission networks are complex and spatially distributed systems, as illustrated in Figure 1.5. They are operated through SCADA systems and are complemented by a set of application specific software, usually called energy management systems (EMS), enabling state and measurement estimation and optimal operation under safety and reliability constraints.

As discussed in Giani *et al.* (2009), there are several vulnerabilities in the SCADA system architecture, see Figure 1.6. They include RTUs (A1 and A5), communication networks between the RTUs and the control center (A2 and A6), and the IT software and databases in the control center (A3). In fact, there are



(a) Infection and data recording.

(b) Covert sabotage.

Figure 1.4: Three stages of the Stuxnet attack scenario: infection (dark-red line), data recording (orange line), and sabotage (red line). (a) Exploiting zero-day flaws, Stuxnet is able to compromise computers through an infected USB drive. Once a device is infected, Stuxnet attempts to update its code from the Internet. Unless the compromised device has the specific platform targeted by Stuxnet, the malware remains dormant and continues spreading infection. Using compromised digital certificates, Stuxnet is able to bypass firewalls and it continues spreading itself through the local communication networks of the SCADA system. Stuxnet's peer-to-peer communication capabilities allows the malware to update itself, even when the compromised device does not have direct access to the Internet. Once the targeted PLC is infected, Stuxnet changes its operation mode. Using the PLC rootkit, the malware modifies the PLC code to perform a disclosure attack and record the received data. (b) After recording data for some time, Stuxnet begins sabotaging the physical system through a disruption attack. While changing the control signal sent to the actuators, Stuxnet hides the damage to the plant by feeding the previously recorded data to the SCADA's monitoring systems.

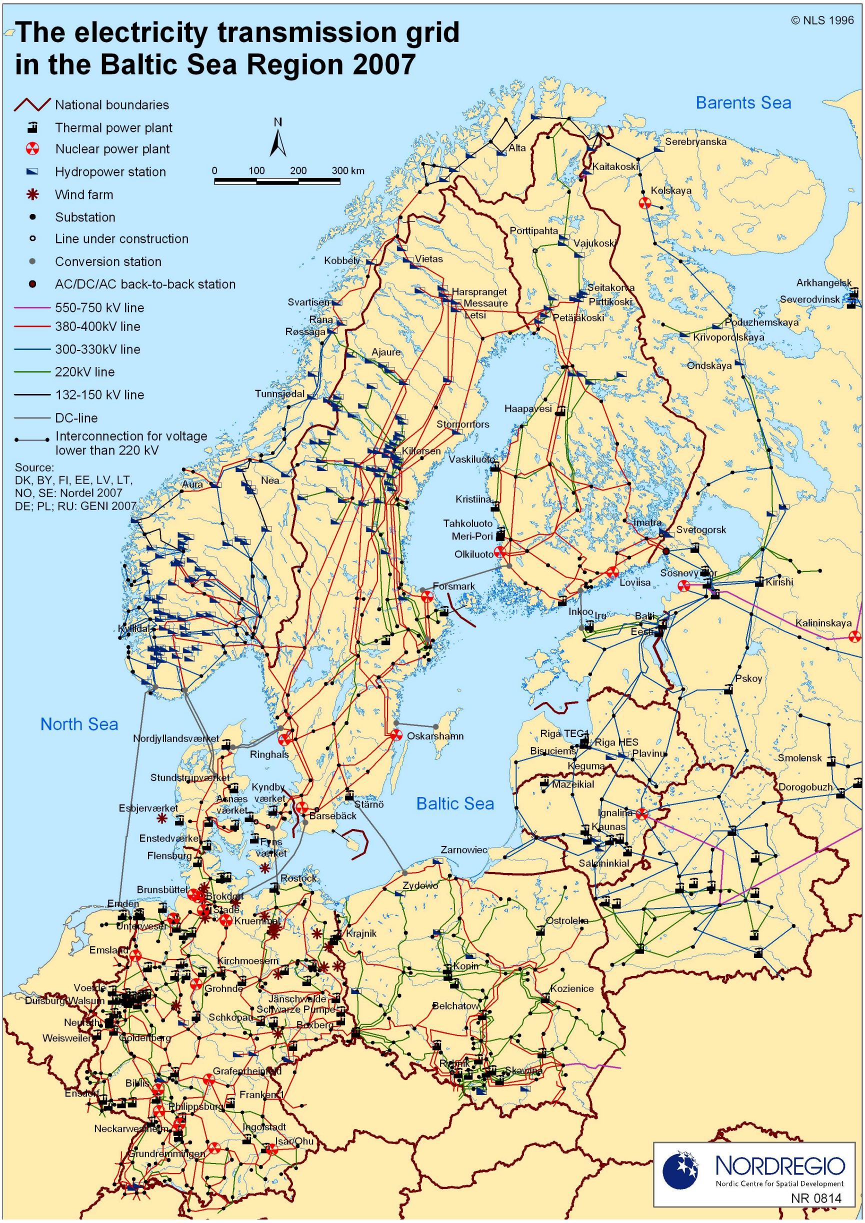


Figure 1.5: The electricity transmission grid in the Baltic Sea Region. Figure provided courtesy of Nordregio (Source: `www.nordregio.se`), Designer: P. G. Lindblom.

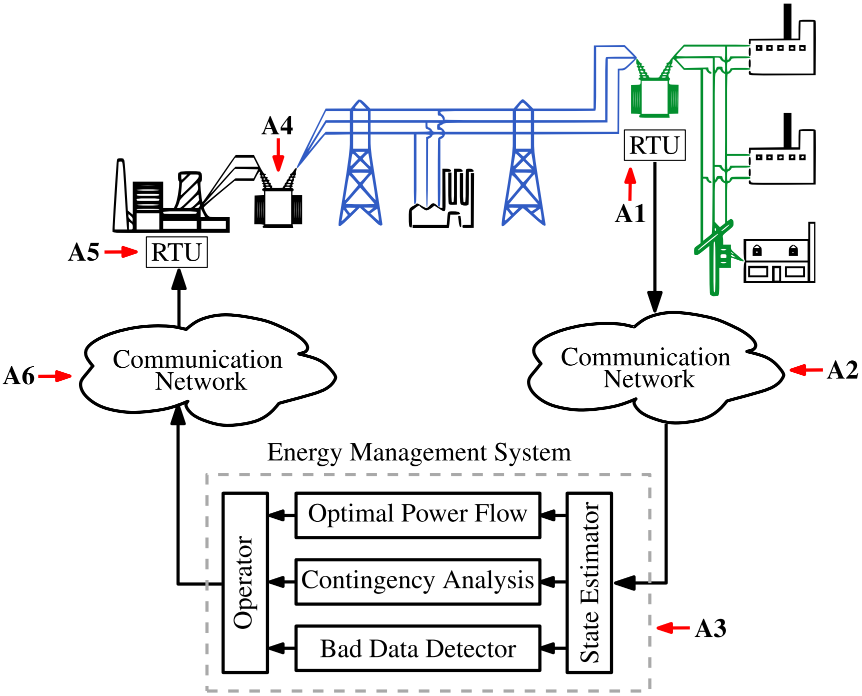


Figure 1.6: Schematic diagram of the electric power transmission network and its SCADA system with possible IT vulnerabilities. Measurements taken from the RTUs are sent through the SCADA system to the control center. The received measurements are used by several EMS applications, which provide state-awareness and control recommendations to human operators. The human operators decide the appropriate control actions and apply them through the SCADA system. (The figure is adapted from U.S.-Canada PSOTF (2004).)

several reports regarding cyber attacks on SCADA systems operating power networks (Gorman, 2009; CBSNews, 2009).

The supervisory operation of some power networks is market-driven, meaning that the prices paid to power producers vary according to the current estimated state of the system and the available resources. The California electricity crisis in 2000–2001 (FERC, 2003), a consequence of both a flawed market design and covert market manipulations, shows that there may exist economic incentive to tamper with the power system operation.

Protecting critical infrastructures such as power transmission networks raises several challenges. Given the large scale of these systems, there exist numerous potential attack points that may be compromised by adversaries. The components of such systems have long life-cycles and, consequently, there exists plenty of legacy

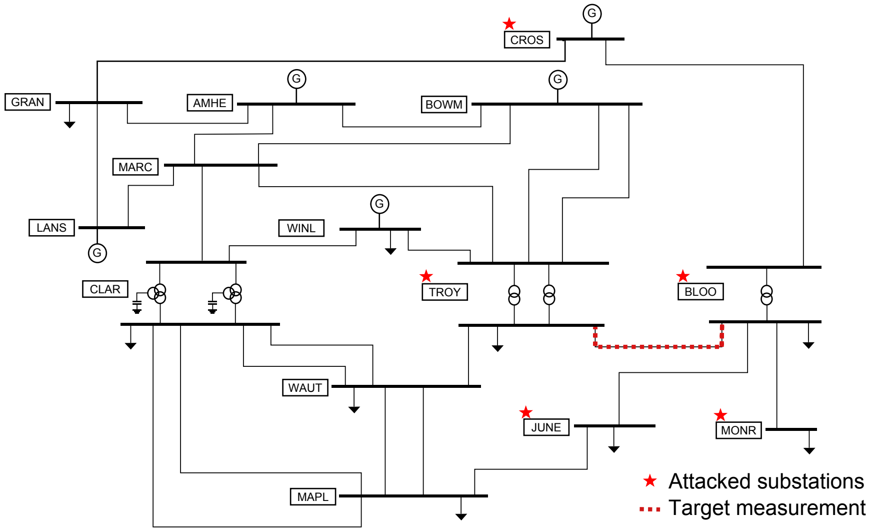


Figure 1.7: Power network considered in Example 1.2. The adversary targets the measured power flow along the red dashed transmission line. To covertly corrupt the target measurement, the adversary performs a coordinated attack on the highlighted substations.

equipment with no cyber protection capabilities. Therefore, efficient approaches to security and resilience are required for these systems. Their importance is illustrated with the following example.

Example 1.2
In this example, a false-data injection attack is carried out on a SCADA EMS software, where the adversary corrupts the sensor measurements gathered by the SCADA system. The EMS software has been configured for the power transmission network presented in Figure 1.7. This network consists of 14 substations and the bus-branch model has 27 buses and 40 branches. Several measurements are available at each substation and are kept in the software database. Specific EMS components are present, such as the state estimator (SE), bad-data detection (BDD), and contingency analysis (CA), as described by Shahidehpour *et al.* (2005).

The adversary desires to covertly corrupt the active power flow measurement from the tie-line between the substations TROY and BLOO. To remain undetected, the adversary must inject false measurements into the SCADA system in a coordinated way, so that the corrupted measurements conform with the model and topology of the power network. In essence, the false-data injected attack remains undetected if the corrupted measurements mimic a feasible state of the power network. Using a simplified network model, the adversary designs a coordinated attack

Table 1.1: Results from the stealthy attack for large bias from Example 1.2

Target bias, (MW)	False value (MW)	Estimate (MW)	#BDD Alarms	#CA Alarms
0	−14.8	−14.8	0	2
50	35.2	36.2	0	2
100	85.2	86.7	0	10
150	135.2	137.5	0	27
200	185.2	—	—	—

that corrupts the target measurement and a few additional ones. This attack only requires the corruption of 7 measurements in total, which are taken from 5 neighboring substations, namely TROY, BLOO, JUNE, MONR, and CROS. Since these corrupted measurements mimic a feasible set of power flows between the substations, the attack bypasses detection.

Table 1.1 shows the results obtained for large a bias injected in the target measurement, when the attack is performed sequentially with steps of 50MW. Observe that the covert attack is successful, with no BDD alarm triggered up to a bias of 150MW, beyond which the SE no longer converges. To better understand the size of this bias, note that the nominal value of the targeted tie-line is 260MW. Thus the attack is able to induce a bias of more than 50% of the nominal value, which reveals that the SCADA EMS software is indeed sensitive to covert false-data injection attacks. Furthermore, the number of warnings given by the CA component increases with the size of the attack. The increased number of CA warnings could lead the operator to take corrective actions, as the CA warnings indicate that the system does not meet the reliability criteria. On the other hand, the optimal power flow (OPF) algorithm would give the operator misleading recommendations, computed based on the compromised state estimate.

The attack scenario from the example shows that a large-scale system may be compromised with only a handful of compromised measurements. Deploying protective resources in these systems requires the identification of the most vulnerable devices in an efficient manner.

1.3 Problem Formulation

This thesis addresses the problem of cyber security and resilience in networked control systems. Traditional cyber security does not consider the interdependencies between the physical components and the cyber systems. On the other hand, control-theoretic approaches typically deal with independent disturbances and faults, thus

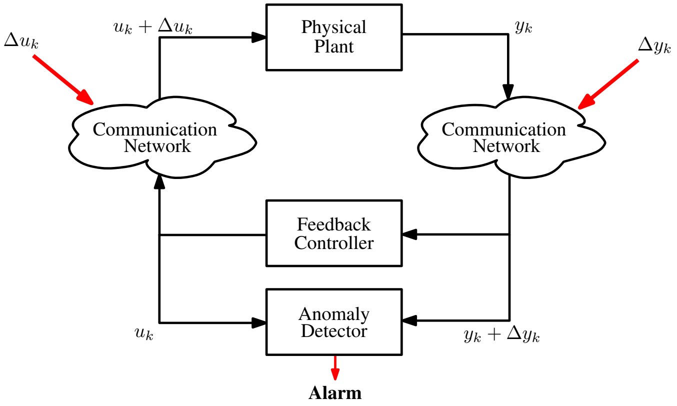


Figure 1.8: Schematic of a networked control system with a communication network that is vulnerable to adversaries.

they are not tailored to handle cyber threats. Theory and tools to analyze and build control system resilience are, therefore, lacking and in need to be developed.

Throughout the thesis we consider a reference architecture of networked control systems with the following four components: the physical plant with sensors and actuators, the communication networks, the digital feedback controller, and the anomaly detector. Such a networked control system under false-data injection attacks is depicted in Figure 1.8. The feedback controller is responsible for controlling the plant, so that it complies with performance and safety requirements. To that end, it receives the measurement signal sent by the sensors and it computes a suitable control signal that is transmitted to the actuators. The anomaly detector monitors the system to detect possible deviations from the nominal behavior and, if needed, triggers an appropriate corrective action. To monitor the system, the anomaly detector relies on an accurate model of the plant, the control action computed by the controller, and the measurements received from the sensors. All the data exchange between the plant, the controller, and anomaly detector is performed through the communication network. An adversary potentially injects false data Δu_k and Δy_k in the control command received by the actuators, $\tilde{u}_k$, and in the measurements received by the controller, $\tilde{y}_k$, respectively.

The thesis focuses on the following groups of questions related to the networked control system in Figure 1.8:

Q1 **Reference Architecture:** What core components should be considered in a reference architecture for cyber-secure and resilient networked control systems?

Q2 **Modeling Framework:** How can a malicious adversary be modeled from a control-theoretic perspective? What is the right level of modeling detail?

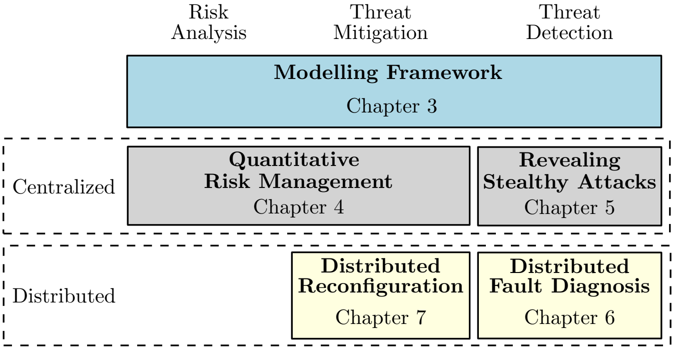


Figure 1.9: Summary of the thesis' contributions aligned with the three main actions to increase resilience: risk analysis, threat mitigation, and threat detection.

Q3 **Cyber Security Metrics:** What metrics could be used to assess and compare threats? How can they be used to prioritize attack scenarios and assess the effectiveness of defensive actions?

Q4 **Defensive Actions:** What tools and actions can be devised to increase the cyber security and resilience of control systems? Can such methods be implemented and used in real-time?

These questions are tackled throughout this thesis, contributing towards a framework to analyze, identify, and evaluate the consequences of vulnerabilities in control systems, as well as to propose and devise effective protection schemes.

1.4 Thesis Outline and Contributions

This thesis is the compilation of results presented or submitted to peer-reviewed scientific venues. The contributions are illustrated in Figure 1.9 and summarized as follows.

In Chapter 2, the existing frameworks for fault-tolerant control and IT security are revisited, which tackles **Q1** by considering the literature. Based on these frameworks, a common defense methodology to ensure security and resilience is identified. It builds upon three main functionalities: risk analysis, threat mitigation, and threat detection. The contributions of each chapter are aligned with them, as depicted in Figure 1.9.

Chapter 3 addresses **Q2** by establishing a modeling framework to capture the essence of attack scenarios with resourceful and knowledgeable adversaries that have the specific goal of covertly disrupting the physical system. The models are used to describe and analyze several attack scenarios.

Chapter 4 and Chapter 5 build upon the architecture proposed in Chapter 3. Risk analysis is the main focus of Chapter 4, where cyber security metrics are proposed, answering **Q3**. The metrics can be used to decide which assets to protect in order to mitigate threats and, thus, increase security. Chapter 5 considers a particular type of covert attack that has been identified as a high-impact attack in Chapter 4. Several schemes to reveal and detect such threats are proposed addressing **Q4**. The results and tools developed in these two chapters are envisioned to be used offline in a centralized manner.

In Chapter 6 and Chapter 7, we tackle **Q4** by devising distributed tools that may be used in real-time to improve resilience. Chapter 6 deals with distributed methods to detect threats, while Chapter 7 proposes a distributed scheme to remove defective devices and add new components while minimizing the loss of performance.

In the following, we provide more details regarding the contents of each chapter, and list the collection of papers they are based on. The order of the authors' names reflects the work load of writing the publications, where the first and second authors are the main scientific contributors for the results.

Chapter 2: Background

The chapter begins with a brief background on networked control systems, followed by a brief survey of fault-tolerant control and IT security frameworks. A short discussion of the differences and similarities between fault-tolerant control and resilient control is also presented, followed by a summary of recent work on cyber-secure and resilient control systems. The chapter concludes with a description of the experimental setups used in the thesis.

Chapter 3: A Modeling Framework for Constrained Malicious Adversaries

In this chapter, we consider a typical networked control architecture under both cyber and physical attacks. First, a generic model for malicious adversaries is discussed, where the adversary's intent is to disrupt the system behavior while remaining undetected. The adversary is constrained in terms of the available model knowledge, disclosure, and disruption capabilities. An attack-scenario space is introduced, with dimensions corresponding to these resources, in which several attack scenarios are placed and compared.

Secondly, it is shown that attack scenarios corresponding to denial-of-service, replay, zero-dynamics, and bias injection attacks on linear time-invariant control systems can be analyzed using this framework. Experimental setups are used to illustrate the attack scenarios, their consequences, and potential counter-measures.

This work is based on the following publications.

A. Teixeira, I. Shames, H. Sandberg, and K. H. Johansson. 2014. A secure control framework for resource-limited adversaries. *Automatica.* To appear.

A. Teixeira, D. Pérez, H. Sandberg, and K. H. Johansson. 2012. Attack models and scenarios for networked control systems. In Proceedings of the *1st International Conference on High Confidence Networked Systems, CPSWeek.*

A. Teixeira, G. Dán, H. Sandberg, and K. H. Johansson. 2011. Cyber security study of a scada energy management system: stealthy deception attacks on the state estimator. In Proceedings of the *18th IFAC World Congress.*

Chapter 4: Cyber Security Metrics for Networked Control Systems

Using the modelling framework outlined in Chapter 3, in Chapter 4 we address cyber security of networked control systems under the perspective of risk management. The notion of risk is defined in terms of a threat's scenario, impact, and likelihood. Emphasis is given to the assessment and treatment of risk. In particular, quantitative tools to analyse the risk of threats of static and dynamic systems are presented. First, we propose a security metric to quantify the likelihood of false-data injection threats on static electric power system. Secondly, we consider dynamic systems and propose security metrics to analyse both the likelihood and impact of threats.

The proposed security metrics aim at quantifying the risk of attack scenarios for the present configuration and model of the system. As such, these methods are not executed based on real-time data. The outcome from the security metrics may be used for risk mitigation, which is also discussed and illustrated on static and dynamics systems.

This work is based on the following publications.

A. Teixeira, K. C. Sou, H. Sandberg, and K. H. Johansson. 2014. Secure control systems: a quantitative risk management approach. *IEEE Control System Magazine.* To appear.

A. Teixeira, K. C. Sou, H. Sandberg, and K. H. Johansson. 2013. Quantifying cyber-security for networked control systems. In Danielle C. Tarraf, editor, *Control of Cyber-Physical Systems*, number 449 in Lecture Notes in Control and Information Sciences, pages 123–142. Springer International Publishing.

A. Teixeira, S. Amin, H. Sandberg, K. H. Johansson, and S. S. Sastry. 2010. Cyber security analysis of state estimators in electric power systems. In Proceedings of the *49th IEEE Conference on Decision and Control.*

H. Sandberg, A. Teixeira, and K. H. Johansson. 2010. On security indices for state estimators in power networks. In Proceedings of the *First Workshop on Secure Control Systems, CPSWeek.*

Chapter 5: Revealing Stealthy Attacks in Networked Control Systems

In Chapter 5, the problem of revealing stealthy data-injection attacks on networked control systems is addressed. In particular, we consider the scenario where the adversary performs zero-dynamics attacks on the system. First, we characterize and analyze the stealthiness properties of these attacks for linear time-invariant systems. Then, we tackle the problem of detecting such attacks by modifying the system's structure. Our results provide necessary and sufficient conditions that the modifications should satisfy in order to detect the attack. The results and proposed detection methods are illustrated through numerical examples.

This work is based on the following paper.

> A. Teixeira, I. Shames, H. Sandberg, and K. H. Johansson. 2012. Revealing stealthy attacks in control systems. In Proceedings of the *50th Annual Allerton Conference on Communication, Control, and Computing.*

Chapter 6: Distributed Fault Detection and Isolation in Networked Systems

The ability to maintain state awareness in the face of unexpected and unmodeled errors and threats is a defining feature of a resilient control system. Therefore, Chapter 6 considers physical and cyber threats on networked systems and distributed fault detection and isolation (FDI) schemes. The networked system is composed of interconnected second-order linear time-invariant systems. The subsystems are represented by nodes in a graph, while the edges correspond to the interconnections between subsystems.

Considering threats that may occur on the nodes or edges, we propose a distributed scheme based on unknown input observers (UIO) to jointly detect and isolate these threats. It is proved that, for these networked systems, one can construct a bank of UIO and use them to detect and isolate threats on nodes and edges through a distributed implementation. Moreover, the importance of certain network measurements is shown by providing infeasibility results with respect to available measurements and threats under consideration.

As our second contribution, we analyze the behavior of the proposed scheme under model uncertainties caused by the addition or removal of edges. We propose a novel distributed FDI scheme based on local models and measurements that is resilient to changes outside of the local subsystem and achieves fault detection, as well as fault isolation.

Our third contribution addresses the complexity reduction of the distributed FDI method by characterizing the minimum amount of model information and measurements needed to achieve FDI and by reducing the number of monitoring nodes. The proposed methods can be fused to design a scalable and resilient distributed FDI architecture that achieves local FDI despite unknown changes outside the local subsystem. The proposed approach is illustrated by numerical experiments on the IEEE 118-bus power network benchmark.

This work is based on the following publications.

A. Teixeira, I. Shames, H. Sandberg, and K. H. Johansson. 2014. Distributed fault detection and isolation resilient to network model uncertainties. *IEEE Transactions on Cybernetics.* To appear.

I. Shames, A. Teixeira, H. Sandberg, and K. H. Johansson. 2012. Distributed fault detection and isolation with imprecise network models. In Proceedings of the *American Control Conference.*

I. Shames, A. M. H. Teixeira, H. Sandberg, and K. H. Johansson. 2011. Distributed fault detection for interconnected second-order systems. *Automatica*, 47(12):2757–2764.

I. Shames, A. M. H. Teixeira, H. Sandberg, and K. H. Johansson. 2010. Distributed Fault Detection for Interconnected Second-Order Systems with Applications to Power Networks. In Proceedings of the *First Workshop on Secure Control Systems, CPSWeek.*

A. Teixeira, H. Sandberg, and K. H. Johansson. 2010. Networked control systems under cyber attacks with applications to power networks. In Proceedings of the *American Control Conference.*

Chapter 7: Distributed Reconfiguration in Networked Control Systems

In this chapter, we address the problem of distributed reconfiguration of networked control systems under the faulty sensors and actuators. In particular, we consider systems with redundant sensors and actuators cooperating to recover from the faults. Reconfiguration is performed while minimizing quadratic estimation and control costs. A model-matching condition is imposed on the reconfiguration scheme, in order to maintain the nominal closed-loop behavior. It is shown that the reconfiguration and its underlying computation can be distributed. Stability of the closed-loop system under the distributed reconfiguration scheme is analyzed. The approach is illustrated in a numerical example.

This work is based on the following publication.

A. Teixeira, J. Araújo, H. Sandberg, and K. H. Johansson. 2013. Distributed actuator reconfiguration in networked control systems. In Proceedings of the *4th IFAC Workshop on Distributed Estimation and Control in Networked Systems (NecSys).*

Chapter 8: Conclusions and Future Work

A summary of the thesis contributions is given and future research directions are discussed.

Other contributions

The following publications by the author had a significant influence on some of the contributions, but are not covered in the thesis.

E. Ghadimi, A. Teixeira, I. Shames, and M. Johansson. 2014. Optimal parameter selection for the alternating direction method of multipliers (ADMM): quadratic problems. *IEEE Transactions on Automatic Control.* To appear.

E. Ghadimi, A. Teixeira, M. Rabbat, and M. Johansson. 2014. The ADMM algorithm for distributed averaging: convergence rates and optimal parameter selection. In Proceedings of the *48th Asilomar Conference on Signals, Systems, and Computers*, Pacific Grove, CA, USA. To appear.

M. Juelsgaard, A. Teixeira, M. Johansson, R. Wisniewski, and J. D. Bendtsen. 2014. Distributed coordination of household electricity consumption. In Proceedings of the *IEEE Multi-conference on Systems and Control*, Antibes, France.

J. Araújo, A. Teixeira, E. Henriksson, and K. H. Johansson. 2014. A downsampled controller to reduce network usage with guaranteed closed-loop performance. In Proceedings of the *53rd IEEE Conference on Decision and Control.* To appear.

A. Teixeira, G. Dán, H. Sandberg, R. Berthier, R.B. Bobba, and A. Valdes. 2014. Security of smart distribution grids: Data integrity attacks on integrated volt/VAR control and countermeasures. In Proceedings of the *American Control Conference.*

F. Farokhi, A.M.H. Teixeira, and C. Langbort. 2014. Gaussian cheap talk game with quadratic cost functions: When herding between strategic senders is a virtue. In Proceedings of the *American Control Conference.*

A. Teixeira, E. Ghadimi, I. Shames, H. Sandberg, and M. Johansson. 2013. Optimal scaling of the ADMM algorithm for distributed quadratic programming. In Proceedings of the *52nd IEEE Conference on Decision and Control.*

I. Shames, A.M.H. Teixeira, H. Sandberg, and K.H. Johansson. 2012. Fault detection and mitigation in Kirchhoff networks. *IEEE Signal Processing Letters*, 19(11):749–752.

I. Shames, A. Teixeira, H. Sandberg, and K. H. Johansson. 2012. Agents misbehaving in a network: a vice or a virtue? *IEEE Network Magazine*, 26 (3):35–40.

E. Ghadimi, A. Teixeira, I. Shames, and M. Johansson. 2012. On the optimal step-size selection for the alternating direction method of multipliers. In Proceedings of the *3rd IFAC Workshop on Distributed Estimation and Control in Networked Systems, NecSys.*

A. Teixeira, H. Sandberg, G. Dán, and K. H. Johansson. 2012. Optimal power flow: closing the loop over corrupted data. In Proceedings of the *American Control Conference.*

J. Anderson, A. Teixeira, H. Sandberg, and A. Papachristodoulou. 2011. Dynamical system decomposition using dissipation inequalities. In Proceedings of the *50th IEEE Conference on Decision and Control and European Control Conference.*

I. Shames, A. Teixeira, H. Sandberg, and K. H. Johansson. 2010a. Distributed leader selection without direct inter-agent communication. In Proceedings of the *2nd IFAC Workshop on Distributed Estimation and Control of Networked Systems, NecSys.*

Chapter 2

Background

Resilience is the ability to maintain acceptable levels of operation in the presence of abnormal conditions. Robust and fault-tolerant frameworks have been designed to ensure resilience with respect to disturbances and faults. However, fault-tolerance is not enough to design resilient systems: the ability to secure the system against malicious adversaries is also required. Although there is a vast literature regarding security of IT systems, the same cannot be said for control systems.

In this chapter, we revisit some of the main concepts and tools pertaining resilient networked control systems that are used throughout the thesis. First, we present a brief summary of recent developments in the area of networked control systems, which are described and modeled in more detail in Chapter 3. By highlighting potential benefits and tackling many problems of using communication networks for control, the field of networked control systems has contributed to the pervasive use of IT infrastructures in safety-critical control systems. This trend leads to challenges regarding cyber security and resilience.

Second, we proceed with an overview of the classical frameworks for fault-tolerant control and IT security, followed by a comparison of fault-tolerant control and resilient control approaches. A succinct review of recent related work on cyber-secure control is also provided, to serve as non-exhaustive list of problems that have been tackled in the emerging and dynamic field of cyber-secure and resilient networked control systems. The chapter concludes with a brief description of the main applications and experimental setups considered in the thesis.

2.1 Networked Control Systems

The technological developments in computer and communication technologies triggered several paradigm shifts in control systems over the last decades. Until the 1960s, feedback controllers were mostly comprised of mechanical or analog electronic devices that exchanged analog measurements and control signals with the plant through dedicated wired media (Åström and Kumar, 2014). At the time, many control challenges dealt with stability and regulation problems. The digital revolution was already ongoing, and it soon reached a level of maturity that led to the use of digital computers and communication networks in many control applications. The use of digital technologies enabled the integration of multiple sensors and actuators, which communicated with the controller through shared wired media. This new paradigm raised several novel challenges, such as digital controller design, data sampling, and state estimation, which were addressed by the modern control theory. During the 1970s, digital controllers and communication infrastructures were developed for spatially-distributed systems (Samad *et al.*, 2007), which are now an integral part of SCADA systems. Initially, these systems used proprietary hardware, software, and wired communication technologies, making them closed to external networks and hard to interface with solutions from other vendors. Therefore, given the natural "security through obscurity" of these systems, cyber security was not a main concern (Samad *et al.*, 2007).

The further technological advances since the 1970s prompted a pervasive use of IT infrastructures in many engineered systems. Communication technologies were be standardized (Galloway and Hancke, 2013), leading to the proliferation of protocols such as FieldBus and CAN, commonly used in SCADA and automotive systems, respectively. "Security through obscurity" became outdated, as details of communication protocols became openly available. In parallel, wireless technologies, such as cellular communications, were under active development in the 1970s (Åström and Kumar, 2014). Devices with wireless communication capabilities are suitable for operating in remote locations, given their reduced installation cost compared to wired solutions. Therefore, wireless devices became an integral component of SCADA solutions for large-scale spatially-distributed systems, such as electric power networks. On the other hand, wireless communications are naturally more vulnerable to external adversaries than wired technologies, since the communication medium is easily accessible.

These recent technological developments led to two main research directions within the controls community (Baillieul and Antsaklis, 2007), which are revisited below. The first deals with the effects of unreliable communication technologies in systems controlled over communication networks, while the second leverages on communication networks to distributedly control and monitor large-scale systems. Later, we give an overview also on a third research direction that is currently emerging, namely, to address the increased exposure to cyber threats that stems from the use of pervasive and open IT infrastructures.

2.1.1 Control over Communication Networks

Digital controllers and digital communication networks, through which measurements and control signals are transmitted, have been present in industrial systems for several decades (Åström and Wittenmark, 1997). Initially, the digital devices were connected through reliable wired communication networks, with few or no data losses (Samad *et al.*, 2007). Due to the high wiring costs, the communication medium was shared between all the devices in the network, which caused delays in the data exchange. As such, the main concern until the early 1990s was the effect of varying delays on the control system performance (Richard, 2003).

As the computational and communication hardware cost is reduced, wireless devices with low-cost computational capabilities become an appealing choice for spatially-distributed control systems. However, wireless communication networks have characteristics and inherent limitations that may hinder the control performance. The design of control systems have recently addressed several of these issues, for instance, packet losses (Gupta *et al.*, 2007), limited data-rate (Ishii and Francis, 2002), and out-of-order packets (Bar-Shalom, 2002). However, approaches focusing solely on controller design may prove insufficient, when the time-scales of control systems and communication networks become closer. In such cases, the inter-play between the control system's sampling time and the communication networks parameters becomes more significant and cannot be neglected. Different approaches have been put forward to tackle this challenge, such as the use of event-triggered sampling (Wang and Lemmon, 2011), co-design of controller and communication network (Demirel *et al.*, 2014), and wireless medium access mechanisms (Ramesh *et al.*, 2013), to name a few examples.

2.1.2 Control of Networked Systems

The challenge of controlling large-scale interconnected systems has been addressed since the 1970s, such as the hierarchical and decentralized control frameworks (Siljak, 1991; Lunze, 1992). These frameworks considered spatially distributed physical systems with a sparse structure, e.g., electric power networks. A typical approach is to decompose the global system into a set of smaller interconnected systems, for which local controllers are designed (Siljak, 1991). Apart from decomposing the system, one of the main challenges of decentralized control is to design the local controllers so that the stability and performance of the overall system are guaranteed.

The use of wireless communication networks in control systems led to new possibilities and problems. By using communication networks, the local controllers became able to communicate and exchange information with each other, triggering a shift towards the distributed control framework depicted in Figure 2.1. Some of the challenges have been addressed, such as the design of distributed controllers (Bamieh *et al.*, 2002; Langbort *et al.*, 2004), distributed state estimation (Khan and Moura, 2008), and distributed fault detection (Ferrari *et al.*, 2009), among others.

In addition to the challenges from the decentralized control, new opportunities came to light with the distributed control approach. Once physically-decoupled systems become coupled through controllers and communication networks, the structure of the network plays an important role in the behavior of the global system. Such observation contributed to a large body of research with direct application to the behavior of complex networks (Barrat *et al.*, 2008), motion of animal groups (Nabet *et al.*, 2009), and multi-agent systems and cooperative robotics (Olfati-

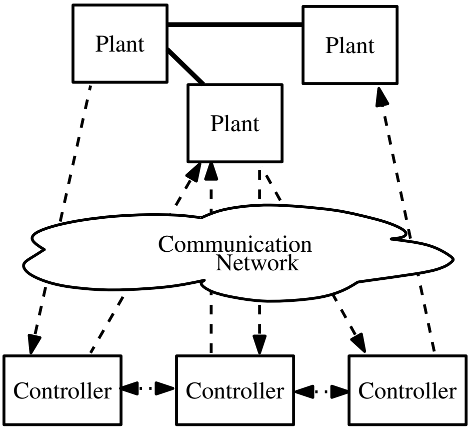


Figure 2.1: Schematic of a distributed control architecture.

Saber *et al.*, 2007; Qin *et al.*, 2012), among others. Chapter 6 relates to the latter, by developing schemes to monitor large-scale multi-agent systems in the presence of faults. Likewise, the topic of multi-agent systems is also considered in Chapter 7, where networks of intelligent sensors and actuators cooperate to recover from faults.

2.1.3 Cyber Security in Networked Control Systems

The recent developments in control over communication networks and control of networked systems may be considered as initial steps towards future systems, where cyber and physical components are tightly coupled and intertwined. A particular example is the Internet-of-Things vision (Atzori *et al.*, 2010), where multiple heterogeneous devices are able to communicate and interact with each other to achieve common goals. This vision builds on the maturity of wireless technologies and embedded computational hardware platforms. By embedding low-cost hardware in sensors, actuators, and other devices in the physical environment, they can be used to take automatic decisions based on information exchanged locally through communication networks.

However, as illustrated in Figure 2.2, each communication link and device with communication capabilities may be vulnerable to cyber attacks from malicious and knowledgeable adversaries. Therefore, the use of IT platforms increases the exposure of networked control systems to vulnerabilities and cyber threats, which leads to several challenges regarding cyber security and resilience. In the following, we review some of the existing work in this area. In particular, Section 2.2 describes fault-tolerant control. Similarly, Section 2.3 discusses the IT security framework to handle cyber threats in traditional IT systems. Their shortcomings are briefly discussed in Section 2.4, where these methodologies are integrated together as a possible framework to design cyber-secure and resilient networked control systems.

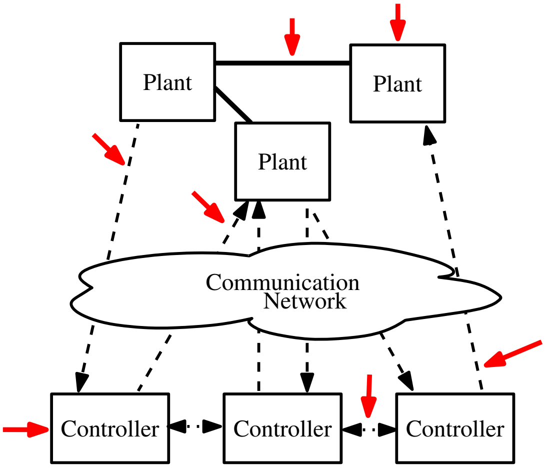


Figure 2.2: Schematic of a distributed control architecture under cyber and physical threats.

2.2 Fault-Tolerant Control Systems

Performance, reliability, and safety are essential properties of control systems, specially in safety-critical applications such as aircrafts, automotive industry, and industrial robotics. These systems have high hardware redundancy to ensure a reliable operation, possessing sets of redundant actuators and sensors. Managing the redundancy of the system is crucial to achieve safety and reliability. In the 1970s, the proliferation of digital computers reached the aircraft industry, leading the way for fly-by-wire systems (Fly, 1973). The use of digital computers in aircrafts also enabled the design of automatic systems to detect hardware failures (Willsky, 1976) and to efficiently manage redundancy and reconfigure the system (Megna and Szalai, 1977). The detection and reconfiguration mechanisms are core components of the fault-tolerant control architecture depicted in Figure 2.3 (Zhang and Jiang, 2008). This fault-tolerant control architecture will be used as a reference architecture throughout the thesis. Fault detection and isolation methods for large-scale dynamic models are studied in Chapter 6, while a distributed reconfiguration mechanism is outlined in Chapter 7.

The problem of fault-tolerant control has been extensively addressed since the 1970s, see Patton (1997); Zhang and Jiang (2008) and references therein. The following subsections provide a general overview of model-based fault diagnosis methods (Chen and Patton, 1999; Ding, 2008; Hwang *et al.*, 2010) and fault-tolerant control approaches (Zhou *et al.*, 1996; Zhang and Jiang, 2008).

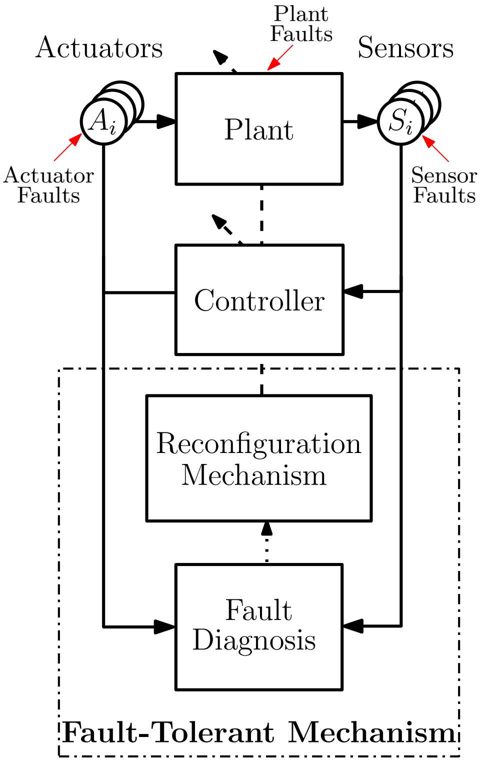


Figure 2.3: Fault-tolerant control architecture. The fault diagnosis component monitors the system for faults. The detection of faults triggers actions from the reconfiguration mechanism. Using the fault diagnosis information (dotted line), the reconfiguration mechanism modifies the controller and the plant's structure (dashed lines) to maintain adequate levels of performance.

2.2.1 Model-Based Fault Detection and Isolation

The objective of fault detection is to assess whether the system is in nominal behavior (no faults), or in an abnormal behavior (with faults). In model-based fault detection, the nominal behavior of the system can be predicted based on plant models and inputs. The basic principle in model-based fault detection is to compare the predicted and real system trajectories, obtaining the so-called residue, as illustrated in Figure 2.4. The system is declared faulty if there is a significant mismatch indicated by the residue signal. Therefore, one important issue in fault detection is the residue evaluation (Hwang *et al.*, 2010). The objective of this evaluation is to decide whether or not a fault is present, for a given residue signal. In deterministic systems, residue evaluation may be performed by comparing the norm of the residue signal against a threshold chosen to ensure robustness to uncertainties (Ding, 2008). In

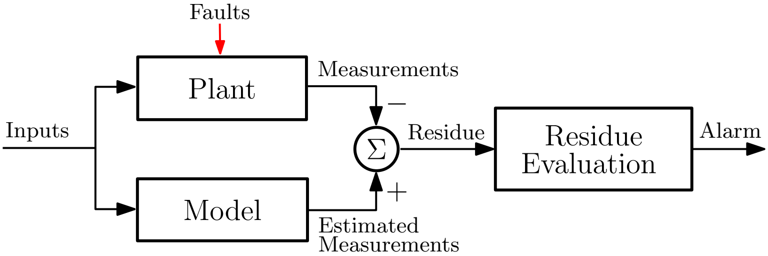


Figure 2.4: Model-based fault detection scheme. The plant model and inputs are used to estimate the output signal. The mismatch between the actual output and its estimate is evaluated to detect faults.

stochastic systems, the statistical model of the residue signal can be used to design optimal evaluation schemes in the form of hypothesis tests, for instance the generalized likelihood ratio test, sequential probability ratio test, and CUSUM (Basseville and Nikiforov, 1993; Hwang *et al.*, 2010).

Residue generation

Hwang *et al.* (2010) give an overview of the several approaches to model-based fault detection, isolation, and recovery. Regarding fault detection, one of the main problem is the computation of the residue signal, i.e., a signal quantifying the mismatch between the real and predicted outputs. This is particularly important in the presence of measurement and process noise, unknown disturbances, and model uncertainties. A widely used class of model-based residue generation schemes is the observer-based approach (Patton and Chen, 1997). In this approach an observer is designed to estimate the state and output of the plant, which is then compared to the real plant output to generate the residue. Some examples are used next to illustrate the main concepts behind model-based fault detection and isolation.

Example 2.1
Consider the static model

$$y = Cx + Bu + Ff = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 1 \end{bmatrix} x + Bu + Ff,$$

where the matrices C, B, and F are known, y is the set of measurements, x is the unknown state, u is the known control input, and f is a possible fault. Moreover, note that the matrix C has full column rank. Consider the nominal fault-free case where one has $f = 0$. An observer-based approach to generate a residue is to estimate the state x through linear least-squares, yielding $\hat{x} = (C^\top C)^{-1}C^\top(y - Bu)$.

This estimate can then be used to generate the following residue

$$r = y - Bu - C\hat{x} = (I - C(C^\top C)^{-1}C^\top)(y - Bu).$$

Note that, in the faulty case $f \neq 0$, we have $r = (I - C(C^\top C)^{-1}C^\top)f$. This residue can detect faults f only if they do not satisfy the model, i.e. $f \notin \operatorname{Im}(C)$.

The former example illustrated an observer-based method for a linear static system. Similar approaches exist for dynamical systems as well, using for instance full-order observers (Patton and Chen, 1997) or Kalman filters (Chow and Willsky, 1984). In the presence of additional uncertainties as unknown disturbances, other techniques must be employed. Examples of such techniques include robust observers compensating the disturbance effect (Douglas and Speyer, 1995), optimization-based observers mitigating the disturbance effects while maximizing the sensitivity to faults (Chung and Speyer, 1998), and unknown input observers (UIO) that are able to completely decouple the state estimate from disturbances (Chen *et al.*, 1996). The UIO approach is depicted in the next example and will be used in Chapter 6.

Example 2.2

Consider the previous example, but with an unknown disturbance and no fault

$$y = Cx + Bu + Dd = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 1 \end{bmatrix} x + Bu + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} d,$$

where d is a scalar disturbance. To obtain a residue decoupled from d, one can pre-multiply the measurements by $P = I - D(D^\top D)^{-1}D^\top$, resulting in $w = P(y - Bu) = PCx + PDd = PCx$. If $\tilde{C} = PC$ is full-column rank, the disturbance decoupled residue can be computed as

$$\tilde{r} = w - \hat{w} = (I - \tilde{C}(\tilde{C}^\top \tilde{C})^{-1}\tilde{C}^\top)P(y - Bu) = (I - \tilde{C}(\tilde{C}^\top \tilde{C})^{-1}\tilde{C}^\top)PCx.$$

The UIO is a dynamic equivalent to this example.

In addition to fault detection, it is useful to locate the faulty component in the system, so called fault isolation (Ding, 2008; Hwang *et al.*, 2010). Fault isolation is usually a harder problem than fault detection and may require additional model knowledge. Since for fault isolation one needs to distinguish between different faults, several models are required, as indicated for two faults in Figure 2.5.

A common approach is to constrain the design of the residue generator, such that the residues have a certain structure facilitating isolation. Possible methods include the Beard-Jones filter, designed so that each fault excites the residue in a given direction, or the structured residues approach, where a bank of observers is jointly designed to ensure isolation. Two particular cases of the structured residues

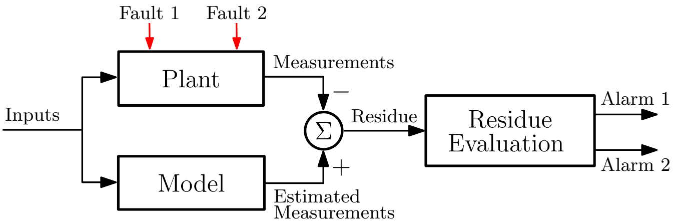


Figure 2.5: Model-based fault isolation scheme.

approach are the dedicated observer scheme where each observer is sensitive to only one fault, and the generalized observer scheme where each observer is sensitive to all but one fault. The following example illustrates the latter method.

Example 2.3
Consider the static model in Example 2.1 with three faults

$$y = Cx + Bu + Ff + = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 1 \end{bmatrix} x + Bu + \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} f_1 \\ f_2 \\ f_3 \end{bmatrix},$$

where f_1, f_2, and f_3 are scalar faults. Assume only a single fault occurs at a time. The generalized observer scheme is used to isolate the faults where three residues are designed, each sensitive to all but one fault. Denoting r_1 as the residue insensitive to f_1, f_1 could be treated as a disturbance and r_1 generated using the approach in Example 2.2. Repeating the procedure for r_2 and r_3, a bank of residues is obtained with the following sensitivity table

	f_1	f_2	f_3
r_1	0	1	1
r_2	1	0	1
r_3	1	1	0

where 1 (0) denotes that the residue is sensitive to (decoupled from) a given fault. The faults can thus be isolated once they are detected. The generalized observer scheme will be used in Chapter 6 for fault isolation in large-scale dynamical systems.

2.2.2 Fault-Tolerant Control

The different methodologies to achieve fault-tolerant control can be broadly classified as being either passive or active (Zhang and Jiang, 2008). The set of passive methods do not use real-time information regarding the fault. Instead, these methods restrict their attention to a set of faults that can be characterized and modeled offline. Using these models, the controller is designed so that it mitigates any fault in the considered set. On the other hand, active approaches use real-time fault information to react to faults, such as in the methods described previously. The real-time information regarding the fault can be used to reconfigure the control system in a suitable way.

In the following, we describe some of the concepts behind passive and active fault tolerant control schemes. First, the robust control problem is described and mapped onto the class of passive approaches. Later, reconfigurable control schemes are discussed as part of the active approaches.

Robust control approach

In the classical control design problem, the main objective is to stabilize the system while attenuating disturbances and noise, under the assumption that the plant and disturbance models are known (Zhou *et al.*, 1996). In practice, there are always discrepancies between the models and the actual system. For this reason, the design of control systems able to handle model uncertainty and unmodeled disturbances has long been a concern. It has been formulated as the robust control design problem. By modelling faults as unmodeled disturbances or model uncertainty, the design of robust controllers to mitigate faults is part of the passive fault-tolerant control approaches (Zhang and Jiang, 2008).

Robust control theory has contributed with several frameworks to handle model uncertainties and disturbances, see Zhou *et al.* (1996). In all these frameworks, the robust controller is designed to withstand disturbances and uncertainty belonging to a given set of interest. For instance, in the $\mathcal{H}_\infty$ robust control design, introduced by Zames (1981), the controller aims at minimizing the system's output energy with respect to the worst-case disturbance with bounded energy.

Example 2.4

Consider the static model in the previous examples with one fault f and control input u

$$y = Cx + Bu + Ff$$

where x is unknown. To measure the system's performance, we consider the quadratic cost function

$$J(x, u, f) = y^\top y.$$

The objective of the robust control design is to compute u so that the cost remains small under the influence of the worst-case bounded fault. In particular, the worst-case bounded fault aims at maximizing the cost $J(x, u, f)$ while satisfying the constraint $f^\top f \leq 1$. Formally, this can be formulated as a game-theoretic problem, where the control u and the fault f compete to, respectively, minimize and maximize the cost:

$$\begin{aligned} \underset{u}{\text{minimize}} \quad \underset{f}{\text{maximize}} \quad & J(x,u,f) \\ \text{subject to} \quad & f^\top f \leq 1. \end{aligned}$$

The dynamic version of this robust control design problem is known as the $\mathcal{H}_\infty$ control problem (Basar and Bernhard, 1995).

There exist drawbacks in using robust control techniques to mitigate faults. One of particular interest is that the performance of the robustly controlled system may be poor under nominal conditions, i.e., without faults. This drawback motivates the use of active fault-tolerant control schemes.

Reconfigurable control approach

Since the 1970s, much research has been conducted in active fault-tolerant control schemes (Maciejowski, 1997; Lunze and Richter, 2008; Zhang and Jiang, 2008). The rational behind the active approaches is to modify the nominal control system only when faults are present, as to ensure good performance under nominal conditions. Several active fault-tolerant schemes are available in the literature, e.g., adaptive controllers (Tao *et al.*, 2002), switching controllers (Yang *et al.*, 2009), and online controller reconfiguration (Lunze and Steffen, 2006).

Reconfigurable control proposes methods to reconfigure the control system after a fault has been detected and diagnosed, while avoiding a complete redesign. The overall objective of control reconfiguration is to minimize the loss in performance inflicted by the fault. This goal may be achieved, for instance, by ensuring the system's stability, maintaining a similar closed-loop behavior as before the fault (also known as model-matching), or achieving the same equilibrium point. Model-matching reconfiguration, in particular, has been the focus of much research in this area (Lunze and Richter, 2008). Chapter 7 follows this direction in a distributed setting.

Example 2.5
Consider a scalar dynamical system with 3 actuators

$$\dot{x}(t) = x(t) + \sum_{i=1}^{3} u_i(t)$$

where $x(t) \in \mathbb{R}$ is the scalar state and $u_i(t) \in \mathbb{R}$ is the i-th input. Suppose the following state-feedback control policy is used to stabilize the system

$$u(t) = \begin{bmatrix} u_1(t) \\ u_2(t) \\ u_3(t) \end{bmatrix} = \begin{bmatrix} K_1 \\ K_2 \\ K_3 \end{bmatrix} x(t) = \begin{bmatrix} -0.1 \\ -0.4 \\ -1 \end{bmatrix} x(t),$$

yielding the closed-loop system

$$\dot{x}(t) = \left(1 + \sum_{i=1}^{3} K_i\right) x(t) = -0.5x(t).$$

After the complete failure of the third actuator, i.e., $u_3(t) = 0$ after the fault, the closed-loop dynamics become

$$\dot{x}(t) = \left(1 + \sum_{i=1}^{2} K_i\right) x(t) = 0.5x(t),$$

resulting in an unstable system. We use the ideas of reconfigurable control through model-matching to recover the closed-loop dynamics before the fault. Given the failure of the third actuator, the closed-loop dynamics before the fault can be recovered with any controller satisfying the model-matching constraint

$$K_1 + K_2 = -1.5.$$

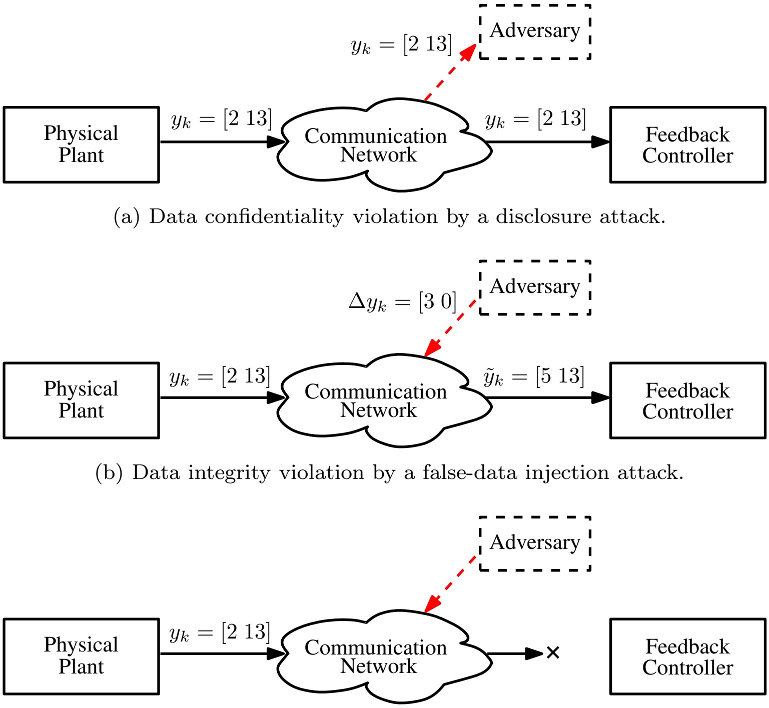
Note that this under-determined equation admits an infinite number of solutions, which indicates that the set of actuators is redundant. A possible way to obtain a unique solution is to assign a convex cost $J_i(K_i)$ to each actuator and satisfy the model-matching constraint while minimizing the sum of costs:

$$\begin{aligned} \underset{K_1, K_2}{\text{minimize}} \quad & J_1(K_1) + J_2(K_2) \\ \text{subject to} \quad & K_1 + K_2 = -1.5. \end{aligned}$$

The former optimization problem is known as the control allocation problem (Johansen and Fossen, 2013). A generalized formulation of this problem for higher-order systems is tackled in Chapter 7.

2.3 Secure IT Systems

Information is a key asset in knowledge-driven societies, which require a reliable and continuous availability of data and services. Redundant and fault-tolerant architectures are thus required to build IT systems resilient to faults and disturbances (Koren and Krishna, 2010). Additionally, IT systems must also be defended against malicious adversaries that aim at disrupting or gaining access to the information flow. Next, we revisit the main concepts in IT security.



(a) Data confidentiality violation by a disclosure attack.

(b) Data integrity violation by a false-data injection attack.

(c) Data availability violation by a denial-of-service attack.

Figure 2.6: Cyber attacks on the communication links of a networked control system.

2.3.1 Fundamental Security Properties

Three fundamental properties of information and services in IT systems are mentioned in the computer security literature (Bishop, 2002) using the acronym CIA: confidentiality, integrity, and availability. Confidentiality concerns the concealment of data, ensuring it remains known to the authorized parties alone. Integrity relates to the trustworthiness of data, meaning there is no unauthorized change to the information between the source and destination. Availability considers the timely access to information or system functionalities.

Figure 2.6 illustrates cyber attacks that violate these security properties. In all three cases, the plant is sending the measurement vector $y_k = [2\ 13]^\top$ to the controller through a communication network. This is a private message, hence only the plant and the controller should know its contents. In Figure 2.6a, the adversary is able to eavesdrop on the communication, thus getting access to the message contents. Therefore confidentiality is violated. Another scenario occurs in Figure 2.6b, where the adversary succeeds in sending the false measurement vector $\tilde{y}_k = y_k + \Delta y_k$ to the controller, as if it was the plant sending it. Here data integrity is violated.

In our final example, illustrated in Figure 2.6c, the message sent by the plant is actually blocked and does not reach the controller. Hence data availability is compromised.

The violations presented in these examples were caused by disclosure, deception, and denial-of-service attacks, respectively. Whereas in IT systems the impact of such cyber attacks remains in the cyber realm, in networked control systems they may carry dire consequences to the physical side. Instances of these attacks and their consequences on control systems are illustrated in Chapter 3. Deception attacks as in Figure 2.6b are further analyzed in Chapter 4 and Chapter 5.

The objective of IT security is to ensure that data and IT services have the three properties described in this subsection. In the next subsection, we describe a conceptual framework to achieve the latter goal.

2.3.2 IT Security Reference Architecture

One of the existing standards for security of networked IT systems is the security architecture for Open Systems Interconnection (OSI) (ITU, 1991). The standard provides a systematic framework to describe IT security requirements and characterize approaches to satisfy such requirements. In particular, the security architecture for OSI (ITU, 1991) considers three main concepts: security policy, security services, and security mechanisms. The security policy is a set of requirements and rules stating what behaviors are allowed or not in secure systems. Security services are different functionalities that may be combined to ensure a given security policy. Security mechanisms are tools and procedures designed to prevent, detect, or recover from attacks. Several security mechanisms may be used to achieve a given security service.

As an example, consider a security policy stating that confidentiality violations, as illustrated in Figure 2.6a, are not acceptable. This policy may be achieved, for instance, using the following security services: access control and authentication. Access control prevents unauthorized devices from accessing the transmitted data, using mechanisms such as access control lists. Note that the access control service relies on the authentication service, which verifies the identity of devices requesting access to the transmitted data. The authentication service may be implemented using security mechanisms such as digital signatures and encryption.

In addition to the conceptual security framework, the security standard also maps several basic security services to the different layers of the OSI reference model for communication protocols (Krutz, 2006). In fact, several approaches in the literature are aligned with the layered approach of the reference security architecture. For instance, the survey by Chen *et al.* (2009) discusses several methodologies for security of sensor networks, where security mechanisms and services for different layers were proposed. Next, we summarize the OSI model for communication networks.

The OSI reference architecture proposes a layered high-level model for communication protocols, as depicted in Figure 2.7. Each layer is defined as a set of

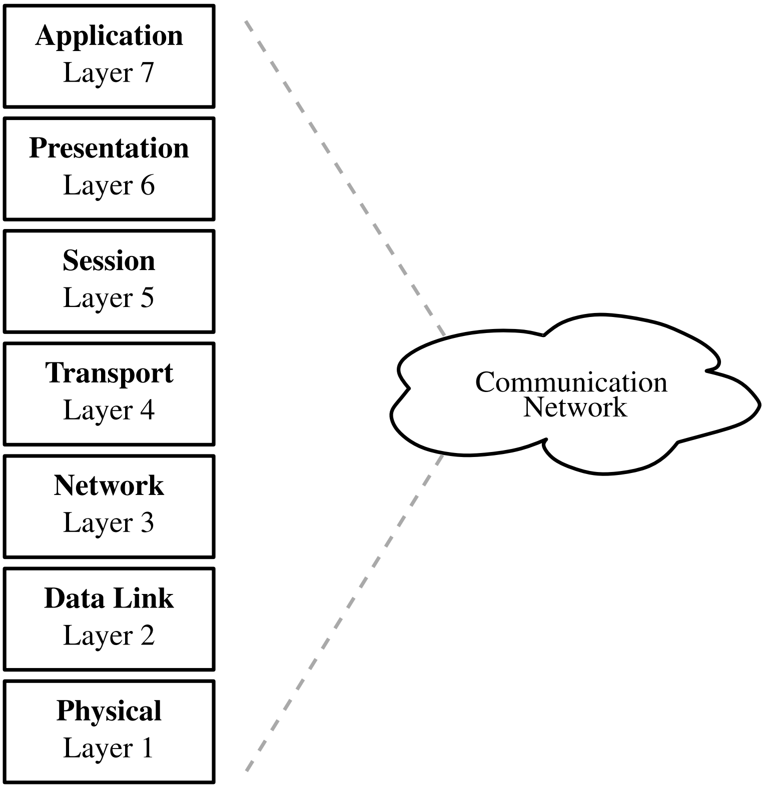


Figure 2.7: The OSI reference model for communication networks, composed of seven hierarchical layers.

well-defined functions that service the layer above and are serviced by the one below. Moreover, each layer within a particular device only interacts with the same layer of other devices. The OSI model is comprised of seven layers, out of which, according to Tanenbaum and Wetherall (2010), the most prevalent layers in practice are the physical, data link, network, transport, and application layers.

The physical layer concerns the conversion of raw digital data, such as bits, into physical signals that are propagated through a physical transmission medium. The data link layer is responsible for mediating the access of several devices to a shared physical medium, as well as ensuring an error-free flow of data-frames between devices. The network layer tackles routing and device addressing functionalities, while the transport layer manages the end-to-end connection, by ensuring that all data are carried from source to destination without errors. The application layer, the highest layer in the model, provides the users' application software with support functions and enables the use of the lower-level communication protocols. There-

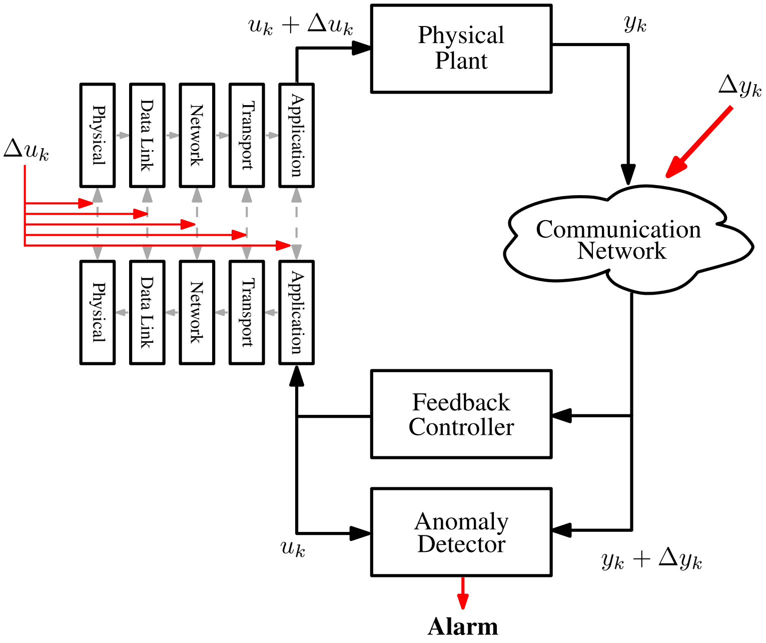


Figure 2.8: Schematic of a networked control system with a communication network that is vulnerable to adversaries. The communication network to the left is represented by the OSI reference model, where the cyber attack may occur at different layers.

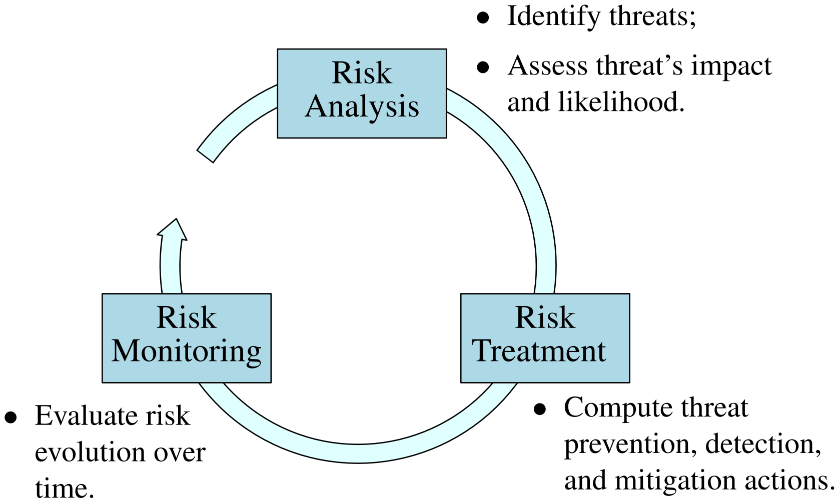
fore, digital controllers, actuators, sensors, and other devices with communication capabilities lie on top of the application layer, as illustrated in Figure 2.8.

This thesis focuses on the user application on top of the OSI reference model, namely the physical plant and the control and monitoring algorithms. Addressing security and resilience at this conceptual level provides yet another layer of defense against malicious threats.

2.3.3 Risk Management

The risk management framework (Bishop, 2002; U.S. DHS, 2011; NIST, 2012) is another common methodology to enhance a system's cyber security. The main objective of risk management is to assess and minimize the risk of threats, where the notion of risk is defined as follows (Kaplan and Garrick, 1981).

Definition 2.3.1. *Consider a given threat scenario, the corresponding impact to the system, and the likelihood of such scenario. The risk of the system is denoted as the set of triplets* $\text{Risk} \triangleq \big\{(\text{Scenario, Impact, Likelihood})\big\}$.



Risk Analysis
• Identify threats;
• Assess threat's impact and likelihood.
Risk Treatment
• Compute threat prevention, detection, and mitigation actions.
Risk Monitoring
• Evaluate risk evolution over time.

Figure 2.9: Diagram of the risk management cycle. Risk of threats is continuously minimized by iteratively performing risk analysis, risk treatment, and risk monitoring.

Since risk may vary over time, with the appearance of new threat scenarios and ageing of the system, risk must be continuously managed to ensure security. Such requirement leads to the risk management cycle depicted in Figure 2.9, which is composed of risk analysis, risk treatment, and risk monitoring.

Risk analysis identifies threats and assesses the respective likelihood and impact on the system. Threat scenarios may be identified based on historical and empirical data of cyber attacks, expert knowledge, and known vulnerabilities in the system (NIST, 2012). The report (NES, 2014) provides a good example of power system related threat scenarios identified from expert knowledge. The likelihood of a given threat depends on the components compromised by the adversary in a given attack scenario and their respective vulnerability. Quantitative methods can be used to identify the minimal set of components that need to be compromised for each attack scenario (Sommestad *et al.*, 2013; Sandberg *et al.*, 2010), while the vulnerability of each compromised components is obtained by qualitative means such as expert knowledge and historical and empirical data (Sommestad *et al.*, 2013). The potential impact of a threat may be assessed by qualitative and quantitative methods, for instance, by modeling the system and simulating the attack scenarios (Sridhar *et al.*, 2012).

The risk of different threat scenarios may be summarized in a two-dimensional risk matrix (NIST, 2012), where each dimension corresponds to the likelihood and impact of threats, respectively. Additionally, the risk of different threats may be

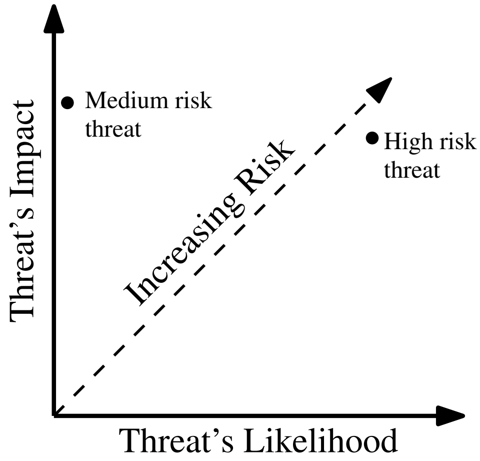


Figure 2.10: A risk matrix plot. Two threats with similar impact but different likelihoods are depicted. Threats with high impact and high likelihood yield a high risk.

compared through increasing functions of the threat's impact and likelihood. As an example, Figure 2.10 illustrates medium and high risk threats with similar impact but different likelihood.

Actions minimizing the risk of threats are determined within the risk treatment step. The different actions can be classified as prevention, detection, and mitigation. Prevention aims at decreasing the likelihood of attacks by reducing the vulnerability of the system components, for instance, by encrypting the communication channels, using firewalls, and intelligent routing algorithms (Vukovic *et al.*, 2012). Regarding the disclosure attacks considered in Figure 2.6a, encryption of the communication link corresponds to a preventive action.

Detection is an approach in which the system is continuously monitored for anomalies caused by adversary actions. Examples of detection schemes include anti-virus softwares, network traffic analysis (Garitano *et al.*, 2011), and fault detection algorithms (Ding, 2008). Such schemes are similar to the model-based fault detection approach, described in Section 2.2.1. These principles are indeed used in practice, for instance to detect abnormal data traffic using statistical models (Zhang *et al.*, 2009).

Once an anomaly or attack is detected, mitigation actions may be taken to disrupt and neutralize the attack. The attack may be neutralized by replacing the compromised components or using redundant components. In the case of the denial-of-service attack in Figure 2.6c, one could have a mitigation scheme where the data are re-sent using a different path from source to destination, thus avoiding the compromised links.

The effectiveness of the defensive actions and the evolution of risk over time is evaluated throughout the risk monitoring stage. Risk monitoring continuously assesses the known and newly discovered vulnerabilities of the system, as well as the deployment of the threat mitigation actions. For instance, in the case of deception attacks the attacker may find attack strategies that bypass the current detection mechanisms. This particular scenario is explored in Chapter 3 to Chapter 5.

2.4 Cyber-Secure and Resilient Control Systems

The design of resilient control systems can leverage the approaches from fault-tolerant systems and risk management. In fact, from a risk perspective, faults and malicious attacks can both be seen as threats with different scenarios, impact, and likelihood. Therefore, the risk management framework outlined in Section 2.3.3 may handle faults and attacks in a holistic way. Conversely, fault-tolerant control tools can be used to detect and attenuate the consequences of cyber attacks on networked control systems, since these attacks affect the physical behavior of the system similar to faults. However, there are substantial conceptual and technical differences between the fault-tolerant and resilient control frameworks that motivate the need for specific theories and methodologies to address security issues in control systems.

Cyber attacks and faults have inherently distinct characteristics, which pose different challenges. Faults are considered as physical events that affect the system behavior, where simultaneous events are assumed to be non-colluding, i.e., the events do not act in a coordinated way. On the other hand, cyber attacks may be performed over a significant number of attack points in a coordinated fashion (Teixeira *et al.*, 2011; Smith, 2011). Moreover, faults do not have an intent or objective to fulfill, as opposed to cyber attacks that do have a malicious intent. In Chapter 3, several attack scenarios exploiting these issues are discussed in detail.

To better illustrate the subtle differences between faults and attacks, we revisit a particular fault-tolerant approach. The $\mathcal{H}_\infty$ fault-tolerant control problem considers a bounded fault that aims at maximizing the cost function. In this setting, the unique aim of the fault is to disrupt the system performance, while no other goals are considered, e.g. stealthiness. A substantially different approach is taken in Example 2.6, where the adversary aims at maximizing the cost function while remaining undetected.

Example 2.6

Consider the static model in the previous examples with one fault f and control input u

$$y = Cx + Bu + Ff,$$

where x is unknown. To measure the system's performance, we consider the cost function

$$J(x, u, f) = y^\top y.$$

In the robust control design problem described in Example 2.4, the fault is seen as a malicious adversary that aims at maximizing the cost $J(x,u,f)$. The fault is also constrained in magnitude by $f^\top f \leq 1$, which is required for the problem to be well-posed and admit finite solution. However, other constraints typical in adversarial setting, such as remaining undetected, are not considered. Next we describe a scenario where the adversary is also constrained to remain covert.

Recall the anomaly detector described in Example 2.1

$$r = y - C\hat{x} = \underbrace{(I - C(C^\top C)^{-1}C^\top)}_{=C_e}(y - Bu),$$

where an anomaly is detected if the residue's norm exceeds a certain threshold: $\|r\| > \delta$.

Consider the attack scenario where the adversary intends to maximize the cost $J(x,u,f)$, while remaining undetected by having $\|r\| \leq \delta$. Formally, the robust control design problem under such scenario can be formulated as a game-theoretic problem:

$$\begin{array}{lll} \underset{u}{\text{minimize}} & \underset{f}{\text{maximize}} & J(x,u,f) \\ & \text{subject to} & f^\top F^\top C_e^\top C_e F f \leq \delta. \end{array}$$

While the similarity to the classical robust $\mathcal{H}_\infty$ control problem is clear, there are substantial differences regarding, for instance, conditions for the game to admit a finite-valued solution. The dynamic version of this attack scenario is tackled in Chapter 4.

The distinct characteristics of faults and attacks lead to quite different approaches. Increased resilience may be achieved through mainly three actions: prevention, detection, and mitigation (Bishop, 2002; Isermann, 2006). These actions need to be tailored to the specific properties of faults and attacks to efficiently and effectively ensure resiliency. For instance, prevention, detection, and mitigation of faults may be achieved by maintenance, on-line monitoring, and timely repair of the physical components of the system, respectively. On the other hand, preventing, detecting, and mitigating cyber attacks on control systems must use mechanisms that consider both the cyber and physical realms, such as encryption and improved control algorithms (Pang and Liu, 2012). Furthermore, ensuring security may involve addressing large number of threats, thus requiring attack impact analysis and the use of risk assessment methods (Sridhar *et al.*, 2012). Several of these issues are presented in the thesis and have also been addressed in recent work on secure control systems.

2.4.1 Related Work

Next, we provide a brief review of recent work on cyber-secure and resilient control systems. While relevant related work is also discussed in each chapter of the thesis, this section covers other interesting work that may not be directly related to the different chapters.

An overview of existing cyber threats and vulnerabilities in networked control systems is presented in Cárdenas *et al.* (2008b,a) and Cárdenas *et al.* (2009). Particularly, realistic and rational adversary models are mentioned as one of the key items in security for control systems. To grasp the relevance of such features, recall that cyber and physical attacks may affect the plant directly. These attacks can be modeled as faults. In the framework of security, however, such attacks are endowed with intelligence and intent, as opposed to faults. Therefore, these attacks may exploit vulnerabilities existing in the traditional fault detection mechanisms and remain undetected, as illustrated in Example 2.1. In fact, Amin *et al.* (2010) reported experimental stealthy data deception attacks on water irrigation canals controlled by SCADA systems. Smith (2011) characterized stealthy attack policies for scenarios where the attacker is able to perform disclosure and deception attacks on all the sensors, illustrating it on the same water irrigation system. A similar approach is followed in Pasqualetti *et al.* (2013), where the stealthy attack policies are characterized from networked systems modeled by differential-algebraic equations. Additionally, centralized and distributed detection schemes targeting detectable attacks are proposed.

Instances of stealthy false-data injection attacks have recently been studied for systems with static models. For example, in the case of electric power networks, an adversary with perfect model knowledge has been considered in Liu *et al.* (2009). The work by Kosut *et al.* (2010, 2011) considered stealthy attacks with limited resources and proposed improved detection methods, while Sandberg *et al.* (2010) analyzed the minimum number of sensors required for stealthy attacks. A corresponding measurement security metric for studying sets of vulnerable sensors was proposed in Sandberg *et al.* (2010). The consequences of these attacks have also been analyzed in Xie *et al.* (2010) and Teixeira *et al.* (2012a). In particular, Teixeira *et al.* (2011) analyzed attack policies with limited model knowledge and performed experiments on a power system control software, showing that such attacks are stealthy and can induce the erroneous belief that the system is at an unsafe state. This experiment inspired the second motivational example in Chapter 1, being described in more detail in Chapter 3.

Efficient methods to compute all stealthy attacks on power network measurements were proposed by Giani *et al.* (2013), with and without assuming that all power flows are measured. Similarly, Sou *et al.* (2013b) proposed methods based on minimum-cut algorithms to exactly compute stealthy attacks on power networks, while assuming that all power flows are measured.

The protection of power systems has also been addressed in the literature. For instance, Dán and Sandberg (2010) proposed the use of greedy algorithms to deploy secure measurements, while Kim and Poor (2011) followed a similar direction by considering the deployment of encryption and PMUs. Furthermore, Vukovic *et al.* (2012) considered the communication network topology and proposed schemes to reroute measurements such that stealthy attacks become more difficult to accomplish.

The risk management approach for power systems has also been considered. In particular, the survey Sridhar *et al.* (2012) discusses the risk management framework for several layers of power systems, namely generation, transmission, and distribution. Risk assessment for power networks is further examined in Bompard *et al.* (2009) under a game-theoretic approach.

Cyber attacks have also been addressed in the context of dynamic control systems. The work by Fawzi *et al.* (2012) considers a finite time-interval and characterizes the number of corrupted channels that cannot be detected during that interval. Mo and Sinopoli (2009) considered replay attacks on wireless networks performing state estimation, which are a particular class of deceptions attacks. They proposed a novel detection scheme tailored to this class of attacks, which was later optimally designed by (Chabukswar *et al.*, 2011). Other work analyzes denial-of-service attacks, where the optimal attack policy under finite resources is characterized (Amin *et al.*, 2009; Gupta *et al.*, 2010).

Impact of false-data injection attacks has also been considered in the literature. For linear networked control systems under false-data injection attacks, Mo and Sinopoli (2012) propose methods to approximate the reachable set of states for stealthy adversaries. The safety of Automatic Generation Control for power system under deception attacks was considered in Esfahani *et al.* (2010) and the authors showed that the cyber attacks could violate the system safety constraints.

In the context of multi-agent systems, rational attackers performing stealthy deception attacks were also considered for networks computing linear functions, where each node is modeled as a first-order system (Sundaram and Hadjicostis, 2011; Pasqualetti *et al.*, 2012). The class of stealthy deception attacks was characterized in terms of the number of compromised nodes and the network connectivity. The work by Sundaram *et al.* (2012) considered the detection and mitigation of false-data injection attacks on linear information dissemination algorithms over communication networks. A different approach is proposed inLeBlanc *et al.* (2013), where the resilience of a local consensus scheme to attacks is characterized in terms of the communication graph.

Other challenges were also considered for multi-agent systems. For instance, optimal adversary policies for data injection attacks using full model knowledge and state information were derived in Khanafer *et al.* (2012), while Zhu and Martinez (2012) tackled replay attacks on multi-agent systems, by proposing distributed control algorithms to mitigate the attacks.

Game-theoretic approaches to secure control are available in the literature. Amin *et al.* (2013) analyzed security incentives for interdependent networked control systems. A dynamic game-theoretic approach was proposed by Zhu and Başar (2012) to tackle cascading failures, by jointly considering IT security and robust control policies. Using a stochastic game-theoretic setting, Miao *et al.* (2013) proposed a controller switching policy to detect replay attacks.

Benchmark examples for security in networked control systems were described in Rieger (2010) and numerical experiments on a benchmark process plant were reported by Cárdenas *et al.* (2011). In the latter, although the adversary's objectives and a mathematical formulation for the effects of cyber attacks were given, the attack policies did not make full use of the adversary's resources. Consequently, since the worst-case attack policy was not considered, other attack scenarios with similar resources might yield more dire consequences.

2.5 Applications and Experimental Setups

The results described in the thesis are illustrated through experiments and numerical simulations on testbeds related to power systems and process control. The architecture and models of these testbeds are described in the following.

2.5.1 Power Transmission Networks

SCADA systems in power transmission networks have evolved substantially since they were introduced in the 1960s (Wu *et al.*, 2005). The early systems were mainly used for logging data. Today modern SCADA systems are enhanced by Energy Management Systems (EMS) providing system-wide monitoring and control to meet performance and reliability constraints (Balu *et al.*, 1992; Shahidehpour *et al.*, 2005).

Due to constraints of traditional technologies, only quasi-steady state dynamics are captured by current SCADA EMS. However, with the advent of new sensors such as Phasor Measurement Units (PMUs), transient behaviors of power transmission networks can be captured. This leads to the so-called Wide-Area Monitoring and Control Systems (WAMS/WAMC), providing yet another layer of control.

In the following, cyber threats to power networks are discussed and the EMS components and the WAMS system are briefly described.

Cyber threats

There are several threats in a SCADA system. In Figure 2.11 we illustrate some of these threats and the respective entry points to the SCADA EMS. The measurements sent by the RTU (A2) and the system information in the SCADA databases (A3) could be targets of disclosure attacks to gain access to confidential data, such as the power network model. A denial-of-service attack could be performed on the communication links between the RTUs and the control center (A2 and A6), resulting in loss of availability. Another attack scenario corresponds to deception attacks on the RTU data sent to the control center (A1–A3), resulting in a violation of data integrity. This scenario is further discussed in Chapter 3 to Chapter 5, where we characterize the class of stealthy deception attacks bypassing existing detection schemes, similar to the scenario illustrated in Figure 2.6b.

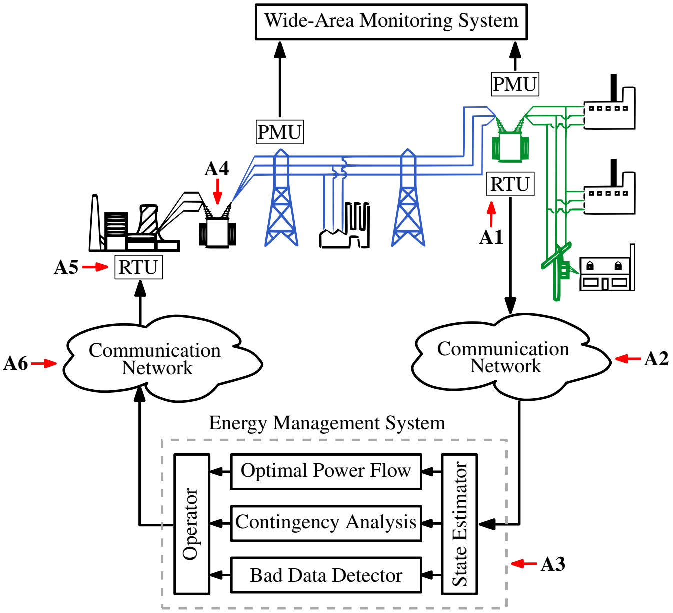


Figure 2.11: A schematic diagram of a power network with a SCADA system, a WAMS monitoring faults, and possible IT vulnerabilities.

Energy management system

Figure 2.11 illustrates some of the components in traditional SCADA EMS systems. Power networks are hybrid systems, having analog variables, such as voltages and currents, and digital variables like breaker status. System-wide measurements of these variables are taken locally at the substation level, gathered by RTUs, and transmitted to the control center through the communication network. Since not all variables are measured, the current state of the power network needs to be estimated based on the received measurements and a detailed system model. The optimal state and measurement estimates are computed by the state estimator (SE). Possible measurement errors can be handled *a posteriori* by bad data detectors (BDD).

The SE provides system observability to operators and other EMS tools, thus being an integral tool in power network operation. As shown in Figure 2.11, contingency analysis (CA) tools use the estimates to evaluate if the system meets the required reliability criteria in the presence of equipment failures. Optimal power flow (OPF) analysis based on the estimates evaluates possible improvements in performance. Based on the recommendations from the CA and OPF, the human operator chooses suitable control actions to be applied to the power network, as illustrated in Figure 2.11.

To acquaint the reader with the EMS components used in the thesis, next, we revisit standard power network models and SE and BDD algorithms available in the literature (Abur and Exposito, 2004).

Measurement model

For an electric power network with N buses, the $n = 2N - 1$ dimensional state vector x is $(\theta^\top, V^\top)^\top$, where $V = (V_1, \dots, V_N)$ is the vector of bus voltage magnitudes and $\theta = (\theta_2, \dots, \theta_N)$ vector of phase angles. This state vector is the minimal information needed to characterize the operating point of the power network. Without loss of generality, we let bus 1 be the reference bus, hence all phase-angles are taken relatively to this bus and $\theta_1 = 0$. The m-dimensional measurement vector y can be grouped into two categories: (1) y_P, the active power flow measurements P_{ij} from bus i to j and active power injection measurement P_i at bus i, and (2) y_Q, the reactive power flow measurements Q_{ij} from bus i to j, reactive power injection measurement Q_i and V_i voltage magnitude measurement at bus i. The neighborhood set of bus i, which consists of all buses directly connected to this bus, is denoted by N_i. The power injections at bus i are described by

$$\begin{array}{rcl} P_i &=& V_i \sum_{j \in N_i} V_j \left(G_{ij} \cos \theta_{ij} + B_{ij} \sin \theta_{ij} \right) \\ Q_i &=& V_i \sum_{j \in N_i} V_j \left(G_{ij} \sin \theta_{ij} - B_{ij} \cos \theta_{ij} \right) \end{array},$$

and the power flows from bus i to bus j are described by

$$\begin{array}{rcl} P_{ij} &=& V_i^2 (g_{si} + g_{ij}) - V_i V_j \left(g_{ij} \cos \theta_{ij} + b_{ij} \sin \theta_{ij} \right) \\ Q_{ij} &=& -V_i^2 (b_{si} + b_{ij}) - V_i V_j \left(g_{ij} \sin \theta_{ij} - b_{ij} \cos \theta_{ij} \right) \end{array},$$

where $\theta_{ij} = \theta_i - \theta_j$ is the phase angle difference between bus i and j, g_{si} and b_{si} are the shunt conductance and susceptance of bus i, g_{ij} and b_{ij} are the conductance and susceptance of the branch from bus i to j, and $Y_{ij} = G_{ij} + jB_{ij}$ is entry (i, j) of the nodal admittance matrix. More detailed formulas may be found in Abur and Exposito (2004).

The nonlinear measurement model is defined by

$$y = h(x) + v, \tag{2.1}$$

where $h(\cdot)$ is the m-dimensional nonlinear measurement function assumed to be twice continuously differentiable, and $v = (v_1, \dots, v_m)^\top$ the measurement error vector. Usually $m \gg n$, meaning that there is high measurement redundancy. We assume v_i are zero-mean independent Gaussian random variables with variances σ_i^2. Thus we have $v \sim \mathcal{N}(0, R)$ where $R = \operatorname{diag}(\sigma_1^2, \dots, \sigma_m^2)$ is the covariance matrix.

DC measurement model: The DC network model is a linear measurement model obtained by neglecting the coupling between active and reactive power components and assuming that the voltage magnitudes are constant at 1pu (per unit), there are no branch resistances and shunt admittances, and the phase-angles θ_i are close to zero. With a slight abuse in notation, in the DC model the state corresponds to the phase-angles and is denoted as $x = \theta$, while the active power measurements are denoted by y. The DC model assumption leads to the measurement equations

$$\begin{aligned} P_i &= \sum_{j \in N_i} b_{ij}(\theta_i - \theta_j) \\ P_{ij} &= -b_{ij}(\theta_i - \theta_j). \end{aligned}$$

The resulting linear measurement model is then given by

$$y = C_{DC}x + v. \tag{2.2}$$

State estimator

The SE problem is to find the best n-dimensional state x for the measurement model (2.1) in a weighted least-squares (WLS) sense. Defining the residue vector $r(x) = y - h(x)$, we can write the unconstrained WLS problem as

$$\underset{x \in \mathbb{R}^n}{\text{minimize}} \quad r(x)^\top R^{-1} r(x). \tag{2.3}$$

The state estimate $\hat{x}$ is a minimizer to this problem. The solution can be found using the Gauss-Newton method, which solves a sequence of the normal equations

$$\left(C^\top(x^k)R^{-1}C(x^k)\right)\Delta x^k = C^\top(x^k)R^{-1}r(x^k), \tag{2.4}$$

for $k = 0, 1, \ldots$, where

$$C(x^k) \triangleq \left.\frac{\partial h(x)}{\partial x}\right|_{x=x^k}$$

is called the Jacobian matrix of the measurement model $h(x)$ and $\Delta x^k = x^{k+1} - x^k$. The normal equations yield a unique solution if the measurement Jacobian matrix $C(x^k)$ is full column rank. In this case, the power network is said to be observable. Consequently, the matrix $\left(C^\top(x^k)R^{-1}C(x^k)\right)$ in (2.4) is then positive definite and the Gauss-Newton step generates a descent direction. The estimation algorithm is finalized when the stopping criteria $\|\Delta x^k\| \leq \epsilon$ for some $\epsilon > 0$ is met.

For notational convenience, throughout the next sections we will use C as $C(x^k)$, Δx as Δx^k, and r as $r(x^k) = y - h(x^k)$.

DC state estimator: Considering the linear DC model (2.2), the SE problem (2.3) reduces to a constrained linear least-squares problem. In the unconstrained-case the optimal estimate is obtained using the normal equation

$$\hat{x} = (C_{DC}^\top R^{-1} C_{DC})^{-1} C_{DC}^\top R^{-1} y.$$

Bad data detection

The BDD detects measurements corrupted by errors (Abur and Exposito, 2004). This can be achieved by hypothesis tests using the statistical properties of the measurement residue (2.5).

Consider the measurement $y = h(x) + v$ and suppose the optimal estimate in the least squares sense, $\hat{x}$, is obtained with the Gauss-Newton method. The first-order approximation of the measurement residue $r(\hat{x}) = y - h(\hat{x})$ is given by

$$r = Sv, \tag{2.5}$$

where $S = I - C(C^\top C)^{-1}C^\top$. An expression similar to (2.5) can be obtained for the measurement residue in the DC SE by replacing C with C_{DC}. Given the residue (2.5), evaluated at the optimal estimate $\hat{x}$, an alarm indicating the presence of bad data is triggered if the residue's norm exceeds a threshold $\delta > 0$:

$$r^\top r = v^\top S v \geq \delta.$$

Wide-area monitoring and control systems

Monitoring schemes are today implemented in a centralized control center through a single state estimator. The core methodology for state estimation of power systems dates from 1970 (Schweppe and Wildes, 1970; Abur and Exposito, 2004). Due to the low sampling frequency of the sensors, a steady-state approach is taken and reliability is ensured by over-constraining the network operation. Dynamic faults, such as generator electro-mechanical oscillations, may pass undetected by such schemes based on steady-state models and measurements.

Recently, measurement units with higher sampling rate have been developed, such as PMUs, opening the way to dynamic state estimators and model-based fault detection schemes taking into account the dynamics of the system. An example of the new opportunities is the WAMS, which uses data from several PMUs to perform real-time monitoring (Machowski *et al.*, 2008). Several implementations of WAMS have recently been performed (Phadke and de Moraes, 2008). In a survey, Chompoobutrgool *et al.* (2011) present an overview of possible uses for WAMS, such as dynamic state estimation and fault monitoring through Kalman filters. These technological developments allow for new opportunities to be envisioned, such as a PMU-enabled WAMS monitoring for the system for physical faults illustrated in Figure 2.11. This can serve as motivation for the contributions in Chapter 6, where a distributed model-based fault monitoring scheme is proposed.

2.5.2 Process Control Testbed

Our process control testbed is the quadruple-tank process (QTP) (Johansson, 2000) controlled through a wireless communication network, as shown in Figure 2.12. The plant model can be found in Johansson (2000)

$$
\begin{aligned}
\frac{\mathrm{d}h_1(t)}{\mathrm{d}t} &= -\frac{a_1}{A_1}\sqrt{2gh_1(t)} + \frac{a_3}{A_1}\sqrt{2gh_3(t)} + \frac{\gamma_1 k_1}{A_1}U_1(t),\\
\frac{\mathrm{d}h_2(t)}{\mathrm{d}t} &= -\frac{a_2}{A_2}\sqrt{2gh_2(t)} + \frac{a_4}{A_2}\sqrt{2gh_4(t)} + \frac{\gamma_2 k_2}{A_2}U_2(t),\\
\frac{\mathrm{d}h_3(t)}{\mathrm{d}t} &= -\frac{a_3}{A_3}\sqrt{2gh_3(t)} + \frac{(1-\gamma_2)k_2}{A_3}U_2(t),\\
\frac{\mathrm{d}h_4(t)}{\mathrm{d}t} &= -\frac{a_4}{A_4}\sqrt{2gh_4(t)} + \frac{(1-\gamma_1)k_1}{A_4}U_1(t),\\
L_1(t) &= h_1(t),\\
L_2(t) &= h_2(t),
\end{aligned}
$$

where $h_i \in [0,\ 30]$ are the water-levels in each tank, A_i the cross-section area of the tanks, a_i the cross-section area of the outlet hole, k_i the pump constants, γ_i the flow ratios and g the gravity acceleration. The system has two outputs $L_1(t)$

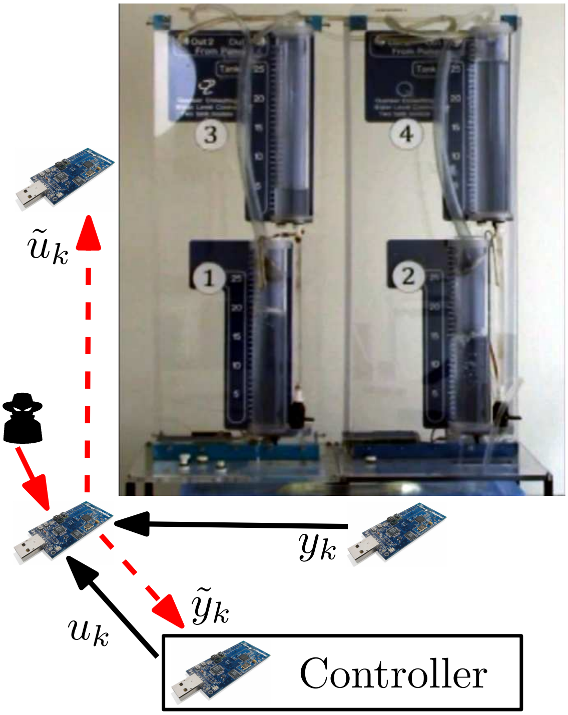


Figure 2.12: Schematic diagram of the testbed with the quadruple-tank process and a multi-hop communication network.

and $L_2(t)$ measuring $h_1(t)$ and $h_2(t)$, respectively, and two inputs, $U_1(t)$ and $U_2(t)$, corresponding to the voltages applied to electrical pumps that drive the flow of water into the tanks.

The system is linearized at a given equilibrium point, denoted as h_i^0, u_i^0, and y_i^0. Defining the state, input, and output of the linearized system as $x_i(t) = h_i(t) - h_i^0$, $u_i(t) = U_i(t) - U_i^0$, and $y_i(t) = L_i(t) - L_i^0$, respectively, the linearized dynamics are given by

$$\begin{aligned} \frac{\mathrm{d}x(t)}{\mathrm{d}t} &= Ax(t) + Bu(t), \\ y(t) &= Cx(t), \end{aligned} \tag{2.6}$$

with

$$A = \begin{bmatrix} -\frac{a_1}{A_1}\sqrt{\frac{g}{2h_1^0}} & 0 & \frac{a_3}{A_1}\sqrt{\frac{g}{2h_3^0}} & 0 \\ 0 & -\frac{a_2}{A_2}\sqrt{\frac{g}{2h_2^0}} & 0 & \frac{a_4}{A_2}\sqrt{\frac{g}{2h_4^0}} \\ 0 & 0 & -\frac{a_3}{A_3}\sqrt{\frac{g}{2h_3^0}} & 0 \\ 0 & 0 & 0 & -\frac{a_4}{A_4}\sqrt{\frac{g}{2h_4^0}} \end{bmatrix},$$

$$B = \begin{bmatrix} \frac{\gamma_1 k_1}{A_1} & 0 \\ 0 & \frac{\gamma_2 k_2}{A_2} \\ 0 & \frac{(1-\gamma_2)k_2}{A_3} \\ \frac{(1-\gamma_1)k_1}{A_4} & 0 \end{bmatrix}, \quad C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

The system has adjustable zero-dynamics with respect to $u_1(t)$ and $u_2(t)$. In particular, the zero-dynamics are unstable if $0 < \gamma_1 + \gamma_2 < 1$ (Johansson, 2000). The process is controlled using a centralized LQG controller with integral action running in a remote computer and a wireless network is used for the communications. A Kalman filter-based anomaly detector is running in the remote computer. The communication network has four nodes, including a relay node, as illustrated in Figure 2.12. Cyber attacks are performed through the relay node. The adversary may access and corrupt all sensor and actuator measurements.

Chapter 3

A Modeling Framework for Constrained Malicious Adversaries

In this chapter, we describe a modeling framework to capture different attack scenarios on control systems. Unlike other information technology (IT) systems where cyber security mainly involves the protection of data-related properties and services, cyber attacks on networked control systems may influence physical processes through feedback actuation. Therefore, networked control system security needs to consider threats at both the cyber and physical layers. Furthermore, in the study of cyber attacks on control systems, it is of the utmost importance to capture the adversary's resources and knowledge. To this end, we propose the attack-scenario space illustrated in Figure 3.1 to capture and qualitatively categorize different cyber threats, which depicts several attack scenarios as points. Note that each example corresponds to a given instance of an attack scenario.

We propose three dimensions for the attack-scenario space: the adversary's *a priori* system model knowledge, disclosure, and disruption resources. Although adversaries possess several other features, the proposed three dimensions are quite relevant from a control system's perspective and allow a straightforward categorization of many attack scenarios studied in the literature. The *a priori* model knowledge can be used by the adversary to construct more complex attacks, possibly harder to detect and with more severe consequences. Similarly, disclosure resources, such as data sniffers, enable the adversary to obtain sensitive information about the system during the attack by violating data confidentiality. Note that disclosure resources alone cannot disrupt the system operation. An example of an attack using only disclosure resources is the eavesdropping attack, illustrated in Figure 3.1. On the other hand, disruption resources, such as data spoofers and jammers, can be used to affect the system operation. For instance, the system operation may be disrupted when data integrity or availability properties are violated. In particular, this characterization fits the Stuxnet malware, which had resources to record and manipulate data in the SCADA network, as described in Chapter 1. Moreover, the complexity and operation of Stuxnet also indicate that its developers had access to

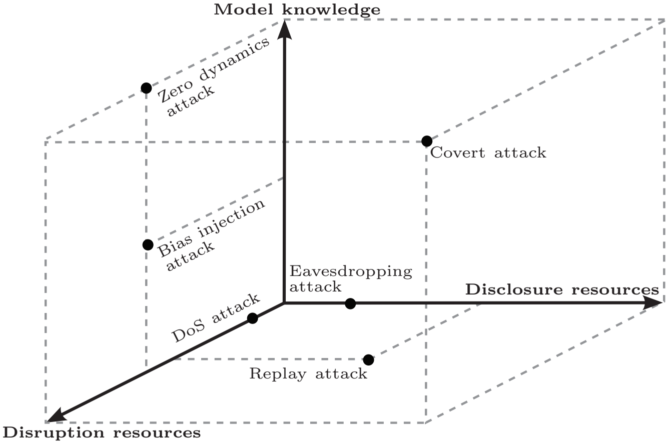


Figure 3.1: The cyber physical attack-scenario space. Each point depicts the qualitative classification of a given attack scenario.

a reasonable amount of knowledge of both physical and cyber components of the target control system.

3.1 Related Work

Cyber-attacks on control systems compromising measurement and actuator data integrity and availability have been considered in Cárdenas *et al.* (2008b), where the authors modeled the attack effects on the physical dynamics. Several attack scenarios have been simulated and evaluated on the Tennessee-Eastman process control system (Cárdenas *et al.*, 2011) to study the attack impact and detectability. The attack scenarios in Cárdenas *et al.* (2011) are related to the ones considered in this chapter, but we quantify the attack resources and policies in a more systematic way.

Availability attacks have been analyzed in Amin *et al.* (2009); Gupta *et al.* (2010) for resource-constrained adversaries with full-state information. Particularly, the authors considered denial-of-service (DoS) attacks in which the adversary could tamper with the communication channels and prevent measurement and actuator data from reaching their destination, rendering the data unavailable. A particular instance of the DoS attack, where the adversary does not have any *a priori* model knowledge, i.e. the attack in Amin *et al.* (2009), is represented in the attack-scenario space in Figure 3.1. However, some instances of DoS attacks may use additional resources and model knowledge, see Gupta *et al.* (2010).

Deception attacks compromising data integrity have recently been tackled. Pang and Liu (2012) proposed an encryption and predictive control scheme to prevent and mitigate deception attacks on control systems. Replay attacks on the sensor measurements, which is a particular kind of deception attack, have been analyzed by Mo and Sinopoli (2009). The authors considered the case where all the existing sensors were attacked and proposed suitable counter-measures to detect the attack. In this attack scenario, the adversary does not have any model knowledge, but is able to access and corrupt the sensor data through disclosure and disruptive resources, as depicted in Figure 3.1.

Another class of deception attacks, false-data injection attacks, has been studied in recent work. For instance, in the case of power networks, an adversary with perfect model knowledge has been considered by Liu *et al.* (2009), who showed that the adversary could corrupt measurements in a coordinated way while remaining undetected. The consequences of these attacks have also been analyzed (Xie *et al.*, 2010; Teixeira *et al.*, 2012a). The models used in the previous work are static, hence these attack scenarios are closest to the bias injection attack shown in Figure 3.1.

Data injection attacks on dynamic control systems were also considered. Smith (2011) characterizes the set of attack policies for covert (undetectable) false-data injection attacks with detailed model knowledge and full access to all sensor and actuator channels. Similarly, Pasqualetti *et al.* (2011) described the set of undetectable false-data injection attacks for omniscient adversaries with full-state information, but possibly compromising only a subset of the existing sensors and actuators. Data confidentiality was violated in these attack scenarios, as the adversary had access to either measurement and actuator data or full-state information. These attacks are, therefore, placed close to the covert attack in Figure 3.1.

Most of the recent work on cyber security of control systems has considered scenarios where the adversary has access to a large set of resources and knowledge, thus being placed far from the origin of the attack-scenario space in Figure 3.1. A large part of the attack-scenario space has not been explored yet. In particular, the class of detectable attacks that do not trigger conventional alarms has yet to be covered in depth.

3.2 Contributions and Outline

In this chapter, we consider a typical networked control architecture under both cyber and physical attacks. A generic adversary model applicable to several attack scenarios is discussed and the attack resources are mapped to the corresponding dimensions of the attack-scenario space depicted in Figure 3.1. Although the framework is presented for linear time-invariant (LTI) systems, the conceptual components and methodology may be applied to other classes of systems.

To illustrate the proposed framework, we consider a LTI system under several attack scenarios, where the adversary's goal is to drive the system to an unsafe state while remaining stealthy. Exploiting the properties of LTI systems, for each scenario we formulate the corresponding stealthy attack policy and comment on the attack's performance. Furthermore, we describe the adversary's capabilities along each dimension of the attack-scenario space in Figure 3.1, namely the disclosure resources, disruption resources, and model knowledge. Some of the attack scenarios analyzed in the thesis have been staged on a SCADA EMS software for power transmission networks and on a wireless quadruple-tank testbed, described in Chapter 2. The results from the staged attacks are presented and discussed later in this chapter.

One of the analyzed attack scenarios corresponds to a novel type of detectable attack, the bias injection attack. Although this attack may be detected, it can drive the system to an unsafe region and it only requires limited model knowledge and no information about the system state. Stealthiness conditions for this attack are provided, as well as a methodology to assess the attack impact on the physical state of the system.

The outline of the chapter is as follows. The system architecture and model are described in Section 3.3, while Section 3.4 contains the adversary model and a detailed description of the attack resources on each dimension of the attack-scenario space. The framework introduced in the previous sections is then illustrated for five particular attack scenarios in Section 3.5, where the adversary aims at driving the system to an unsafe state while remaining stealthy. The attack policy, attack performance, and required model knowledge, disclosure, and disruption resources are described for each attack scenario. The results of the experiments for some of the attack scenarios in two experimental testbeds are presented and discussed in Section 3.6, followed by a summary in Section 3.7.

3.3 Networked Control System

In this section, we describe the networked control system structure, where we consider four main components: the physical plant, the communication network, the feedback controller, and the anomaly detector. Although the networked control system architecture is presented for LTI systems, the same components can be used when considering other classes of systems.

3.3.1 Physical Plant and Communication Network

The physical plant is modeled in a discrete-time state-space form

$$\mathcal{P}: \begin{cases} x_{k+1} = Ax_k + B\tilde{u}_k + Gw_k + Ff_k \\ y_k = Cx_k + v_k, \end{cases} \tag{3.1}$$

where $x_k \in \mathbb{R}^{n_x}$ is the state variable, $\tilde{u}_k \in \mathbb{R}^{n_u}$ the control actions applied to the process, $y_k \in \mathbb{R}^{n_y}$ the measurements from the sensors at the sampling instant $k \in \mathbb{Z}$, and $f_k \in \mathbb{R}^{n_f}$ is the unknown signal representing the effects of anomalies, usually denoted as fault signal in the fault diagnosis literature (Ding, 2008). The process and measurement noise, $w_k \in \mathbb{R}^{n_x}$ and $v_k \in \mathbb{R}^{n_y}$, represent the discrepancies between the model and the real process, due to unmodeled dynamics or disturbances, for instance, and we assume their means are respectively bounded by δ_w and δ_v, i.e. $\bar{w} = \|\mathbb{E}\{w_k\}\| \leq \delta_w$ and $\bar{v} = \|\mathbb{E}\{v_k\}\| \leq \delta_v$.

The physical plant operation is supported by a communication network through which the sensor measurements and actuator data are transmitted, which at the plant side correspond to y_k and $\tilde{u}_k$, respectively. At the controller side we denote the sensor and actuator data by $\tilde{y}_k \in \mathbb{R}^{n_y}$ and $u_k \in \mathbb{R}^{n_u}$, respectively. Since the communication network may be unreliable, the data exchanged between the plant and the controller may be altered, resulting in discrepancies in the data at the plant and controller ends. In this work we do not consider the usual communication network effects such as packet losses and delays. Instead, we focus on data corruption due to malicious cyber attacks, as described in Section 3.4. Therefore, it is assumed that, first, any possible mismatches between the transmitted and received data are due to malicious adversaries alone. Second, the communication network is assumed to be reliable and not affecting the data flowing through it.

Given the physical plant model (3.1) and assuming an ideal communication network, the networked control system is said to have a *nominal behavior* if $f_k = 0$, $\tilde{u}_k = u_k$, and $\tilde{y}_k = y_k$. The absence of either one of these condition results in an abnormal behavior of the system.

3.3.2 Feedback Controller

In order to comply with performance requirements in the presence of the unknown process and measurement noises, we consider that the physical plant is controlled by an appropriate linear time-invariant feedback controller (Zhou *et al.*, 1996). The output-feedback controller can be written in a state-space form as

$$\mathcal{F} : \begin{cases} z_{k+1} = A_c z_k + B_c \tilde{y}_k \\ u_k = C_c z_k + D_c \tilde{y}_k, \end{cases} \tag{3.2}$$

where the states of the controller, $z_k \in \mathbb{R}^{n_z}$, may include the process state and tracking-error estimates. Given the plant and communication network models, the controller is supposed to be designed so that acceptable performance is achieved under nominal behavior.

3.3.3 Anomaly Detector

In this subsection we consider the anomaly detector that monitors the system to detect possible anomalies, i.e. deviations from the nominal behavior. The anomaly detector is supposed to be collocated with the controller, therefore it only has access to $\tilde{y}_k$ and u_k to evaluate the behavior of the plant.

Several approaches to detecting malfunctions in control systems are available in the fault diagnosis literature (Ding, 2008; Hwang *et al.*, 2010). Here we consider a general form of an observer-based fault detection filter

$$\mathcal{D} : \begin{cases} s_{k+1} = A_e s_k + B_e u_k + K_e \tilde{y}_k \\ r_k = C_e s_k + D_e u_k + E_e \tilde{y}_k, \end{cases} \tag{3.3}$$

where $s_k \in \mathbb{R}^{n_s}$ is the state of the anomaly detector and $r_k \in \mathbb{R}^{n_r}$ is the residue evaluated to detect and locate existing anomalies.

Define $\|r_k\|_p \triangleq \left(\sum_{i=1}^{n_r} |r_{(i),k}|^p\right)^{1/p}$ as the p-norm of r_k for $1 \leq p < \infty$, with $r_{(i),k}$ as the i-th entry of the vector r_k and $\|r_k\|_\infty \triangleq \max_i \ |r_{(i),k}|$. The anomaly detector is designed by choosing A_e, B_e, K_e, C_e, D_e, and E_e such that

1. under nominal behavior of the system (i.e., $f_k = 0$, $u_k = \tilde{u}_k$, $y_k = \tilde{y}_k$), the expected value of r_k converges asymptotically to a neighborhood of zero, i.e., $\lim_{k\to\infty} \mathbb{E}\{r_k\} \in \mathcal{B}_{\delta_r}$, with $\delta_r \geq 0$ and $\mathcal{B}_{\delta_r} \triangleq \{r \in \mathbb{R}^{n_r} : \ \|r\|_p \leq \delta_r\}$;

2. the residue is sensitive to the anomalies (i.e., different fault signals with $f_k \not\equiv 0$ and $f_k \equiv 0$ for all k result in different residues).

The characterization of $\mathcal{B}_{\delta_r}$ depends on the noise terms and can be found in Ding (2008) for particular values of p. Given the residue signal over the time-interval $[k_0,\ k_f]$, $\mathbf{r}_{[k_0,\ k_f]} = [r_{k_0}^\top \ \ldots \ r_{k_f}^\top]^\top$, an alarm is triggered if

$$\mathbf{r}_{[k_0,\ k_f]} \notin \mathcal{U}_{[k_0,\ k_f]}, \tag{3.4}$$

where the set $\mathcal{U}_{[k_0,\ k_f]}$ is chosen so that the number of anomaly misdetections and false-alarms are minimized. This necessarily requires no alarm to be triggered in the noiseless nominal behavior i.e., $\mathbf{r}_{[k_0,\ k_f]} \in \mathcal{U}_{[k_0,\ k_f]}$ if for all $k \in [k_0,\ k_f]$ it holds that $r_k \in \mathcal{B}_{\delta_r}$. Such set-based detection fits several residual evaluation techniques presented in Frank and Ding (1997). For instance, one can take $\mathcal{U}_{[k_0,\ k_f]}$ to be a bound on the energy of the residue signal over the time-interval $[k_0,\ k_f]$, resulting in

$$\mathcal{U}_{[k_0,\ k_f]} = \{\mathbf{r}_{[k_0,\ k_f]} : \|\mathbf{r}_{[k_0,\ k_f]}\|_2 \leq \delta\},$$

for some $\delta \in (0,\ \infty)$.

3.4 Adversary Models

The adversary model considered in this thesis is illustrated in Figure 3.2 and is composed of an attack policy and the adversary resources i.e., the system model knowledge, the disclosure resources, and the disruption resources. Each of the adversary resources can be mapped to a specific axis of the attack-scenario space in Figure 3.1: $\mathcal{K} = \{\hat{\mathcal{P}},\ \hat{\mathcal{F}},\ \hat{\mathcal{D}}\}$ is the *a priori* model knowledge possessed by the adversary; $\mathcal{I}_k$ corresponds to the set of sensor and actuator data available to the adversary at time k, thus being mapped to the disclosure resources; a_k is the attack vector at time k that may affect the system behavior using the disruption resources captured by $\mathbf{B}$, as defined later in the present section. The attack policy mapping $\mathcal{K}$ and $\mathcal{I}_k$ to a_k at time k is denoted as

$$a_k = g(\mathcal{K}, \mathcal{I}_k).$$

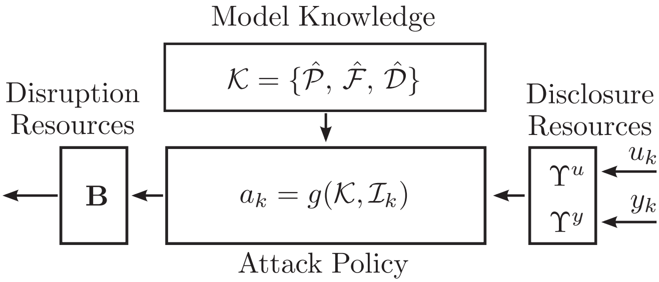


Figure 3.2: Adversary model for a point in the attack-scenario space in Figure 3.1.

Examples of attacks policies for different attack scenarios are given in Section 3.5.

In this section, we describe the networked control system under attack with respect to the attack vector a_k. Then, we detail the adversary's model knowledge, the disclosure resources, and the disruption resources. Models of the attack vector a_k for particular disruption resources are also given.

3.4.1 Networked Control System under Attack

The system components under attack are now characterized for the attack vector a_k, which also includes the fault vector f_k. Stacking the states of the plant and controller as $\eta_k = [x_k^\top \quad z_k^\top]^\top$, the dynamics of the closed-loop system composed by $\mathcal{P}$ and $\mathcal{F}$ under the effect of a_k can be written as

$$\begin{aligned}
\eta_{k+1} &= \mathbf{A}\eta_k + \mathbf{B}a_k + \mathbf{G}\begin{bmatrix} w_k \\ v_k \end{bmatrix} \\
\tilde{y}_k &= \mathbf{C}\eta_k + \mathbf{D}a_k + \mathbf{H}\begin{bmatrix} w_k \\ v_k \end{bmatrix} \\
u_k &= \mathbf{C}_u\eta_k + D_c\mathbf{D}a_k + D_c\mathbf{H}\begin{bmatrix} w_k \\ v_k \end{bmatrix},
\end{aligned} \tag{3.5}$$

where the system matrices are

$$\mathbf{A} = \begin{bmatrix} A + BD_cC & BC_c \\ B_cC & A_c \end{bmatrix}, \quad \mathbf{G} = \begin{bmatrix} G & BD_c \\ 0 & B_c \end{bmatrix},$$

$$\mathbf{C} = \begin{bmatrix} C & 0 \end{bmatrix}, \mathbf{H} = \begin{bmatrix} 0 & I \end{bmatrix}, \quad \mathbf{C}_u = \begin{bmatrix} D_cC & C_c \end{bmatrix}.$$

The matrices $\mathbf{B}$ and $\mathbf{D}$ capture the way in which the attack vector a_k affects the plant and controller. These matrices are characterized for some attack scenarios in Section 3.4.4.

Similarly, using $\mathcal{P}$, $\mathcal{F}$, and $\mathcal{D}$ as in (3.1), (3.2), and (3.3), respectively, and stacking the states of the plant, controller, and anomaly detector as $\xi_k = [\eta_k^\top \quad s_k^\top]^\top$ the residue dynamics under attack are described by

$$\begin{aligned} \xi_{k+1} &= \mathbf{A}_e \xi_k + \mathbf{B}_e a_k + \mathbf{G}_e \begin{bmatrix} w_k \\ v_k \end{bmatrix} \\ r_k &= \mathbf{C}_e \xi_k + \mathbf{D}_e a_k + \mathbf{H}_e \begin{bmatrix} w_k \\ v_k \end{bmatrix}, \end{aligned} \tag{3.6}$$

where

$$\begin{aligned} \mathbf{A}_e &= \begin{bmatrix} \mathbf{A} & 0 \\ B_e \mathbf{C}_u + K_e \mathbf{C} & A_e \end{bmatrix}, & \mathbf{B}_e &= \begin{bmatrix} \mathbf{B} \\ (B_e D_c + K_e)\mathbf{D} \end{bmatrix}, \\ \mathbf{C}_e &= \begin{bmatrix} D_e \mathbf{C}_u + E_e \mathbf{C} & C_e \end{bmatrix}, & \mathbf{G}_e &= \begin{bmatrix} \mathbf{G} \\ (B_e D_c + K_e)\mathbf{H} \end{bmatrix}, \\ \mathbf{D}_e &= (D_e D_c + E_e)\mathbf{D}, & \mathbf{H}_e &= (D_e D_c + E_e)\mathbf{H}. \end{aligned}$$

3.4.2 Model Knowledge

The amount of *a priori* knowledge regarding the control system is a core component of the adversary model, as it may be used, for instance, to render the attack undetectable. In general, we may consider that the adversary has an estimate of the model of the plant ($\hat{\mathcal{P}}$) and the algorithms used in the feedback controller ($\hat{\mathcal{F}}$) and the anomaly detector ($\hat{\mathcal{D}}$), thus denoting the adversary knowledge by $\mathcal{K} = \{\hat{\mathcal{P}}, \hat{\mathcal{F}}, \hat{\mathcal{D}}\}$. Figure 3.1 illustrates several types of attack scenarios with different levels of model knowledge. In particular, note that the replay attacks do not need any knowledge of the system components, thus having $\mathcal{K} = \emptyset$, while the covert attack requires full knowledge about the system, hence $\mathcal{K} = \{\mathcal{P}, \mathcal{F}, \mathcal{D}\}$.

3.4.3 Disclosure Resources

The disclosure resources enable the adversary to gather sequences of data from the calculated control actions u_k and the sensor measurements y_k through disclosure attacks. Denote $\mathcal{R}^u \subseteq \{1, \ldots, n_u\}$ and $\mathcal{R}^y \subseteq \{1, \ldots, n_y\}$ as the disclosure resources, i.e. the set of actuator and sensor channels that can be accessed during disclosure attacks, and let $\mathcal{I}_k$ be the control and measurement data sequence gathered by the adversary from time k_0 to k. The disclosure attacks can then be modeled as

$$\mathcal{I}_k \triangleq \mathcal{I}_{k-1} \cup \left\{ \begin{bmatrix} \Upsilon^u & 0 \\ 0 & \Upsilon^y \end{bmatrix} \begin{bmatrix} u_k \\ y_k \end{bmatrix} \right\},$$

where $\mathcal{I}_{k_0} = \emptyset$ and $\Upsilon^u \in \mathbb{B}^{|\mathcal{R}^u| \times n_u}$ and $\Upsilon^y \in \mathbb{B}^{|\mathcal{R}^y| \times n_y}$ are the time-invariant binary incidence matrices mapping the data channels to the corresponding data gathered by the adversary.

As seen in the above description of disclosure attacks, the physical dynamics of the system are not affected by these type of attacks. Instead, these attacks gather intelligence that may enable more complex attacks, such as the replay attacks depicted in Figure 3.1.

3.4.4 Disruption Resources

Disruption resources are related to the attack vector a_k and may be used to affect the several components of the system, as seen in the system dynamics under attack (3.5) and (3.6). The way a particular attack disturbs the system operation depends not only on the respective resources, but also on the nature of the attack. For instance, a physical attack directly perturbs the system dynamics, whereas a cyber attack disturbs the system through the cyber physical couplings. To better illustrate this discussion we now consider physical and data deception attacks.

Physical Attacks

Physical attacks may occur in control systems, often in conjunction with cyber attacks. For instance, in the experiments reported in Amin *et al.* (2010), water was pumped out of an irrigation system, while the water level measurements were corrupted so that the attack remained stealthy. Since physical attacks are similar to the fault signals in (3.1), in the following sections we consider f_k to be the physical attack modifying the plant dynamics as

$$\begin{aligned} x_{k+1} &= Ax_k + B\tilde{u}_k + Gw_k + Ff_k \\ y_k &= Cx_k. \end{aligned}$$

Considering $a_k = f_k$, the resulting system dynamics are described by (3.5) and (3.6) with

$$\mathbf{B} = \begin{bmatrix} F \\ 0 \end{bmatrix}, \quad \mathbf{D} = 0.$$

Note that the disruption resources in this attack are captured by the matrix $F \in \mathbb{R}^{n_x \times n_f}$.

Data Deception Attacks

The deception attacks modify the control actions u_k and sensor measurements y_k from their calculated or real values to the corrupted signals $\tilde{u}_k$ and $\tilde{y}_k$, respectively. Denoting $\mathcal{R}_I^u \subseteq \{1, \dots, n_u\}$ and $\mathcal{R}_I^y \subseteq \{1, \dots, n_y\}$ as the deception resources, i.e. set of actuator and sensor channels that can be affected, the deception attacks are modeled as

$$\tilde{u}_k \triangleq u_k + \Gamma^u b_k^u, \quad \tilde{y}_k \triangleq y_k + \Gamma^y b_k^y, \tag{3.7}$$

where the signals $b_k^u \in \mathbb{R}^{|\mathcal{R}_I^u|}$ and $b_k^y \in \mathbb{R}^{|\mathcal{R}_I^y|}$ represent the data corruption and $\Gamma^u \in \mathbb{B}^{n_u \times |\mathcal{R}_I^u|}$ and $\Gamma^y \in \mathbb{B}^{n_y \times |\mathcal{R}_I^y|}$ ($\mathbb{B} \triangleq \{0, 1\}$) are the binary incidence matrices mapping the data corruption to the respective data channels. The matrices Γ^u and Γ^y indicate which data channels can be accessed by the adversary and are directly related to the adversary resources in deception attacks.

Defining $a_k = [b_k^{u\top} \quad b_k^{y\top}]^\top$, the system dynamics are given by (3.5) and (3.6) with

$$\mathbf{B} = \begin{bmatrix} B\Gamma^u & BD_c\Gamma^y \\ 0 & B_c\Gamma^y \end{bmatrix}, \quad \mathbf{D} = \begin{bmatrix} 0 & \Gamma^y \end{bmatrix}.$$

Note that deception attacks do not possess any disclosure capabilities, as depicted in Figure 3.1 for examples of deception attacks such as the bias injection attack.

3.4.5 Attack Goals and Constraints

In addition to the attack resources, the attack scenarios need to also include the intent of the adversary, namely the attack goals and constraints shaping the attack policy $g(\cdot, \cdot)$. The attack goals can be stated in terms of the attack impact on the system operation, while the constraints may be related to the attack detectability.

Several physical systems have tight operating constraints that, if not satisfied, might result in physical damage to the system. In this work, we use the concept of safe regions to characterize safety constraints:

Definition 3.4.1. *At a given time instant k, the system is said to be safe if $x_k \in \mathcal{S}_x$, where $\mathcal{S}_x$ is a closed and compact set with non-empty interior.*

Assumption 3.4.1. *The system is in a safe state at the beginning of the attack, i.e. $x_{k_0} \in \mathcal{S}_x$.*

The physical impact of an attack can be evaluated by assessing whether or not the state of the system remained in the safe set during and after the attack. The attack is considered successful if the state is driven out of the safe set.

Regarding the attack constraints, we consider that adversaries are constrained to remain stealthy. Furthermore, we consider the disruptive attack component consists of only physical and data deception attacks, and thus we have the attack vector $a_k = [f_k^\top \quad b_k^{u\top} \quad b_k^{y\top}]^\top$. Given the anomaly detector described in Section 3.3 and denoting $\mathbf{a}_{[k_0,\, k_f]} = [a_{k_0}^\top \ \dots \ a_{k_f}^\top]^\top$ as the attack signal, the set of stealthy attacks are defined as follows:

Definition 3.4.2. *The attack signal $\mathbf{a}_{[k_0,\, k_f]}$ is stealthy over the time-interval $[k_0, k_f]$ if $\mathbf{r}_{[k_0,\, k_f]} \in \mathcal{U}_{[k_0,\, k_f]}$.*

Note that the above definition is dependent on the initial state of the system at k_0, as well as the noise terms w_k and v_k.

Since the closed-loop system (3.5) and the anomaly detector (3.6) under linear attack policies are LTI systems, each of these systems can be separated into two additive components: the nominal component with $a_k = 0$ and the following systems

$$\begin{aligned} \eta^a_{k+1} &= \mathbf{A}\eta^a_k + \mathbf{B}a_k \\ \tilde{y}^a_k &= \mathbf{C}\eta^a_k + \mathbf{D}a_k \end{aligned} \tag{3.8}$$

and

$$\begin{aligned} \xi^a_{k+1} &= \mathbf{A}_e\xi^a_k + \mathbf{B}_e a_k \\ r^a_k &= \mathbf{C}_e\xi^a_k + \mathbf{D}_e a_k, \end{aligned} \tag{3.9}$$

with $\eta^a_0 = \xi^a_0 = 0$.

Assume that the system is behaving nominally before the attack and that, given the linearity of (3.6), there exists a set $\mathcal{U}^a_{[k_0,\, k_f]} \triangleq \{\mathbf{r}_{[k_0,\, k_f]} :\ \|\mathbf{r}_{[k_0,\, k_f]}\|_q \leq \delta\}$ such that having $\mathbf{r}^a_{[k_0,\, k_f]} \in \mathcal{U}^a_{[k_0,\, k_f]}$ implies that $\mathbf{r}_{[k_0,\, k_f]} \in \mathcal{U}_{[k_0,\, k_f]}$ also holds. We make the following definition:

Definition 3.4.3. *The attack signal* $\mathbf{a}_{[k_0,\, k_f]}$ *is* δ-stealthy *over the time-interval* $[k_0,\ k_f]$ *if* $\mathbf{r}^a_{[k_0,\, k_f]} \in \mathcal{U}^a_{[k_0,\, k_f]}$.

Albeit more conservative than Definition 3.4.2, Definition 3.4.3 only depends on the attack signals $\mathbf{a}_{[k_0,\, k_f]}$. Thus the stealthiness of linear attacks on LTI systems may be analyzed independently of the noise inputs. Similarly, the impact of attacks on the closed-loop system can be analyzed through the linear system (3.8), as illustrated in Section 3.5.5 for the bias injection attack. For other classes of systems, e.g., nonlinear or switched systems, the analysis and characterization of attacks may have to consider the noise terms directly.

3.5 Attack Scenarios

In this section, using the framework introduced earlier, we consider several attack scenarios where the adversary's goal is to drive the system to an unsafe state while remaining stealthy. For each scenario, we formulate the corresponding stealthy attack policy and comment on the attack's performance. Furthermore, we also describe the adversary's capabilities along each dimension of the attack-scenario space in Figure 3.1, namely the disclosure resources, disruption resources, and model knowledge. A subset of these scenarios is illustrated by experiments on a process control testbed in Section 3.6.

3.5.1 Denial-of-Service Attack

The DoS attacks prevent the actuator and sensor data from reaching their respective destinations and results in the absence of data. To model absent data, we consider one of the typical mechanisms used by digital controllers to deal with unavailable data (Schenato, 2009), in which the absent data are replaced with the last received data, u_{τ_u} and y_{τ_y} respectively.

Attack policy: Denote $\mathcal{R}^u_A \subseteq \{1, \ldots, n_u\}$ and $\mathcal{R}^y_A \subseteq \{1, \ldots, n_y\}$ as the set of actuator and sensor channels that can be made unavailable and define $S^u_k \in \mathbb{B}^{|\mathcal{R}^u_A| \times |\mathcal{R}^u_A|}$ and $S^y_k \in \mathbb{B}^{|\mathcal{R}^y_A| \times |\mathcal{R}^y_A|}$ as boolean diagonal matrices where the i-th diagonal entry indicates whether a DoS attack is performed ($[S^{(\cdot)}_k]_{ii} = 1$) or not ($[S^{(\cdot)}_k]_{ii} = 0$) on the corresponding channel. Using the latter variables, DoS attacks can be modeled as deception attacks in (3.7) with

$$\begin{aligned} b^u_k &\triangleq -S^u_k \Gamma^{u\top}(u_k - u_{\tau_u}) \\ b^y_k &\triangleq -S^y_k \Gamma^{y\top}(y_k - y_{\tau_y}) \end{aligned} \tag{3.10}$$

and

$$a_k = \begin{bmatrix} -S^u_k \Gamma^{u\top}(u_k - u_{\tau_u}) \\ -S^y_k \Gamma^{y\top}(y_k - y_{\tau_y}) \end{bmatrix}$$

Therefore DoS attacks on the data are a type of disruptive attacks, as depicted in Figure 3.1.

The attack scenario analyzed in this section considers a Bernoulli adversary on the sensor channels following the random policy

$$\begin{aligned} &\mathbb{P}([S^y_k]_{ii} = 1) = 0,\ \forall i = 1, \ldots, |\mathcal{R}^u_A|, \quad k < k_0 \\ &\mathbb{P}([S^y_k]_{ii} = 1) = p,\ \forall i = 1, \ldots, |\mathcal{R}^u_A|, \quad k \geq k_0 \end{aligned}$$

where $p \in [0,\ 1]$ is the probability of blocking the data packet at any given time (Amin *et al.*, 2009).

Attack performance: Although the absence of data packets is not stealthy since it is trivially detectable, DoS attacks may be misdiagnosed as a poor network condition. As for the impact on the closed-loop system, the results available for Bernoulli packet losses readily apply to the current attack scenario (Zhang *et al.*, 2001; Schenato *et al.*, 2007; Schenato, 2009). In particular, we recall the following result applied to the DoS attack (3.10):

Proposition 3.5.1 (Theorem 8 in Zhang *et al.* (2001))**.** *Assume that the closed-loop system with no DoS attack is stable and consider the open-loop system*

$$\eta_{k+1} = \underbrace{\begin{bmatrix} A & BC_c \\ 0 & A_c \end{bmatrix}}_{\mathbf{A}_o} \eta_k.$$

Then, the closed-loop system with Bernoulli DoS attacks is exponentially stable:

1. *for $p \in [0,\ 1)$, if the open-loop system is marginally stable.*
2. *for $p \in [0,\ \bar{p})$, if the open-loop system is unstable, where $\bar{p} = \dfrac{1}{1 - \gamma_2/\gamma_1}$ with $\gamma_1 = \log\left(\max_i\ |\lambda_i(\mathbf{A}_c)|^2\right)$ and $\gamma_2 = \log\left(\max_i\ |\lambda_i(\mathbf{A}_o)|^2\right)$.*

As stated by the previous results, if the open-loop system is unstable, the Bernoulli DoS attack may lead to an unstable closed-loop system, if p is sufficiently close to 1. On the other hand, for open-loop stable systems, the closed-loop system under Bernoulli DoS attacks remains stable.

Disclosure resources: Although the proposed model of DoS attacks in (3.10) contains the control and output signals, note that no disclosure resources are needed in the actual implementation of the attack. Thus we have $\mathcal{R}^u = \mathcal{R}^y = \emptyset$.

Disruption resources: The disruption capabilities correspond to the data channels that the adversary is able to make unavailable, $\mathcal{R}^u_A$ and $\mathcal{R}^y_A$.

Model knowledge: For the Bernoulli attack policy, no *a priori* knowledge of the system model is needed.

3.5.2 Replay Attack

In replay attacks the adversary first performs a disclosure attack from $k = k_0$ until k_r, gathering sequences of data $\mathcal{I}_{k_r}$, and then begins replaying the recorded data at time $k = k_r + 1$ until the end of the attack at $k = k_f > k_r$, as illustrated in Figure 3.3. In the scenario considered here the adversary is also able to perform a physical attack while replaying the recorded data, which covers the experiment on a water management SCADA system reported in Amin *et al.* (2010) and one of Stuxnet's operation mode (Falliere *et al.*, 2011).

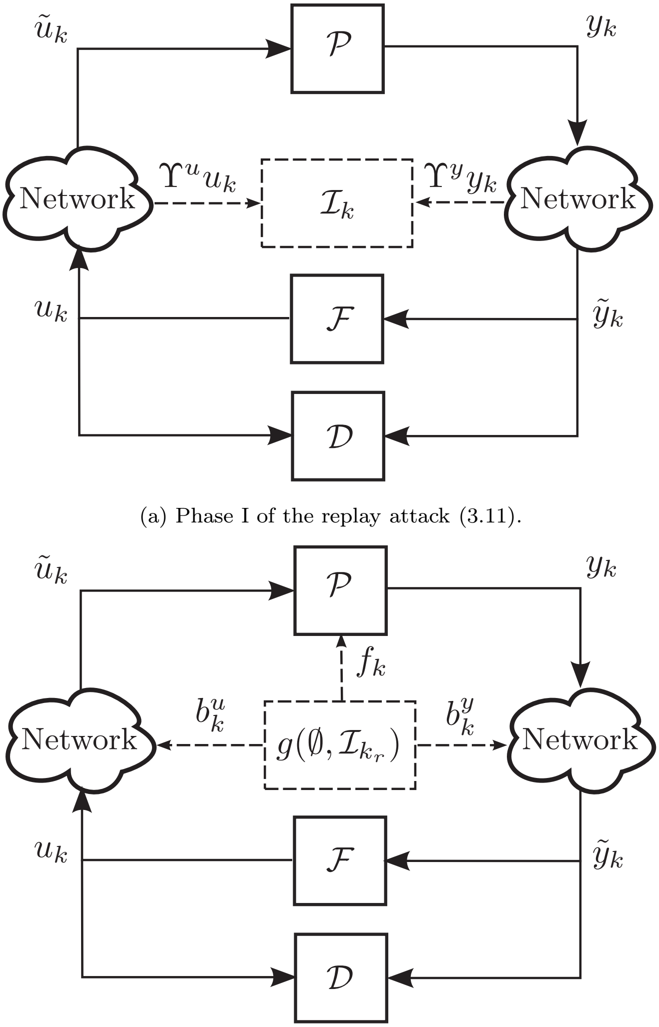
Attack policy: Similar to the work by Mo and Sinopoli (2009), assuming $\mathcal{R}^{(\cdot)} = \mathcal{R}^{(\cdot)}_I$ i.e., the adversary can corrupt the digital channels from which the data sequences are gathered, the replay attack policy can be described in two phases:

$$\text{Phase I:} \quad \begin{cases} a_k = 0, \\ \mathcal{I}_k = \mathcal{I}_{k-1} \cup \left\{ \begin{bmatrix} \Upsilon^u & 0 \\ 0 & \Upsilon^y \end{bmatrix} \begin{bmatrix} u_k \\ y_k \end{bmatrix} \right\}, \end{cases} \tag{3.11}$$

with $k_0 \leq k \leq k_r$ and $\mathcal{I}_{k_0} = \emptyset$ and

$$\text{Phase II:} \quad \begin{cases} a_k = \begin{bmatrix} g_f(\mathcal{K}, \mathcal{I}_{k_r}) \\ \Upsilon^u(u_{k-T} - u_k) \\ \Upsilon^y(y_{k-T} - y_k) \end{bmatrix}, \\ \mathcal{I}_k = \mathcal{I}_{k-1}, \end{cases} \tag{3.12}$$

where $T = k_r - 1 + k_0$ and $k_r + 1 \leq k \leq k_f$. An interesting instance of this attack scenario consists of applying a pre-defined physical attack to the plant, while using replay attacks to render the attack stealthy. In this case the physical attack signal f_k corresponds to an open-loop signal, $f_k = g_f(k)$. Note that, while (3.12) resembles a time-delay of length T, replay attacks differ from delayed data in a subtle but important manner: all measurement data during the attack interval $[k_r + 1,\, k_f]$



(a) Phase I of the replay attack (3.11).

(b) Phase II of the replay attack (3.12).

Figure 3.3: Schematic of the replay attack.

are never available to the anomaly detector. As in Amin *et al.* (2010), this allows the adversary to design the attack so that no alarm is triggered by the anomaly detector.

Attack performance: Mo and Sinopoli (2009) provided conditions under which replay attacks with access to all measurement data channels are stealthy. However, these attacks are not guaranteed to be stealthy when only a subset of the data channels is attacked. In this case, the stealthiness constraint may require additional knowledge of the system model. For instance, the experiment presented in Section 3.6 requires knowledge of the physical system structure, so that f_k only excites the attacked measurements. Hence f_k can be seen as a zero-dynamics attack with respect to the uncompromised measurements, which is characterized in the section below. Since the impact of the replay attack is dependent only on f_k, we refer the reader to Section 3.5.3 for a characterization of the replay attack's impact.

Disclosure resources: The disclosure capabilities required to stage this attack correspond to the data channels that can be eavesdropped by the adversary, namely $\mathcal{R}^u$ and $\mathcal{R}^y$.

Disruption resources: In this case the deception capabilities correspond to the data channels that the adversary can tamper with, $\mathcal{R}^u_I$ and $\mathcal{R}^y_I$. In particular, for replay attacks the adversary can only tamper with the data channels from which data has been previously recorded, i.e. $\mathcal{R}^u_I \subseteq \mathcal{R}^u$ and $\mathcal{R}^y_I \subseteq \mathcal{R}^y$.

Direct disruption of the physical system through the signal f_k depends on having direct access to the physical system, modeled by the matrix F in (3.1).

Model knowledge: Note that no *a priori* knowledge $\mathcal{K}$ on the system model is needed for the cyber component of the attack, namely the data disclosure and deception attack, as seen in the attack policy (3.11) and (3.12). As for the physical attack, f_k, the required knowledge is scenario dependent. In the scenario considered in the experiments described in Section 3.6, this component was modeled as an open-loop signal, $f_k = g_f(k)$.

3.5.3 Zero-Dynamics Attack

Recalling that for linear attack policies the plant and the anomaly detector are LTI systems ((3.8) and (3.9) respectively), Definition 3.4.3 states that attacks are $0-$stealthy (i.e., δ-stealthy with $\delta = 0$) if $r^a_k = 0$ for all $k \geq k_0$. The idea of $0-$stealthy attacks consists of designing an attack policy and attack signal $\mathbf{a}_{[k_0,\, k_f]}$ so that the residue r_k does not change due to the attack. In other words, these attacks are decoupled from the output of the closed-loop linear system (3.6), namely r_k, and their design in general depends on the plant, controller, and anomaly detector dynamics. A particular subset of $0-$stealthy attacks that only depend on the plant dynamics are characterized in the following lemma:

Lemma 3.5.2. *The attack signal* $\mathbf{a}_{[k_0,\, k_f]}$ *is* $0-$*stealthy with respect to an arbitrary anomaly detector* $\mathcal{D}$ *if* $\tilde{y}^a_k = 0$, $\forall k \geq k_0$.

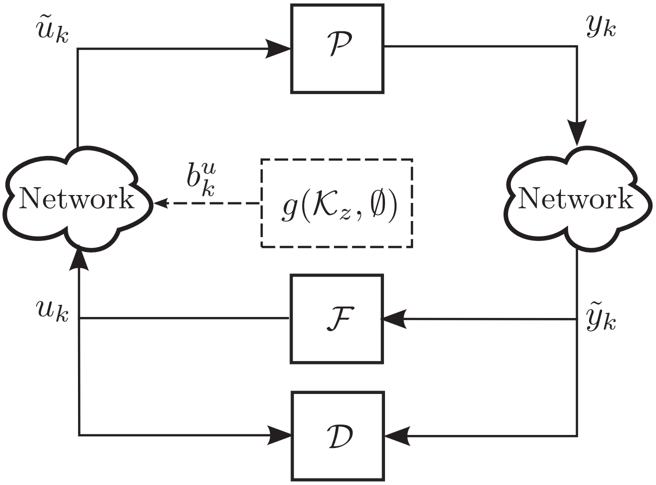


Figure 3.4: Schematic of the zero-dynamics attack.

Proof. Consider an arbitrary controller, anomaly detector, and their corresponding attacked components in (3.8) and (3.9) with $s^a_0 = 0$. From the controller dynamics, it directly follows that $\tilde{y}^a_k = 0$, $\forall k \geq k_0$ results in $u^a_k = 0$, $\forall k \geq k_0$, as the input to the controller, i.e. $\tilde{y}^a_k$, is zero. Since $s^a_0 = 0$ and $\tilde{y}^a_k = u^a_k = 0$, $\forall k \geq k_0$, meaning that the detector's inputs are zero, we then conclude $r^a_k = 0$, $\forall k \geq k_0$. □

Lemma 3.5.2 indicates that 0-stealthy attacks are decoupled from the plant output y_k, thus being stealthy with respect to arbitrary anomaly detectors. Hence finding $0-$stealthy attack signals relates to the output-zeroing problem or zero-dynamics studied in the control theory literature (Zhou *et al.*, 1996). The zero-dynamics attack will be analyzed in further detail in Chapter 4 and Chapter 5.

Note that such an attack requires the perfect knowledge of the plant dynamics $\mathcal{P}$ and the attack signal is based on the open-loop prediction of the output changes due to the attack. This is illustrated in Figure 3.4 where $\mathcal{K}_z$ denote the zero-dynamics and there is no disclosure of sensor or actuator data.

Attack policy: The attack policy corresponds to the input sequence a_k that makes the outputs of the process $\tilde{y}^a_k$ identically zero for all k and is illustrated in Figure 3.4. It can be shown (Zhou *et al.*, 1996) that the solution to this problem is given by the sequence

$$a_k = \nu^k g, \tag{3.13}$$

parameterized by the system zero ν and the corresponding input-zero direction g.

For sake of simplicity we consider a particular instance of this attack, where only the actuator data are corrupted. In this case the zero attack policy corresponds to the transmission zero-dynamics of the plant. The plant dynamics due to an attack on the actuator data are described by

$$\begin{aligned} x^a_{k+1} &= Ax^a_k + Ba_k \\ \tilde{y}^a_k &= Cx^a_k \end{aligned} \tag{3.14}$$

with $a_k = b^u_k$. Given the discrete-time system (3.14) with B having full column rank, the transmission zeros can be calculated as the values $\nu \in \mathbb{C}$ that cause the often called Rosenbrock matrix $P(\nu)$ to lose rank, where

$$P(\nu) = \begin{bmatrix} \nu I - A & -B \\ C & 0 \end{bmatrix}.$$

Those values are called minimum phase or non-minimum phase zeros depending on whether they are stable or unstable zeros, respectively. In discrete-time systems a zero is stable if $|\nu| < 1$ and unstable otherwise.

The input-zero direction can be obtained by solving the following equation

$$\begin{bmatrix} \nu I - A & -B \\ C & 0 \end{bmatrix} \begin{bmatrix} x_0 \\ g \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \tag{3.15}$$

where x_0 is the initial state of the system for which the input sequence (3.13) results in an identically zero output, $\tilde{y}^a_k = 0\,\forall k$.

Lemma 3.5.3. *Let x_0 be the initial state of the system, where x_0 satisfies* (3.15). *The state trajectories generated by the attack are contained in span(x_0), i.e., $x^a_k \in$ span$(x_0)\ \forall k \geq 0$.*

Proof. The proof follows an induction argument. Consider the zero-dynamics attack parameterized by x_0 and g and the state evolution under attack, $x^a_{k+1} = Ax^a_k + \nu^k Bg$ with $x^a_0 = x_0$. For $k = 0$ it follows from (3.15) that $x^a_1 = Ax_0 + Bg = \nu x_0$. Supposing that $x^a_k = \nu^k x_0$ holds for some $k > 0$ yields $x^a_{k+1} = Ax^a_k + \nu^k Bg = \nu^k(Ax^a_0 + Bg) = \nu^{k+1}x_0$ for all $k \geq 0$, thus concluding the proof. □

Attack performance: Note that the zero-dynamics attack is $0-$stealthy only if $x^a_0 = x_0$. However the initial state of the system under attack x^a_0 is defined to be zero at the beginning of the attack. Therefore stealthiness of the attack may be violated for large differences between $x^a_0 = 0$ and x_0. We refer the reader to (Teixeira *et al.*, 2012b) for a detailed analysis of the effects of zero initial conditions on zero-dynamics attacks.

If the zero is stable, that is $|\nu| < 1$, the attack will asymptotically decay to zero, thus having little effect on the plant. However, in the case of unstable zeros the attack grows geometrically, which could cause a great damage to the process. This statement is captured in the following result.

Theorem 3.5.4. *A zero-dynamics attack with $|\nu| > 1$ leads the system to an unsafe state if and only if span(x_0) is not contained in $\mathcal{S}_x$.*

Proof. Follows directly from Lemma 3.5.3 and from the fact that the zero-attack with $|\nu| > 1$ generates an unstable state trajectory moving away from the origin along $\mathrm{span}(x_0)$. □

Disclosure resources: This attack scenario considers an open-loop attack policy and so no disclosure capabilities are required, resulting in $\mathcal{R}^u = \mathcal{R}^y = \emptyset$ and $\mathcal{I}_k^u = \mathcal{I}_k^y = \emptyset\,\forall k$.

Disruption resources: The disruption capabilities in this attack scenario correspond to the ability of performing deception attacks on the actuator data channels. Therefore the required resources are $\mathcal{R}_I^u = \{1, \ldots, n_u\}$, $\mathcal{R}_I^y = \emptyset$, and $F = 0$

Model knowledge: The ability to compute the open-loop attack policy requires perfect knowledge of the zero-dynamics, which we denote as $\mathcal{K}_z$. Moreover, the zero-dynamics can be computed from the plant dynamics, namely A, B, and C. No knowledge of the feedback controller or anomaly detector is assumed in this scenario.

Although the former analysis considers LTI systems, the concept of zero-dynamics has been extended to other classes of system, e.g., nonlinear systems (Isidori, 1995). Hence zero-dynamics attacks could be directly extended to other classes of system in the noiseless case. In the presence of noise however, the interplay between the zero-dynamics and the noise inputs is not trivial and requires further analysis.

3.5.4 Local Zero-Dynamics Attack

In the previous scenario the zero-dynamics attack was characterized in terms of the entire system. Here we further restrict the adversary resources by considering that the adversary has disruption resources and knows the model of only a subset of the system. In particular, we rewrite the plant dynamics (3.14) as

$$\begin{aligned}
\begin{bmatrix} x_{k+1}^1 \\ x_{k+1}^2 \end{bmatrix} &= \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \begin{bmatrix} x_k^1 \\ x_k^2 \end{bmatrix} + \begin{bmatrix} B_1 \\ 0 \end{bmatrix} a_k \\
\tilde{y}_k^a &= \begin{bmatrix} C_1 & C_2 \end{bmatrix} \begin{bmatrix} x_k^1 \\ x_k^2 \end{bmatrix}
\end{aligned}$$

with $a_k \in \mathbb{R}^{n_u^1}$ and assume the adversary has access to only A_{11}, A_{21}, B_1, and C_1. From the adversary's view, this local system is characterized by

$$\begin{aligned}
x_{k+1}^1 &= A_{11} x_k^1 + B_1 a_k + A_{12} x_k^2 \\
y_k^l &= \begin{bmatrix} C_1 \\ A_{21} \end{bmatrix} x_k^1,
\end{aligned}$$

where y_k^l encodes the measurements depending on the local state, $C_1 x_k^1$, and the interaction between the local subsystem and the remaining subsystems, $A_{21} x_k^1$.

Attack policy: Similar to the zero-dynamics attack, the attack policy is given by the sequence $a_k = \nu^k g$, where g is the input zero direction for the chosen zero ν. The input zero direction can be obtained by solving

$$\begin{bmatrix} \nu I - A_{11} & -B_1 \\ C_1 & 0 \\ A_{21} & 0 \end{bmatrix} \begin{bmatrix} x_0^1 \\ g^1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}.$$

Note that the zero-dynamics parameterized by g^1 and ν correspond to local zero-dynamics of the global system.

Attack performance: A similar discussion as for the global zero-dynamics attack applies to this scenario. In particular, the stealthiness of the local zero-dynamics attack may be violated for large differences between x_0^1 and 0. Additionally, as stated in Theorem 3.5.4, attacks associated with unstable zeros yielding $|\nu| > 1$ are more dangerous and may lead the system to an unsafe state.

Disclosure resources: This attack scenario considers an open-loop attack policy and so no disclosure capabilities are required, resulting in $\mathcal{R}^u = \mathcal{R}^y = \emptyset$ and $\mathcal{I}_k^u = \mathcal{I}_k^y = \emptyset \; \forall k$.

Disruption resources: The disruption capabilities in this attack scenario correspond to the ability of performing deception attacks on the actuator data channels of the local subsystem. Therefore the required resources are $\mathcal{R}_I^u = \{1, \dots, n_u^1\}$, $\mathcal{R}_I^y = \emptyset$, and $F = 0$.

Model knowledge: The open-loop attack policy requires the perfect knowledge of the local zero-dynamics, denoted as $\tilde{\mathcal{K}}_z$ and obtained from A_{11}, B_1, C_1, and A_{21}.

3.5.5 Bias Injection Attack

Here a particular scenario of false-data injection is considered, where the adversary's goal is to inject a constant bias in the system without being detected. Furthermore, the bias is computed so that the impact at steady-state is maximized.

Attack policy: The bias injection attack is illustrated in Figure 3.5. The attack policy is composed of a steady-state component, the desired bias denoted as a_∞, and a transient component. For the transient, we consider that the adversary uses a low-pass filter so that the data corruptions are slowly converging to the steady-state values. As an example, for a set of identical first-order filters the open-loop attack sequence is described by

$$a_{k+1} = \beta a_k + (1 - \beta) a_\infty^*, \tag{3.16}$$

where $a_0 = 0$ and $0 < \beta < 1$ is chosen to ensure that the attack is δ-stealthy during the transient regime. The steady-state attack policy yielding the maximum impact on the physical system is described below, where the computation of a_∞ is summarized in Theorem 3.5.7 and Theorem 3.5.8.

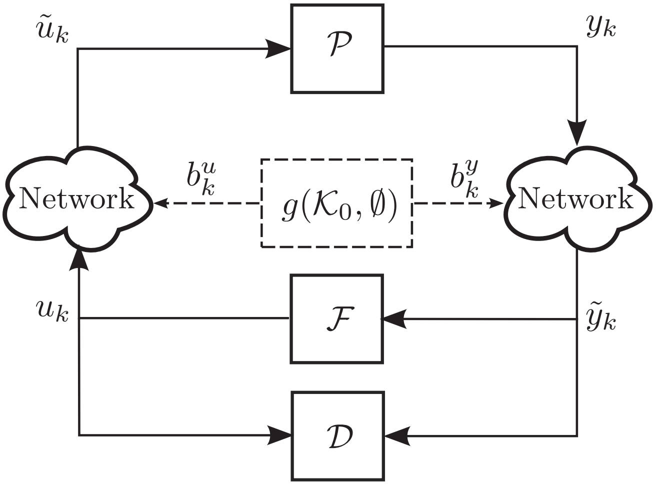


Figure 3.5: Schematic of the bias injection attack.

Attack performance: First the steady-state policy is considered. Denote a_∞ as the bias to be injected and recall the anomaly detector dynamics under attack (3.6). The steady-state detectability of the attack is then dependent on the steady-state value of the residual

$$r_\infty^a = \left(\mathbf{C}_e(I - \mathbf{A}_e)^{-1}\mathbf{B}_e + \mathbf{D}_e\right) a_\infty =: G_{ra}a_\infty.$$

Consider the set $\mathcal{U}_{[0,\,\infty]}^a = \left\{\mathbf{r}_{[0,\,\infty]}^a : \|r_k^a\|_2 \leq \delta,\, \forall k \geq 0\right\}$ and recall Definition 3.4.3 for δ-stealthy attacks. A necessary condition for the bias injection attack to be δ-stealthy is

$$\|G_{ra}a_\infty\|_2 \leq \delta. \tag{3.17}$$

Although attacks satisfying (3.17) could be detected during the transient, incipient attack signals slowly converging to a_∞ may go undetected. In fact, sufficient conditions for the bias attack to be α-stealthy are given in Theorem 3.5.9 and the results are illustrated through experiments in Section 3.6.

The impact of such attacks can be evaluated using the closed-loop dynamics under attack given by (3.5). Recalling that $\eta_k^a = [x_k^{a^\top} \quad z_k^{a^\top}]^\top$, the steady-state impact on the state is given by

$$x_\infty^a = [I \quad 0]\,(I - \mathbf{A})^{-1}\,\mathbf{B}a_\infty =: G_{xa}a_\infty.$$

Consider the following safe set defined in terms of x_k^a.

Definition 3.5.1. *The* $2-$*norm safe set* $\mathcal{S}^2_{x^a}$ *is defined as*

$$\mathcal{S}^2_{x^a} = \left\{ x \in \mathbb{R}^{n_x} : \|x\|_2^2 \leq 1 \right\},$$

and the system is said to be in a safe state if $x^a_k \in \mathcal{S}^2_{x^a}$*.*

For the $2-$norm safe set $\mathcal{S}^2_{x^a}$, the most dangerous bias injection attack corresponds to the δ-stealthy attack yielding the largest bias in the $2-$norm sense, which can be computed by solving

$$\begin{aligned} \underset{a_\infty}{\text{maximize}} \quad & \|G_{xa} a_\infty\|_2^2 \\ \text{subject to} \quad & \|G_{ra} a_\infty\|_2^2 \leq \delta^2. \end{aligned} \tag{3.18}$$

Lemma 3.5.5. *The optimization problem* (3.18) *is bounded if and only if*

$$\operatorname{Ker}(G_{ra}) \subseteq \operatorname{Ker}(G_{xa}).$$

Proof. Suppose that $\operatorname{Ker}(G_{ra}) \neq \emptyset$ and consider the subset of solutions where $a_\infty \in \operatorname{Ker}(G_{ra})$. For this subset of solutions, the optimization problem then becomes

$$\underset{a_\infty \in \operatorname{Ker}(G_{ra})}{\text{maximize}} \quad \|G_{xa} a_\infty\|_2^2 .$$

Since the objective function does not have an upper-bound and the feasible set is unbounded, the optimal value is unbounded unless $G_{xa}a_\infty = 0$ for all $a_\infty \in \operatorname{Ker}(G_{ra})$ i.e., $\operatorname{Ker}(G_{ra}) \subseteq \operatorname{Ker}(G_{xa})$. The proof is completed by noting that the feasible set and the objective function are bounded for all solutions $a_\infty \notin \operatorname{Ker}(G_{ra})$. □

Based on Lemma 3.5.5, below we consider the non-trivial case for which it holds that $\operatorname{Ker}(G_{ra}) \subseteq \operatorname{Ker}(G_{xa})$. The above optimization problem can be transformed into a generalized eigenvalue problem and the corresponding optimal solution is characterized in terms of generalized eigenvalues and eigenvectors. Before formalizing this statement, we introduce the following result:

Lemma 3.5.6. *Let* $Q \in \mathbb{R}^{n\times n}$ *and* $P \in \mathbb{R}^{n\times n}$ *be positive semi-definite matrices satisfying* $\operatorname{Ker}(Q) \subseteq \operatorname{Ker}(P)$ *and define the matrix pencil* $(P, Q) \triangleq (P - \nu Q)$*, with* $\nu \in \mathbb{C}$*. Denote* λ^* *as the largest generalized eigenvalue of the matrix pencil* (P, Q) *and* v^* *as the corresponding eigenvector. Then the matrix* $P - \lambda Q$ *is negative semi-definite for a generalized eigenvalue* λ *if and only if* $\lambda = \lambda^*$*. Moreover, we have* $\lambda^* \geq 0$ *and* $x^\top(P - \lambda^* Q)x = 0$ *with* $Qx \neq 0$ *if and only if* $x \in$ *span*(v^*)*.*

Proof. Define normalrank(P, Q) as the rank of $P - \nu Q$ for almost all values of $\nu \in \mathbb{C}$ and recall that λ is a generalized eigenvalue of (P, Q) if rank$(P - \lambda Q) <$ normalrank(P, Q). Furthermore, denote v as the generalized eigenvector associated with λ for which $(P - \lambda Q)v = 0$ with $v \notin \operatorname{Ker}(Q)$.

Define $T = [V_{\bar{N}}\, V_N] \in \mathbb{R}^{n\times n}$ where the columns of V_N are a basis for $\operatorname{Ker}(Q)$ and $V_{\bar{N}}$ is chosen such that T is nonsingular. Given that $\operatorname{Ker}(Q) \subseteq \operatorname{Ker}(P)$, the coordinate transformation induced by T leads to

$$T(P-\lambda Q)T^{-1} = \begin{bmatrix} \tilde{P}-\lambda\tilde{Q} & 0 \\ 0 & 0 \end{bmatrix},$$

where $\tilde{Q} \succ 0$ and $\tilde{P} \succeq 0$ and we conclude that $P-\lambda Q \preceq 0$ if and only if $\tilde{P}-\lambda\tilde{Q} \preceq 0$. Additionally, we see that all the non-zero generalized eigenvalues of (P, Q) need to reduce the rank of $\tilde{P}-\lambda\tilde{Q}$ and thus need to be positive. Hence we have proved that all generalized eigenvalues are non-negative and that $\lambda^* \geq 0$.

Now we show that $\tilde{P}-\lambda\tilde{Q}$ is indefinite for all generalized eigenvalues $0 < \lambda < \lambda^*$. Let $\bar{\lambda} > 0$ be a generalized eigenvalue of $(\tilde{P}, \tilde{Q})$ with the associated eigenvector $\bar{v}$. Then $\bar{v}^\top(\tilde{P}-\lambda\tilde{Q})\bar{v} = (\bar{\lambda}-\lambda)\bar{v}^\top\tilde{Q}\bar{v}$, which can be made positive or negative for all generalized eigenvalues $\lambda \in (0,\, \lambda^*)$ and thus our assertion is proved.

As the next step, we show that $\tilde{P}-\lambda^*\tilde{Q} \preceq 0$. Since $\tilde{Q}$ is invertible, the generalized eigenvalues of $(\tilde{P}, \tilde{Q})$ correspond to the eigenvalues of the positive semi-definite matrix $M\tilde{P}M$ with $M = \tilde{Q}^{-1/2}$. Furthermore note that $\tilde{P}-\lambda^*\tilde{Q} \preceq 0$ is equivalent to having $M\tilde{P}M - \lambda^* I \preceq 0$, which holds since $M\tilde{P}M$ is positive semi-definite with λ^* as the largest eigenvalue.

Finally, we show that $x^\top(P-\lambda^* Q)x = 0$ with $Qx \neq 0$ if and only if $x \in \operatorname{span}(v^*)$. Given the condition $Qx \neq 0$, it is enough to verify that $x^\top(\tilde{P}-\lambda^*\tilde{Q})x = 0$ for $x \neq 0$ if and only if $x \in \operatorname{span}(\tilde{v}^*)$, where $\tilde{v}^*$ is the generalized eigenvector of $(\tilde{P}, \tilde{Q})$ associated with λ^*. The proof concludes by recalling that $\tilde{P}-\lambda^*\tilde{Q} \preceq 0$, hence $x^\top(\tilde{P}-\lambda^*\tilde{Q})x = 0$ if and only if x belongs to the subspace spanned by the eigenvectors associated with λ^*. □

The optimal bias injection attack in the sense of (3.18) is characterized by the following result:

Theorem 3.5.7. *Consider the* $2-$*norm safe set* $\mathcal{S}^2_{x^a}$ *and the corresponding optimal* δ*-stealthy bias injection attack parameterized by* (3.18)*, which is assumed to be bounded. Denote* λ^* *and* v^* *as the largest generalized eigenvalue and corresponding unit-norm eigenvector of the matrix pencil* $(G_{xa}^\top G_{xa},\, G_{ra}^\top G_{ra})$*. The optimal bias injection attack is given by*

$$a^*_\infty = \pm\frac{\delta}{\|G_{ra}v^*\|_2}v^*,$$

and the corresponding optimal value is $\|G_{xa}a_\infty\|_2^2 = \lambda^*\delta^2$*. Moreover, at steady-state the system is in a safe state if and only if* $\lambda^*\delta^2 \leq 1$*.*

Proof. Let $P,\, Q \in \mathbb{R}^{n\times n}$ be positive semi-definite matrices such that $\operatorname{Ker}(Q) \subseteq \operatorname{Ker}(P)$. Recall that λ is a generalized eigenvalue of (P, Q) if $\operatorname{rank}(P-\lambda Q) < \operatorname{normalrank}(P, Q)$, where $\operatorname{normalrank}(P, Q)$ is defined as the rank of $P-\nu Q$ for almost all values of $\nu \in \mathbb{C}$. Furthermore, denote v as the generalized eigenvector associated with λ for which $(P - \lambda Q)v = 0$ with $v \notin \operatorname{Ker}(Q)$. The necessary and sufficient conditions for the optimization problem (3.18) are given by (Hiriart-Urruty, 2001)

$$\begin{aligned}
(i) \quad 0 &= (G_{xa}^\top G_{xa} - \lambda^* G_{ra}^\top G_{ra}) a_\infty^*, \\
(ii) \quad 0 &= a_\infty^{*\top} G_{ra}^\top G_{ra} a_\infty^* - \delta^2, \\
(iii) \quad 0 &\geq y^\top (G_{xa}^\top G_{xa} - \lambda^* G_{ra}^\top G_{ra}) y, \ \forall\, y \neq 0.
\end{aligned}$$

Suppose λ^* is the largest generalized eigenvalue of $(G_{xa}^\top G_{xa},\, G_{ra}^\top G_{ra})$ and let v^* be the corresponding eigenvector. Scaling v^* by κ so that $a_\infty^* = \kappa v^*$ satisfies $\|G_{ra} a_\infty^*\|_2^2 = \delta^2$ leads to $\kappa = \pm \frac{\delta}{\|G_{ra} v^*\|_2}$, and conditions (i) and (ii) are satisfied. As for condition (iii), note that $G_{xa}^\top G_{xa} - \lambda^* G_{ra}^\top G_{ra}$ is negative semi-definite by Lemma 3.5.6, given that λ^* is the largest generalized eigenvalue, $G_{xa}^\top G_{xa}$ and $G_{ra}^\top G_{ra}$ are positive semi-definite, and the assumption that $\operatorname{Ker}(G_{ra}) \subseteq \operatorname{Ker}(G_{xa})$. To conclude the proof, observe that the optimal value is given by $a_\infty^{*\top} G_{xa}^\top G_{xa} a_\infty^* = \lambda^* a_\infty^{*\top} G_{ra}^\top G_{ra} a_\infty^* = \lambda^* \delta^2 = \|x_\infty^a\|_2^2$ and thus, by definition, $x_\infty^a \in \mathcal{S}_{x^a}^2$ if and only if $\lambda^* \delta^2 \leq 1$. □

More generally, the optimal bias injection attacks for ellipsoidal safe sets

$$\mathcal{S}_{x^a} = \left\{ x^a \in \mathbb{R}^{n_x} : {x^a}^\top P x^a \leq 1 \right\},$$

with P positive definite, can be found by replacing the objective function in (3.18) by $\|P^{1/2} G_{xa} a_\infty\|_2^2$.

In the following, the steady-state attack policy is derived for the following safe set:

Definition 3.5.2. *The infinity-norm safe set $\mathcal{S}_{x^a}^\infty$ is defined as*

$$\mathcal{S}_{x^a}^\infty = \left\{ x \in \mathbb{R}^{n_x} : \|x\|_\infty \leq 1 \right\},$$

and the system is said to be in a safe state if $x_k^a \in \mathcal{S}_{x^a}^\infty$.

Given the infinity-norm safe set $\mathcal{S}_{x^a}^\infty$, the bias injection attack with the largest impact corresponds to the δ-stealthy attack yielding the largest bias in the infinity-norm sense. This attack can be obtained by solving the following optimization problem

$$\begin{aligned}
\underset{a_\infty}{\text{maximize}} \quad & \|G_{xa} a_\infty\|_\infty \\
\text{subject to} \quad & \|G_{ra} a_\infty\|_2 \leq \delta.
\end{aligned} \tag{3.19}$$

A possible method to solve this problem is to observe that

$$\|G_{xa} a_\infty\|_\infty = \underset{i}{\text{maximize}} \ \|e_i^\top G_{xa} a_\infty\|_2,$$

where the vector e_i is $i-$th column of the identity matrix. Thus one can transform the optimization problem (3.19) into a set of problems with the same structure as (3.18), obtaining

$$\begin{array}{lll} \underset{i}{\text{maximize}} & \underset{a_\infty}{\text{maximize}} & \left\|e_i^\top G_{xa} a_\infty\right\|_2 \\ & \text{subject to} & \|G_{ra} a_\infty\|_2 \leq \delta. \end{array} \tag{3.20}$$

Theorem 3.5.8. *Consider the infinity-norm safe set $\mathcal{S}_{x^a}^\infty$ and the corresponding optimal δ-stealthy bias injection attack parameterized by the optimization problem* (3.19)*, which is assumed to be bounded. Let e_i be the $i-$th column of the identity matrix and denote λ_i^* and v_i^* as the largest generalized eigenvalue and corresponding unit-norm eigenvector of the matrix pencil $G_{xa}^\top e_i e_i^\top G_{xa} - \lambda G_{ra}^\top G_{ra}$. Letting $\lambda^* = \max_i \lambda_i^*$, with v^* as the corresponding generalized eigenvector, the optimal bias attack is given by*

$$a_\infty^* = \pm \frac{\delta}{\|G_{ra} v^*\|_2} v^*,$$

and the corresponding optimal value is $\|G_{xa} a_\infty\|_\infty = \sqrt{\lambda^*}\delta$. Moreover, at steady-state the system is in a safe state if and only if $\lambda^* \delta^2 \leq 1$.

Proof. The proof follows directly from considering the set of optimization problems in (3.20) and applying Theorem 3.5.7. □

Recall that the steady-state value of the data corruption a_∞^* is not sufficient for the attack to be δ-stealthy, since the transients are disregarded. In practice, however, it has been observed in the fault diagnosis literature that faults with slow dynamics, also known as incipient faults, are difficult to distinguish from model uncertainty and noise (Chen and Patton, 1999; Zhang *et al.*, 2002). Therefore the low-pass filter dynamics in the attack policy (3.16) could be designed sufficiently slow as to make detection more difficult. Below we provide sufficient conditions under which a given filter parameter β renders the bias attack δ-stealthy with respect to $\mathcal{U}_{[0,\,\infty]}^a = \left\{\mathbf{r}_{[0,\,\infty]}^a : \|r_k^a\|_2 \leq \delta,\ \forall k \geq 0\right\}$.

Theorem 3.5.9. *Consider the attack policy $a_{k+1} = \beta a_k + (1-\beta) a_\infty^*$ with $\beta \in (0,\,1)$. The residual r_k^a is characterized as the output of the autonomous system*

$$\begin{aligned} \psi_{k+1}^a &= \bar{A} \psi_k^a \\ r_k^a &= \bar{C} \psi_k^a \end{aligned}$$

with

$$\bar{A} = \begin{bmatrix} \mathbf{A}_e & \mathbf{B}_e & 0 \\ 0 & \beta I & (1-\beta) I \\ 0 & 0 & I \end{bmatrix}, \quad \psi_0^a = \begin{bmatrix} 0 \\ 0 \\ a_\infty^* \end{bmatrix},$$

$$\bar{C} = \begin{bmatrix} \mathbf{C}_e & \mathbf{D}_e & 0 \end{bmatrix}.$$

Moreover, the attack policy is δ-stealthy for a given β if the following optimization problem admits a solution

$$\begin{aligned} \underset{\gamma,P}{\textit{minimize}} \quad & \gamma \\ \textit{subject to} \quad & \gamma \leq \delta^2, \\ & P \succ 0, \\ & {\psi_0^a}^\top P \psi_0^a \leq 1, \\ & \begin{bmatrix} P & \bar{C}^\top \\ \bar{C} & \gamma I \end{bmatrix} \succeq 0, \\ & \bar{A}^\top P \bar{A} - P \prec 0. \end{aligned}$$

Proof. The autonomous system is directly obtained by considering the augmented state $\psi^a = [{\xi^a_{k|k}}^\top \; a_k^\top \; \upsilon_k^\top]^\top$, where the attack vector a_k corresponds to the state of the low-pass filter bank (3.16) and υ_k the integral state initialized at $\upsilon_0 = a_\infty$. Given this autonomous system, one observes that the attack is δ-stealthy if and only if the corresponding output-peak $\|r_k^a\|_2^2$ is bounded by δ^2 for all $k \geq 0$, given the initial condition parameterized by a_∞^*. The remainder of the proof follows directly from the results in Boyd *et al.* (1994) regarding output-peak bounds for autonomous systems. □

Disclosure resources: Similarly to the zero attack, no disclosure capabilities are required for this attack, since the attack policy is open-loop. Therefore we have $\mathcal{R}^u = \mathcal{R}^y = \emptyset$ and $\mathcal{I}_k^u = \mathcal{I}_k^u = \emptyset$ for all k.

Disruption resources: The biases may be added to both the actuator and sensor data, hence the required resources are $\mathcal{R}_I^u \subseteq \{1, \ldots, n_u\}$, $\mathcal{R}_I^y \subseteq \{1, \ldots, n_y\}$. Since no physical attack is performed, we have $F = 0$.

Model knowledge: As seen in (3.18), the open-loop attack policy (3.16) requires the knowledge of the closed-loop system and anomaly detector steady-state gains G_{ra} and G_{xa}, which we denoted as $\mathcal{K}_0$ as shown in Figure 3.5.

3.6 Experiments

In this section, we report experiments on the two testbeds for electric power networks and process control, respectively, which are described in Chapter 2. Several cyber attacks were staged on the testbeds, according to the different scenarios characterized in the previous section.

3.6.1 Electric Power Systems

Next, we present the results obtained by carrying out a stealthy deception attack on a SCADA EMS software. Before analyzing the results, we briefly describe the experimental setup.

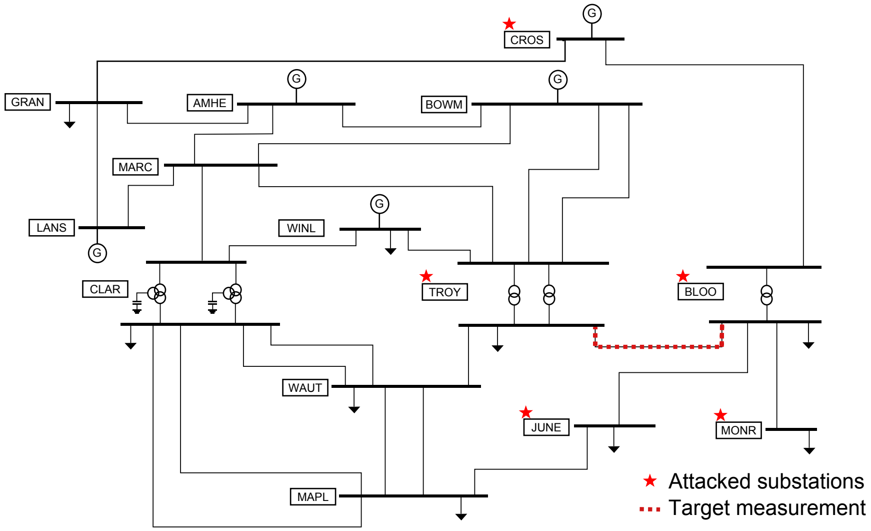


Figure 3.6: Power network considered in the experiment of Section 3.6.1.

Experimental Setup

The EMS software was supplied with the test network presented in Figure 3.6. This network consists of 14 substations and the bus-branch model has 27 buses and 40 branches. Several measurements are available at each substation, which can be corrupted by the adversary.

Specific EMS components, such as the SE and BDD, are configured with unitary weights for all the measurements. As presented in Section 2.5.1, the SE solves the nonlinear weighted least-squares problem, while the BDD algorithm triggers an alarm if the residue norm exceeds a certain threshold.

As described in previous sections, some information about the system is needed to compute stealthy deception attacks. Here we consider a particular class of such information, namely the bus-branch model of the network described in Section 2.5.1. In this experiment, we exported this information to MATLAB using the MATPOWER toolbox, (Zimmerman *et al.*, 2009). A simplified attack was considered, in which only the DC model $y = C_{DC}x$, described in Section 2.5.1, was used to compute the attack.

Attack Scenario

To conduct our experiment we considered measurement number 33, corresponding to the active power flow on the tie-line between TROY and BLOO substations, to be the target measurement that the attacker desires to corrupt. In order to do so without being detected, the attacker needs to perform a coordinated attack by

Table 3.1: Example: adding 100MW to target measurement 33

Measurement index, k	Normalized attack, $\bar{a}_k$	Correct value (MW), y_k^a	False value (MW), $\tilde{y}_k^a$
4	-1	1005.7041	905.7042
21	-0.7774	157.8541	80.1103
24	0.9665	507.7171	604.3638
27	2.7439	40.0006	314.3911
33	**1**	**–14.7971**	**85.2029**
62	0.7774	-123.3764	-45.6327
104	-0.9665	-334.8826	-431.5293

corrupting the value of other power measurements.

Following the bias injection attack presented in Section 3.5.5, the attack vector a corresponds to sensor data corruption and is computed by solving the optimization problem (3.20). Since there are no dynamics and measurement 33 is the only target, we let $G_{xa} = e_{33}^\top$, so that $G_{xa}a = a_{33}$. Recall from Section 2.5.1 that, given the DC model C_{DC}, the residue under sensor data corruption is given by $r^a = G_{ra}a$, where $G_{ra} = I - C_{DC}(C_{DC}^\top C_{DC})^{-1}C_{DC}^\top$. Moreover, we consider the threshold $\delta = 0$, which constrains the attack vector a to be computed so that $r^a = 0$.

Note that the null-space of G_{ra} corresponds to the range-space of C_{DC}, yielding $G_{ra}C_{DC} = 0$. Moreover, we have that $G_{xa}C_{DC} = e_{33}^\top C_{DC}$ is not identically zero, since it corresponds to the 33-rd row of the measurement matrix C_{DC}. Therefore, we conclude that the null-space of G_{ra} is not contained in the null-space of G_{xa}, in which case the attack is unbounded, as stated in Lemma 3.5.5.

Instead of the unbounded optimization problem (3.20), we look at the feasibility problem of computing a normalized attack vector $\bar{a}$ satisfying the equality constraints $G_{ra}\bar{a} = 0$ and $G_{xa}\bar{a} = 1$. Note that all solutions to first constraint can be parameterized as $\bar{a} = C_{DC}\tilde{x}$, for some vector $\tilde{x}$. Hence, the feasibility problem is equivalent to the undetermined set of equations $G_{xa}C_{DC}\tilde{x} = 1$, which admits numerous attack vectors as feasible solutions. To narrow the attack vector candidates, we search for the sparsest candidate by considering the cost function $\|\bar{a}\|_0$ and solving the combinatorial optimization problem

$$\begin{aligned} \underset{\bar{a}}{\text{minimize}} \quad & \|\bar{a}\|_0 \\ \text{subject to} \quad & G_{ra}\bar{a} = 0, \\ & G_{xa}\bar{a} = 1, \end{aligned}$$

where $\|\bar{a}\|_0$ denotes the number of non-zero elements in the vector $\bar{a}$. Later, in Chapter 4, we connect the previous optimization problem to the resources and likelihood of the attack and mention efficient algorithms to solve it.

Solving the latter optimization problem retrieves the normalized attack vector $\bar{a}$ with highest sparsity that stealthily changes the target measurement by 1 MW, as imposed by the constraint $G_{xa}\bar{a} = 1$. Additionally, the normalized attack vector $\bar{a}$, presented in Table 3.1, can be scaled to inject other biases. For instance, in Table 3.1 we can see the correct value of the compromised measurements, denoted by y^a, and the false values sent to the control center, $\tilde{y}^a$, when the objective was to induce a bias of 100MW in the target measurement by having $\tilde{y}^a = y^a + 100\bar{a}$. Such an attack only corrupts 7 measurements in total, which are taken from 5 substations, namely TROY, BLOO, JUNE, MONR, and CROS, all situated in the right side of Figure 3.6. Hence we see that to stealthily attack a single measurement, a local coordinated attack suffices, even for such a large system. Additionally, as discussed in Dán and Sandberg (2010), note that usually all measurements within a given substation are gathered at a single RTU. This means that by breaking into the substation's RTU the attacker gains access to all those measurements, so we can argue that although 7 measurements need to be corrupted, only 5 RTUs need to be compromised.

Experimental Results

The normalized attack vector $\bar{a}$, whose non-zero entries are shown in Table 3.1, was used to corrupt the measurement data according to the attacker's objective. In Figure 3.7, we show the results obtained by performing stealthy deception attacks as described before and naive deception attacks where only the target measurement is compromised. In both cases, the bias in the target measurement was sequentially increased by 10MW at each step. From these results we see that the naive attack was undetected up to a bias of 20MW, while for bias above 30MW this attack was detected and the compromised measurement removed. The coordinated stealthy attack, however, remained undetected for all the bias values showed in the figure. Furthermore we see that the naive attack did not influence the estimate as much as the stealthy one. For stealthy attacks, the relationship between the false and the estimated values is an almost unitary slope, meaning that the operator would see the false values as being truthful.

Table 3.2 shows the results obtained for large bias, where the attacks were performed sequentially with steps of 50MW. We observe that the stealthy attacks were successful, with no BDD alarm triggered up to a bias of 150MW, beyond which the nonlinear SE (2.3) could not be solved.

Although the SE did not converge for attacks above 200MW, it is still surprising to see that attacks based on the linearized model as large as 150MW are successful. To better understand what such a quantity indicates, note that the nominal value of the targeted tie-line is 260MW. Thus the attack was able to induce a bias of more than 50% of the nominal value, which reveals that the SCADA EMS software is indeed sensitive to stealthy deception attacks. Furthermore, notice that the number of warnings given by the contingency analysis (CA) increase with the size of the attack. The increased number of CA warnings could lead the operator to take

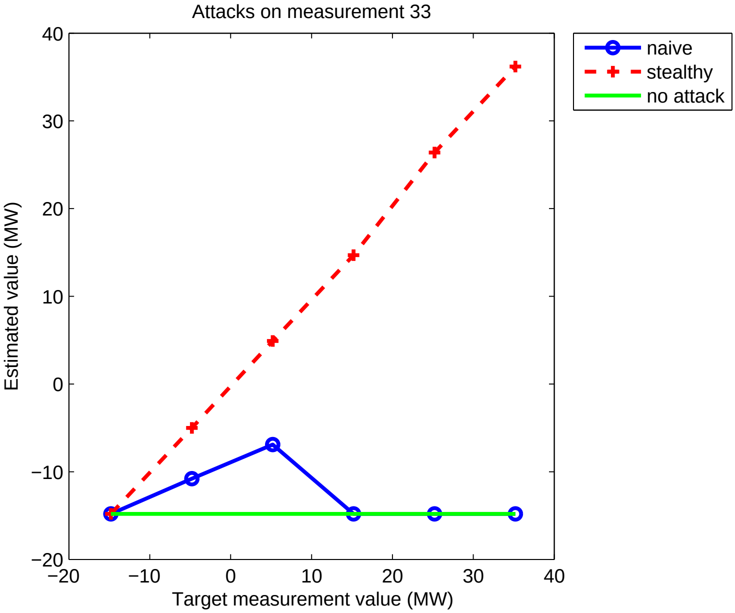


Figure 3.7: Stealthy deception attack.

Table 3.2: Results from the stealthy attack for large bias

Target bias,	False value	Estimate	#BDD	#CA
a_{33}	(MW), $\tilde{y}^a_{33}$	(MW), $\hat{y}^a_{33}$	Alarms	Alarms
0	-14.8	-14.8	0	2
50	35.2	36.2	0	2
100	85.2	86.7	0	10
150	135.2	137.5	0	27
200	185.2	—	—	—

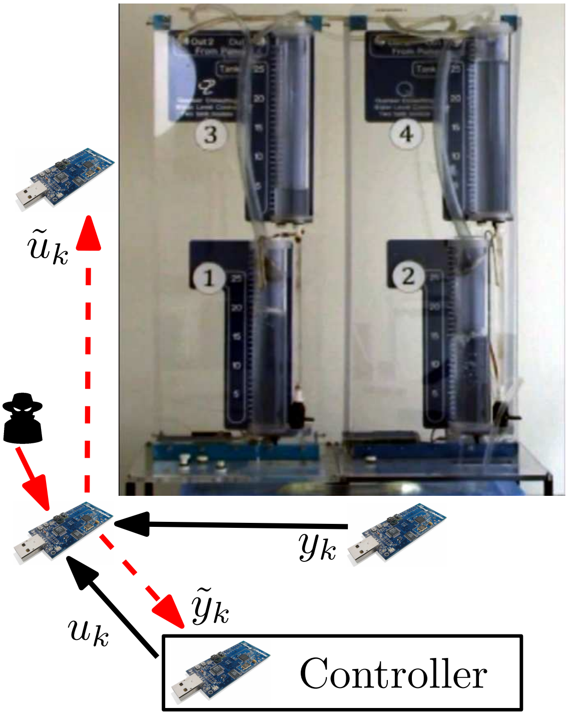


Figure 3.8: Schematic diagram of the testbed with the quadruple-tank process and a multi-hop communication network.

corrective actions. Therefore, we conclude that operators and utilities should care about these scenarios.

We also want to highlight that these results were achieved with a simplified linear model where several parameters, including the correct operating conditions and cross-coupling effects between active and reactive measurements, were disregarded. However, in these scenarios we assumed the attacker had a large amount of resources such as a rather detailed knowledge regarding the network model, the available measurements, and the pseudo-measurements, and access to several RTUs. Most likely, an attacker with such resources could find easier alternative attacks on the power network than the one considered in this section.

3.6.2 Networked Control System Testbed

In this subsection, we consider the process control testbed characterized in Section 2.5.2. The testbed consists of a quadruple-tank process (QTP) (Johansson, 2000) controlled through a wireless communication network, as described in Section 2.5.2 and depicted in Figure 3.8. The nonlinear plant model is linearized for a given operating point. Moreover, given the range of the water levels, the following safe set is considered

$$\mathcal{S}_x = \{x \in \mathbb{R}^{n_x} : \; \|x - \sigma \mathbf{1}\|_\infty \leq 15, \, \sigma = 15\},$$

where $\mathbf{1} \in \mathbb{R}^{n_x}$ is a vector with all entries set to 1.

The process is controlled using a centralized LQG controller with integral action running in a remote computer and a wireless network is used for the communications. A Kalman-filter-based anomaly detector is also running in the remote computer and alarms are triggered according to (3.4), for which we have considered

$$\begin{aligned}\mathcal{U}_{[k_0,\,\infty]} &= \left\{\mathbf{r}_{[k_0,\,\infty]} : \|r_k\|_2 \leq \delta + \delta_r, \; \forall k \geq k_0\right\},\\ \mathcal{U}^a_{[k_0,\,\infty]} &= \left\{\mathbf{r}^a_{[k_0,\,\infty]} : \|r^a_k\|_2 \leq \delta, \; \forall k \geq k_0\right\},\end{aligned}$$

with $\delta_r = 0.15$ and $\delta = 0.25$ for illustration purposes.

Denial-of-Service Attack

Here we consider the case where the QTP suffers a DoS attack on both sensors, while operating at a constant set-point. The state and residual trajectories from this experiment are presented in Figure 3.9. The DoS attack follows a Bernoulli model (Amin *et al.*, 2009) with $p = 0.9$ as the probability of packet loss and the last received data are used in the absence of data. From Proposition 3.5.1, we have that the closed-loop system under such a DoS attack is exponentially stable.

The DoS attack initiates at $t \approx 100\,s$, leading to an increase in the residual due to packet losses. However the residual remained below the threshold during the attack and there were no significant changes in the system's state.

Replay Attack

In this scenario, the QTP is operating at a constant set-point while a hacker desires to steal water from tank 4, the upper tank on the right side. An example of this attack is presented in Figure 3.10, where the replay attack policy is the one described in Section 3.5.2. The adversary starts by replaying past data from y_2 at $t \approx 90\,s$ and then begins stealing water from tank 4 at $t \approx 100\,s$. Tank 4 is successfully emptied and the attacks stops removing water at $t \approx 180\,s$. To ensure stealthiness, the replay attack continues until the system recovered its original setpoint at $t \approx 280\,s$. As we can see, the residue stays below the alarm threshold and therefore the attack is not detected.

Zero-Dynamics Attack

The QTP has a non-minimum phase configuration in which the plant possesses an unstable zero. In this case, as discussed in Section 3.5.3, an adversary able to

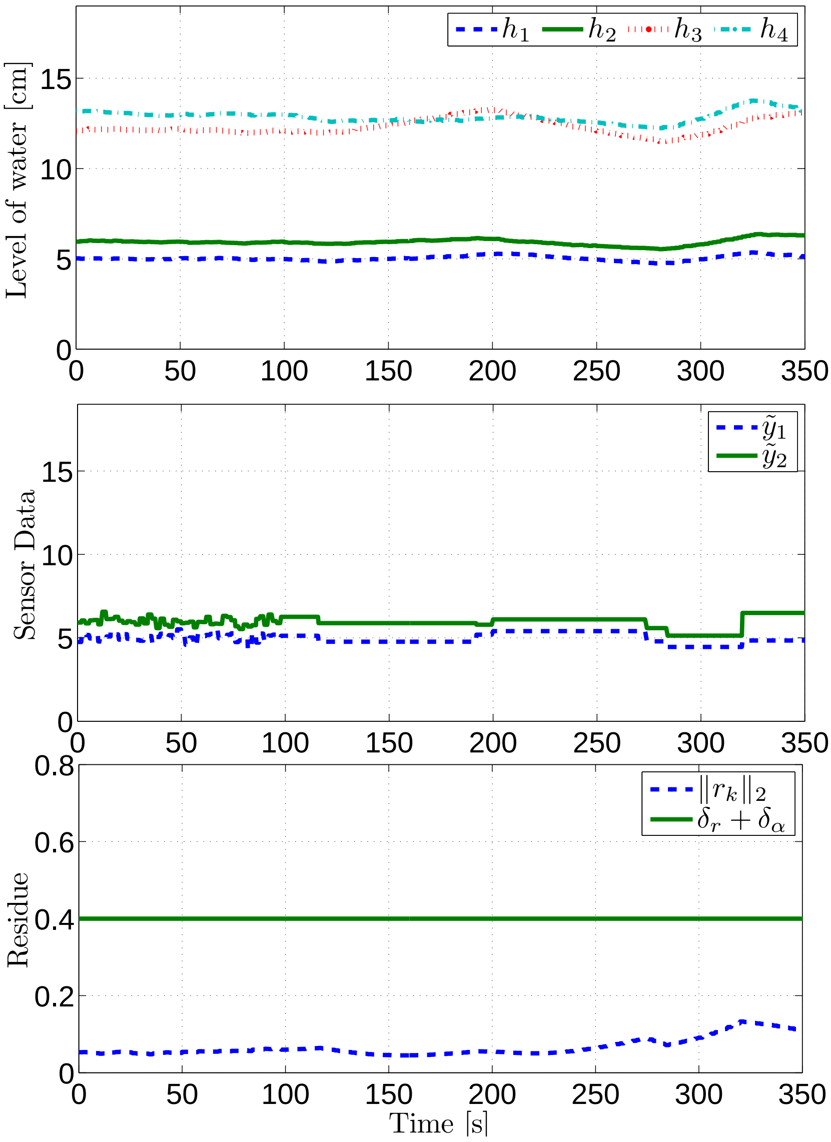


Figure 3.9: Results for the DoS attack performed against both sensors since $t \approx 100\,s$.

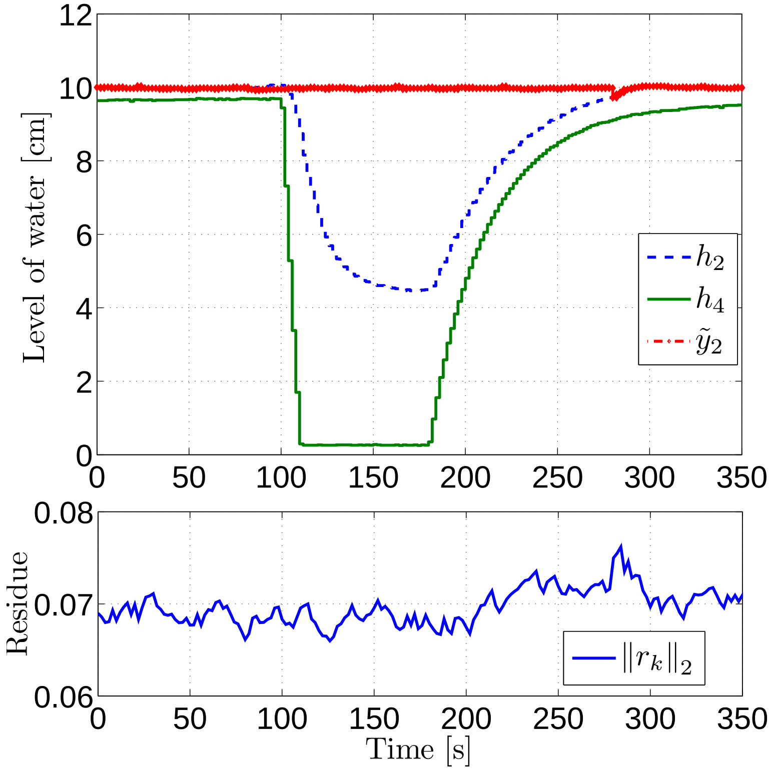


Figure 3.10: Results for the replay attack performed against sensor 2 from $t \approx 90\,s$ to $t \approx 280\,s$. Additionally, the adversary opens the tap of tank 4 at $t \approx 100\,s$ and closes it at $t \approx 180\,s$.

corrupt all the actuator channels may launch a false-data injection attack where the false-data follows the zero-dynamics. Moreover, since the safe region is described by the set $\mathcal{S}_x = \{x \in \mathbb{R}^{n_x} : \ \|x - \sigma\mathbf{1}\|_\infty \leq 15, \sigma = 15\}$, from Theorem 3.5.4 we expect that the zero-dynamics attack associated with the unstable zero can drive the system to an unsafe region. This scenario is illustrated in Figure 3.11.

The adversary's goal is to either empty or overflow at least one of the tanks, considered as an unsafe state. The attack on both actuators begins at $t \approx 30\,s$, causing a slight increase in the residual. Tank 3 becomes empty at $t \approx 55\,s$ and shortly after actuator $\tilde{u}_2$ saturates, producing a steep increase in the residual which then crosses the threshold. However, note that the residual was below the threshold when the unsafe state was reached.

After saturation of the water level and the actuators, the system dynamics change and therefore the attack signal no longer corresponds to the zero-dynamics and is detected, although it has already damaged the system. Thus these attacks are particularly dangerous in processes that have unstable zero-dynamics and in which the actuators are over-dimensioned, allowing the adversary to perform longer attacks before saturating.

Bias Injection Attack

The results for the case where u_1 and y_1 are respectively corrupted with b^u_∞ and b^y_∞ are presented in the Figure 3.12. In this scenario, the adversary aimed at driving the system out of the safe set $\mathcal{S}_x$ while remaining stealthy for $\delta = 0.25$. The bias was slowly injected using a first-order low-pass filter with $\beta = 0.95$ and the following steady-state value, computed using Theorem 3.5.8,

$$a_\infty = \begin{bmatrix} b^u_\infty \\ b^y_\infty \end{bmatrix} = \begin{bmatrix} 2.15 \\ -9.42 \end{bmatrix}.$$

The bias injection began at $t \approx 70\,s$ and led to an overflow in tank 4 at $t \approx 225\,s$. At that point, the adversary started removing the bias and the system recovered the original setpoint at $t \approx 350\,s$. The residual remained within the allowable bounds throughout the attack, thus the attack was not detected.

3.7 Summary

In this chapter, we have analyzed the security of networked control systems. An attack-scenario space based on the adversary's model knowledge, disclosure, and disruption resources was proposed and the corresponding adversary model described. Attack scenarios corresponding to DoS, replay, zero-dynamics, and bias injection attacks were analyzed using this framework. In particular, the maximum impact of stealthy bias injection attacks was derived and it was shown that the corresponding policy does not require perfect model knowledge. These attack scenarios were illustrated using experimental setups based on a SCADA EMS software for electric power networks and a quadruple-tank process controlled over a wireless network.

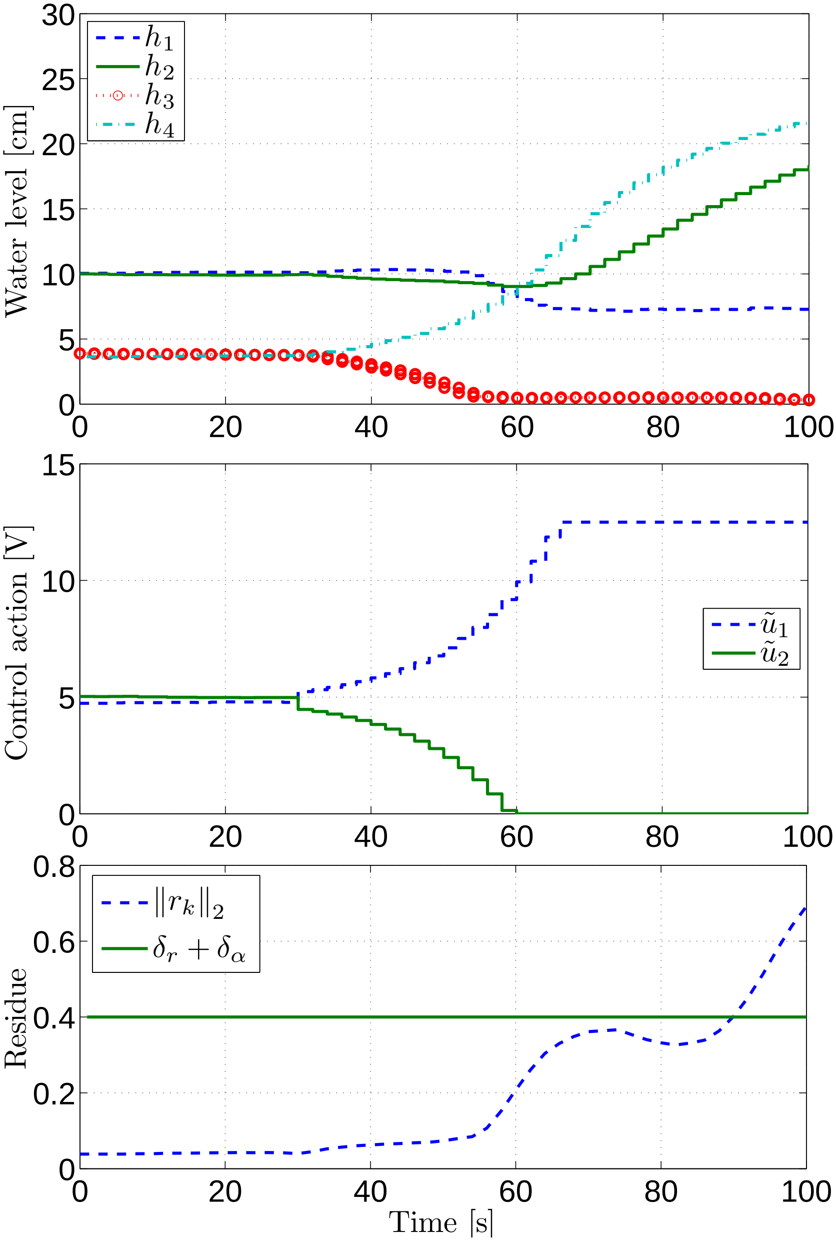


Figure 3.11: Results for the zero-dynamics attack starting at $t \approx 30\,s$. Tank 3 is emptied at $t \approx 55\,s$, resulting in a steep increase in the residual since the linearized model is no longer valid.

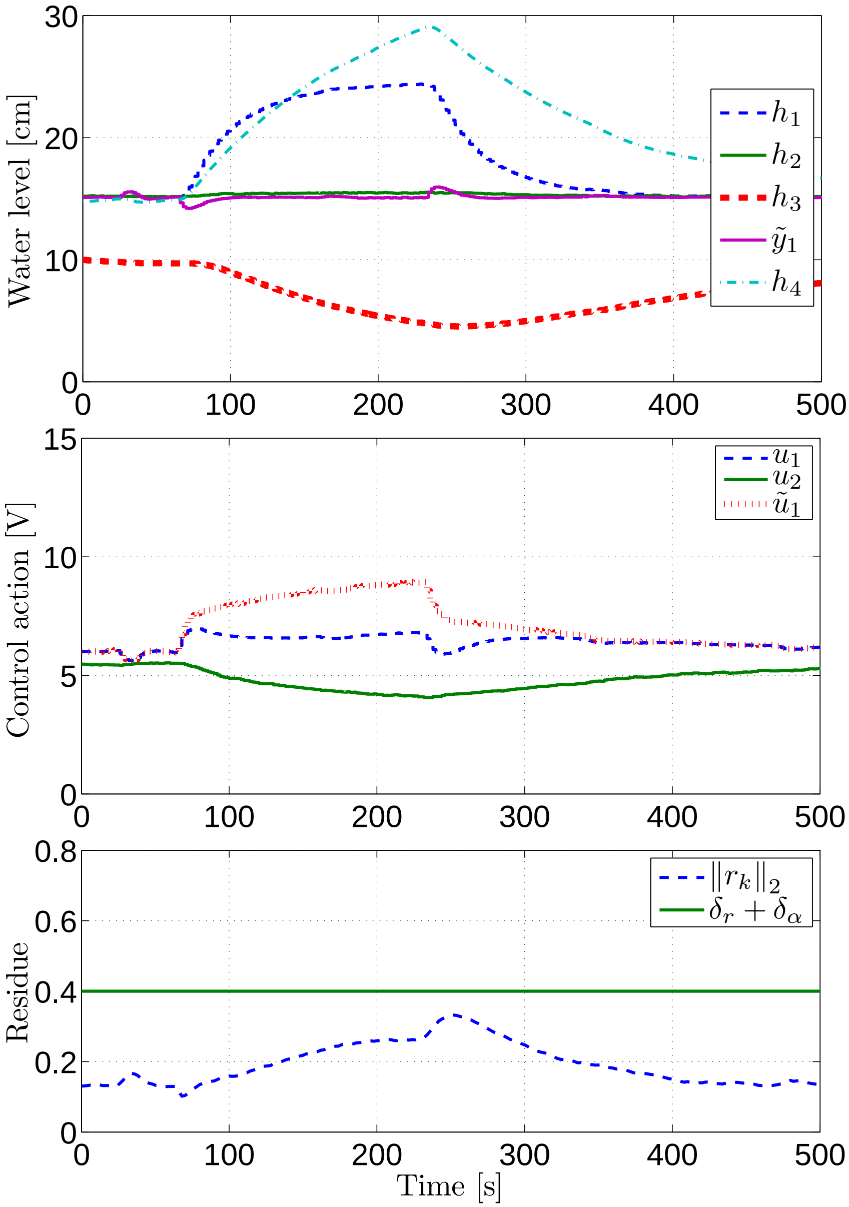


Figure 3.12: Results for the bias attack against the actuator 1 and sensor 1 in the minimum phase QTP. The attack is launched using a low-pass filter in the instant $t \approx 70\,s$ and stopped at $t \approx 230\,s$.

Chapter 4

Cyber Security Metrics for Networked Control Systems

The motivational examples in Chapter 1 and the attack scenarios described in Chapter 3 had a common theme: a knowledgable adversary aiming at disrupting the system in a covert way, without raising alarms. However, mitigating all threats within such class of attacks may be a difficult task, given the large amount of different scenarios that need to be considered. Motivated by this difficulty, in this chapter, we address resiliency of control systems under the perspective of risk management, where the notion of risk is defined in terms of a threat's scenario, impact, and likelihood. In particular, we consider attack scenarios with different sets of disruption resources and aim at developing tools to identify the scenarios yielding the highest impact, while using the least amount of resources.

Contributions and Related Work

Data deception attack is a particular type of a complex cyber attack where the attacker introduces corrupted data in the communication network. Several instances of this scenario have been considered in the context of control systems, see (Cárdenas *et al.*, 2011; Esfahani *et al.*, 2010; Sundaram and Hadjicostis, 2011) and references therein. In this chapter we address stealthy false-data injection attacks that are constructed so that they are not detected based on the control input and measurement data available to anomaly detectors. A sub-class of these attacks have been recently addressed from a system theoretic perspective. In (Smith, 2011) the author characterizes the set of attack policies for stealthy false-data injection attacks with detailed model knowledge and full access to all sensor and actuator channels, while (Pasqualetti *et al.*, 2011) described the set of stealthy false-data injection attacks for omniscient attackers with full-state information, but possibly compromising only a subset of the existing sensors and actuators. Similarly, Fawzi *et al.* (2012) consider a finite time-interval and characterizes the number of corrupted channels that cannot be detected during that time-interval. In the previous approaches, the attacks were constructed so that the system's output remains unchanged by the attack. Instead, we allow more freedom to the adversary and consider attacks that may be theoretically detectable, but are still stealthy since they do not trigger any alarm by the anomaly detector.

In this chapter, we consider the typical architecture for a networked control system under false-data injection attacks and adversary models presented in Chapter 3. Under this framework, various formulations for quantifying cyber security of control systems are proposed and formulated as constrained optimization problems. These formulations capture trade-offs in terms of impact on the control system, attack detectability, and adversarial resources. In particular, one of the formulations considers the minimum number of data channels that need to be corrupted so that the adversary remains stealthy, similarly to the security index for static systems proposed in Sandberg *et al.* (2010). The formulations are related to system theoretic concepts.

The outline of the chapter is as follows. The control system architecture and adversary model are described in Section 4.1. Section 4.2 discusses security metrics for static systems. Regarding dynamical systems, several formulations quantifying cyber security are introduced in Section 4.3 for a given time-horizon and in Section 4.4 for steady-state. Some particular metrics are posed as a mixed integer linear programs in Section 4.5. The security metrics and their application to mitigation risk are illustrated through numerical examples in Section 4.6, followed by conclusions in Section 4.7.

4.1 Problem Formulation

In this section, we recall the networked control system structure presented in Chapter 3 and describe the attack scenario and the main problem to be tackled.

For the networked control system, we consider four main components described in Chapter 3: the physical plant, the communication network, the feedback controller, and the anomaly detector. The physical plant is modeled in a discrete-time state-space form as

$$\mathcal{P} : \begin{cases} x_{k+1} = Ax_k + B\tilde{u}_k \\ y_k = Cx_k \end{cases},$$

where $x_k \in \mathbb{R}^{n_x}$ is the state variable, $\tilde{u}_k \in \mathbb{R}^{n_u}$ the control actions applied to the process, and $y_k \in \mathbb{R}^{n_y}$ the measurements from the sensors. The sensor measurements and actuator data are transmitted through a communication network, which at the plant side correspond to y_k and $\tilde{u}_k$, respectively. At the controller side we denote the sensor and actuator data by $\tilde{y}_k \in \mathbb{R}^{n_y}$ and $u_k \in \mathbb{R}^{n_u}$, respectively.

The feedback controller is described by

$$\mathcal{F} : \begin{cases} z_{k+1} = A_c z_k + B_c\tilde{y}_k \\ u_k = C_c z_k + D_c\tilde{y}_k \end{cases},$$

where the state of the controller is $z_k \in \mathbb{R}^{n_z}$, while the anomaly detector is given by

$$\mathcal{D} : \begin{cases} s_{k+1} = A_e s_k + B_e u_k + K_e \tilde{y}_k \\ r_k = C_e s_k + D_e u_k + E_e \tilde{y}_k \end{cases},$$

where $s_k \in \mathbb{R}^{n_s}$ is the state of the anomaly detector and $r_k \in \mathbb{R}^{n_r}$ is the residue evaluated to detect and locate existing anomalies.

Let $\mathbf{r}_{[k_0,\, k_f]} = \{r_{k_0},\ r_{k_0+1},\ \ldots,\ r_{k_f}\}$ be the residue discrete-time signal in the time-interval $[k_0,\ k_f] = \{k_0,\ \ldots,\ k_f\}$, which is also denoted in vector form as $\mathbf{r}_{[k_0,\, k_f]} \in \mathbb{R}^{n_r(k_f-k_0+1)}$, with $\mathbf{r}_{[k_0,\, k_f]} = [r_{k_0}^\top,\ \ldots,\ r_{k_f}^\top]^\top$. When the time-interval is clearly defined from the context, the short-form notation $\mathbf{r}$ will be used in place of $\mathbf{r}_{[k_0,\, k_f]}$. Given the residue signal over the time-interval $[k_0,\ k_f]$ and a set $\mathcal{U}_{[k_0,\, k_f]}$, an alarm is triggered if

$$\mathbf{r}_{[k_0,\, k_f]} \notin \mathcal{U}_{[k_0,\, k_f]}. \tag{4.1}$$

In particular, we consider a norm-based characterization of $\mathcal{U}_{[k_0,\, k_f]}$, namely

$$\mathcal{U}_{[k_0,\, k_f]} \triangleq \{\mathbf{r} :\ \|\mathbf{r}_{[k_0,\, k_f]}\|_p \leq \delta\},$$

where $\|\mathbf{r}_{[k_0,\, k_f]}\|_p$ with $1 \leq p \leq \infty$ denotes the p-norm of the discrete-time signal $\mathbf{r}$ in the time-interval $[k_0,\ k_f]$.

4.1.1 Attack Scenario: Data Deception

For the attack scenario, data deception attacks are considered. The deception attacks modify the control actions u_k and sensor measurements y_k from their calculated or real values to the corrupted signals $\tilde{u}_k$ and $\tilde{y}_k$, respectively. Denoting $\mathcal{R}_I^u \subseteq \{1, \ldots, n_u\}$ and $\mathcal{R}_I^y \subseteq \{1, \ldots, n_y\}$ as the deception resources, i.e. set of actuator and sensor channels that can be affected, and $|\mathcal{R}_I^u|$ and $|\mathcal{R}_I^u|$ as the respective cardinality of the sets, the deception attacks are modeled as

$$\begin{aligned} \tilde{u}_k &\triangleq u_k + \Gamma^u b_k^u, \\ \tilde{y}_k &\triangleq y_k + \Gamma^y b_k^y, \end{aligned}$$

where the signals $b_k^u \in \mathbb{R}^{|\mathcal{R}_I^u|}$ and $b_k^y \in \mathbb{R}^{|\mathcal{R}_I^y|}$ represent the data corruption and $\Gamma^u \in \mathbb{B}^{n_u \times |\mathcal{R}_I^u|}$ and $\Gamma^y \in \mathbb{B}^{n_y \times |\mathcal{R}_I^y|}$ ($\mathbb{B} \triangleq \{0,\, 1\}$) are the binary incidence matrices mapping the data corruption to the respective data channels. The matrices Γ^u and Γ^y indicate which data channels can be accessed by the adversary and are therefore directly related to the adversary resources in deception attacks. The number of data channels that may be compromised by the adversary are given by $n_a = |\mathcal{R}_I^u| + |\mathcal{R}_I^u|$.

Defining the attack vector $a_k = [b_k^{u\top}\quad b_k^{y\top}]^\top$, the system components under attack are now characterized. Stacking the states of the plant and controller as $\eta_k = [x_k^\top\quad z_k^\top]^\top$ and the states of the plant, controller, and anomaly detector as

$\xi_k = [\eta_k^\top \quad s_k^\top]^\top$, the dynamics of the closed-loop system and the residue dynamics under attack can be written respectively as

$$\begin{aligned} \eta_{k+1} &= \mathbf{A}\eta_k + \mathbf{B}a_k \\ \tilde{y}_k &= \mathbf{C}\eta_k + \mathbf{D}a_k, \end{aligned} \tag{4.2}$$

$$\begin{aligned} \xi_{k+1} &= \mathbf{A}_e\xi_k + \mathbf{B}_e a_k \\ r_k &= \mathbf{C}_e\xi_k + \mathbf{D}_e a_k. \end{aligned} \tag{4.3}$$

The matrices $\mathbf{B}$, $\mathbf{D}$, $\mathbf{B}_e$, and $\mathbf{D}_e$ capture the way in which the attack vector a_k affects the closed-loop and residue dynamics. For data deception attacks, these matrices are characterized as

$$\mathbf{B} = \begin{bmatrix} B\Gamma^u & BD_c\Gamma^y \\ 0 & B_c\Gamma^y \end{bmatrix}, \qquad \mathbf{D} = \begin{bmatrix} 0 & \Gamma^y \end{bmatrix},$$

$$\mathbf{B}_e = \begin{bmatrix} \mathbf{B} \\ (B_eD_c + K_e)\mathbf{D} \end{bmatrix}, \quad \mathbf{D}_e = (D_eD_c + E_e)\mathbf{D}.$$

The remaining matrices are defined in Section 3.4.

Attack Goals and Constraints

In addition to the attack resources, the attack scenarios need to include the adversary's intent, namely the attack goals and constraints shaping the attack policy. The attack goals can be stated in terms of the attack impact on the system operation, while the constraints may be related to the attack detectability.

Several physical systems have tight operating constraints which if not satisfied might result in physical damage to the system. In this work we use the concept of safe sets to characterize the safety constraints.

Definition 4.1.1. *For a given time-interval $[k_0,\, k_f]$, the system is said to be safe if $\mathbf{x}_{[k_0,\, k_f]} \in \mathcal{S}_{[k_0,\, k_f]}$, where $\mathcal{S}_{[k_0,\, k_f]}$ is a compact set with non-empty interior.*

The above definition of safe set $\mathcal{S}_{[k_0,\, k_f]}$ allows one to consider both time-interval and time-instant characterizations of safe regions, for instance signal energy and safe regions of the state space, respectively.

Assumption 4.1.1. *The system is in a safe state at the beginning of the attack, i.e. $\mathbf{x}_{(-\infty,\, k_0-1]} \in \mathcal{S}_{(-\infty,\, k_0-1]}$.*

The physical impact of an attack can be evaluated by assessing whether or not the state of the system remained in the safe set during and after the attack. The attack is considered successful if the state is driven out of the safe set. For simplicity of notation, the safe set $\mathcal{S}_{[k_0,\, k_f]}$ will be simply denoted as $\mathcal{S}$ whenever the time-interval is not ambiguous. Moreover, the safe sets considered in the remainder of this chapter are of the form $\mathcal{S}^p_{[k_0,\, k_f]} = \{\mathbf{x} :\ \|\mathbf{x}_{[k_0,\, k_f]}\|_p \leq 1\}$.

Regarding the attack constraints, we consider that attacks are constrained to remain stealthy. Denote $\mathbf{a}_{[k_0,\, k_f]} = \{a_{k_0}, \ldots, a_{k_f}\}$ as the attack signal, and recall that the residue signal $\mathbf{r}_{[k_0,\, +\infty)}$ is a function of the attack signal. Assuming the system is behaving nominally before the attack, the set of stealthy attacks is defined as follows.

Definition 4.1.2. *The attack signal* $\mathbf{a}_{[k_0,\, k_f]}$ *is stealthy over the time-interval* $[k_0,\ k_f]$ *if* $\mathbf{r}_{[k_0,\, k_f]} \in \mathcal{U}_{[k_0,\, k_f]}$.

4.1.2 Risk Analysis

Recall, from Section 2.3.3, that *risk* is defined as the set of tuples

$$\text{Risk} \equiv \big\{(\text{Scenario, Impact, Likelihood})\big\}\,.$$

In the following, we develop quantitative methods for assessing the cyber security of networked control systems through risk analysis. In particular, given the attack scenario characterized in the previous section, we propose different metrics to assess the risk of different threats, where attacks requiring large amounts of disruption resources are considered to be less likely. The proposed metrics capture trade-offs in terms of impact on the control system, attack detectability, and adversarial resources.

4.2 Static Case

The risk assessment in this section focuses on analyzing the threat's likelihood, indicated by the minimum number of sensors that need to be compromised by the adversary for a given attack scenario. The minimum number of compromised sensors is a relevant indicator of the threat's likelihood because the sensors are often geographically distributed in networked control systems. As a result, coordinated attacks compromising multiple sensors need to be carried out simultaneously in different locations and they are difficult to implement.

The models in Section 4.1 are simplified in two regards: First, the plant is in steady state. That is, in (4.2) and (4.3), the state vectors η_k and ξ_k are constant for all k, so the subscript k is omitted. The second simplification is that there is no feedback control. The simplifications are made because they can lead to a more streamlined presentation of the main concept of risk assessment. In addition, in its own right the simplified structure is relevant in analyzing the cyber security of large-scale systems, such as electric power systems, and gas and water distribution networks. In particular, this simplified structure will be illustrated for electric power systems in Section 4.6. The risk assessment for the general dynamical models will be deferred to a later section in this chapter.

The model for risk assessment is the relationship between the static plant states x and the measurements $\tilde{y}$ received at the anomaly detector. This is described by the expression

$$\tilde{y} = Cx + \Gamma^y b^y = Cx + \Delta y, \tag{4.4}$$

where C is the measurement matrix, and $\Delta y = \Gamma^y b^y$ is the measurement data attack. In a typical static state estimation problem such as the power network case, there are more measurements than states and hence C is assumed to have full column rank (Abur and Exposito, 2004; Monticelli, 1999). Based on the risk assessment model, the least squares estimate of the states is $\hat{x} = \left(C^\top C\right)^{-1} C^\top \tilde{y}$, and the estimate of measurements can be expressed as $\hat{y} = C\hat{x} = C\left(C^\top C\right)^{-1} C^\top \tilde{y}$. Thus, the anomaly detector, which is based on measurement residual, can be described by

$$r \triangleq S\tilde{y} = \Big(I - C\left(C^\top C\right)^{-1} C^\top\Big)\tilde{y}. \tag{4.5}$$

Such an anomaly detector is in general sufficient to detect Δy in the form of a single error involving only one faulty measurement (Abur and Exposito, 2004; Monticelli, 1999). However, in face of a coordinated malicious attack on multiple measurements the anomaly detector can fail. In particular, in (Liu *et al.*, 2009) it was reported that an attack of the form

$$\Delta y = C\Delta x \tag{4.6}$$

for an arbitrary Δx would not result in any residual in (4.5), in addition to the residual caused by other factors such as measurement noise. In fact, the set of stealthy deception attacks with respect to the anomaly detector (4.5) and a zero detection threshold is characterized by (4.6), and these attacks were also experimentally verified in a realistic testbed in Section 3.6.1. Although stealthy attacks may be obtained from (4.6), distinct choices of Δx may yield attack vectors Δy requiring significantly different amount of adversary resources, in terms of the number of nonzero entries of the attack vector Δy and the matrix Γ^y. This number is also an indicator of the likelihood of the success of stealthy attack, as discussed earlier in this subsection.

Next we characterize the stealthy attack vectors with the minimum number of nonzero entries, as a concrete example of the quantitative method for risk assessment.

4.2.1 Minimum-Resource Attacks

There is a significant amount of literature studying the stealthy attack in (4.6) and its consequences to state estimation data integrity (Liu *et al.*, 2009; Kosut *et al.*, 2010; Kim and Poor, 2011; Sou *et al.*, 2013b; Giani *et al.*, 2013)). Liu *et al.* (2009) numerically showed that stealthy attacks $\Delta y = C\Delta x$ are often sparse. To analyze the stealthy attacks with the minimum number of nonzero entries, in Sandberg *et al.* (2010) the notion of security index ρ_j for a measurement j was introduced as the optimal objective value of the following cardinality minimization problem:

$$\begin{aligned} \rho_j \quad \triangleq \quad & \underset{\Delta x \in \mathbb{R}^{n_x}}{\text{minimize}} \quad \|C\Delta x\|_0 \\ & \text{subject to} \quad e_j^\top C\Delta x = 1, \end{aligned} \tag{4.7}$$

where $\|C\Delta x\|_0$ denotes the cardinality (i.e., the number of nonzero entries) of the vector $C\Delta x$, j is the label of the measurement for which the security index ρ_j is computed, and e_j denotes the j-th column of the identity matrix. In Section 3.6.1, the previous optimization problem was solved to compute the sparsest normalized attack used in the experiments on the SCADA EMS software. Computational algorithms solving the combinatorial problem (4.7) are postponed until Section 4.5.

The security index ρ_j is the minimum number of measurements an attacker needs to compromise in order to attack measurement j without being detected by the anomaly detector. In particular, a small ρ_j implies that measurement j is relatively easy to compromise in a stealthy attack, therefore indicating the likelihood of such a threat. As a result, the knowledge of the security indices for all measurements allows the network operator to pinpoint the security vulnerabilities of the network, and to better protect the network with limited resource. For example, Dán and Sandberg (2010) proposed a method to optimally assign limited encryption protection resources to improve the security of the network based on its security indices.

4.3 Dynamical Case: Transient Analysis

In this section, we consider dynamical systems and proposed cyber security metrics assessing both the impact and likelihood of threats. As mentioned in Section 4.1.1, the adversary aims at driving the system to an unsafe state while remaining stealthy. Additionally we consider that the adversary also has resource constraints, in the sense that only a small number of attack points to the system are available. In the following, several formulations for quantifying cyber security of networked control systems are discussed.

Consider the dynamical system in (4.2) and the time-interval $[0,\ N]$ with $k_0 = 0$ and $k_f = N$. Defining $\mathbf{n} = [\eta_0^\top\ \dots\ \eta_N^\top]^\top$, $\mathbf{a} = [a_0^\top\ \dots\ a_N^\top]^\top$, and $\mathbf{y} = [y_0^\top\ \dots\ y_N^\top]^\top$, the state and output trajectories can be described by the following mappings

$$\begin{aligned} \mathbf{n} &= \mathcal{O}_\eta \eta_0 + \mathcal{T}_\eta \mathbf{a} \\ \mathbf{y} &= \mathcal{C}_\eta \mathbf{n} + \mathcal{D}_\eta \mathbf{a}, \end{aligned} \tag{4.8}$$

where

$$\mathcal{O}_\eta = \begin{bmatrix} I \\ \mathbf{A} \\ \mathbf{A}^2 \\ \vdots \\ \mathbf{A}^N \end{bmatrix}, \quad \mathcal{T}_\eta = \begin{bmatrix} \mathbf{D} & 0 & \dots & 0 \\ \mathbf{B} & 0 & \dots & 0 \\ \mathbf{AB} & \mathbf{B} & \dots & 0 \\ \vdots & \vdots & \ddots & 0 \\ \mathbf{A}^{N-1}\mathbf{B} & \mathbf{A}^{N-2}\mathbf{B} & \dots & \mathbf{B} \end{bmatrix},$$

$$\mathcal{C}_\eta = I_{N+1} \otimes \mathbf{C}, \quad \mathcal{D}_\eta = I_{N+1} \otimes \mathbf{D}$$

Similarly for (4.3), defining $\mathbf{e} = [\xi_0^\top \; \dots \; \xi_{N-1|N-1}^\top]^\top$, $\mathbf{r} = [r_0^\top \; \dots \; r_N^\top]^\top$ yields

$$\begin{aligned} \mathbf{e} &= \mathcal{O}_\xi \xi_0 + \mathcal{T}_\xi \mathbf{a} \\ \mathbf{r} &= \mathcal{C}_\xi \mathbf{e} + \mathcal{D}_\xi \mathbf{a}. \end{aligned} \tag{4.9}$$

Recall that the system is operating safely during the time-interval $[k_0,\; k_f]$ if $\mathbf{x} \in \mathcal{S}_{[k_0,\, k_f]}$. Supposing $\mathcal{S}^p_{[k_0,\, k_f]} = \{\mathbf{x} : \; \|\mathbf{x}_{[k_0,\, k_f]}\|_p \le 1\}$ for $p \ge 1$, the system is safe during the time-interval $\{0, 1, \dots, N\}$ if

$$\mathbf{x} \triangleq \mathcal{C}_x \mathbf{n} \in \mathcal{S}^p_{[0,\, N]},$$

where $\mathcal{C}_x = I_{N+1} \otimes [I_n \; 0]$. In particular, for $p = \infty$ we have that the system is safe if $\|\mathbf{x}\|_\infty = \|\mathcal{C}_x \mathbf{n}\|_\infty \le 1$.

4.3.1 Maximum-Impact Attacks

One possible way to quantify cyber security is by analyzing the impact of attacks on the control system, given some pre-defined resources available to the adversary. Recalling the safe set introduced earlier, $\mathcal{S}^p_{[0,\, N]} = \{\mathbf{x} : \; \|\mathbf{x}_{[0,\, N]}\|_p \le 1\}$, the attack impact during the time-interval $[0,\; N]$ is characterized by

$$g_p(\mathbf{n}) = \begin{cases} \|\mathcal{C}_x \mathbf{n}\|_p & \text{, if } \; \mathcal{C}_x \mathbf{n} \in \mathcal{S}^p_{[0,\, N]} \\ +\infty & \text{, otherwise} \end{cases}$$

since the adversary aims at driving the system to an unsafe state. Similarly, recall the set of stealthy attacks $\mathbf{a}$ such that $\mathbf{r} \in \mathcal{U}_{[k_0,\, k_f]} \triangleq \{\mathbf{r} : \; \|\mathbf{r}_{[k_0,\, k_f]}\|_p \le \delta\}$.

The attack yielding the maximum impact can be computed by solving

$$\begin{aligned} \underset{\mathbf{a}}{\text{maximize}} \quad & g_p(\mathbf{n}) \\ \text{subject to} \quad & \left\|\mathcal{C}_\xi \mathbf{e} + \mathcal{D}_\xi \mathbf{a}\right\|_q \le \delta, \\ & \mathbf{e} = \mathcal{O}_\xi \xi_0 + \mathcal{T}_\xi \mathbf{a}, \\ & \mathbf{n} = \mathcal{O}_\eta \eta_0 + \mathcal{T}_\eta \mathbf{a}, \end{aligned} \tag{4.10}$$

with p and q possibly different. Given the objective function $g_p(\mathbf{n})$, the adversary's optimal policy is to drive the system to an unsafe state while keeping the residue below the threshold. When the unsafe state is not reachable while remaining stealthy, the optimal attack drives the system as close to the unsafe set as possible by maximizing $\|\mathbf{x}_{[0,\,N]}\|_p = \|\mathcal{C}_x \mathbf{n}\|_p$.

Letting $\xi_0 = 0$ and $\eta_0 = 0$, the optimal values of (4.10) can be characterized by analyzing the following modified problem

$$\begin{array}{ll} \underset{\mathbf{a}}{\text{maximize}} & \|\mathcal{T}_x \mathbf{a}\|_p \\ \text{subject to} & \|\mathcal{T}_r \mathbf{a}\|_q \leq \delta, \end{array} \tag{4.11}$$

where $\mathcal{T}_x = \mathcal{C}_x \mathcal{T}_\eta$ and $\mathcal{T}_r = \mathcal{C}_\xi \mathcal{T}_\xi + \mathcal{D}_\xi$. The conditions under which (4.11) admits bounded optimal values are characterized in the following result.

Lemma 4.3.1. *The problem* (4.11) *is bounded if and only if* $\operatorname{Ker}(\mathcal{T}_r) \subseteq \operatorname{Ker}(\mathcal{T}_x)$.

Proof. Suppose that $\operatorname{Ker}(\mathcal{T}_r) \neq \emptyset$ and consider the subset of solutions where $\mathbf{a} \in \operatorname{Ker}(\mathcal{T}_r)$. For this subset of solutions, the optimization problem becomes

$$\underset{\mathbf{a} \in \operatorname{Ker}(\mathcal{T}_r)}{\text{maximize}} \quad \|\mathcal{T}_x \mathbf{a}\|_p .$$

Since the latter corresponds to a maximization of a convex function, its solution is unbounded unless $\mathcal{T}_x \mathbf{a} = 0$ for all $\mathbf{a} \in \operatorname{Ker}(\mathcal{T}_r)$ i.e., $\operatorname{Ker}(\mathcal{T}_r) \subseteq \operatorname{Ker}(\mathcal{T}_x)$. For $\mathbf{a} \notin \operatorname{Ker}(\mathcal{T}_r)$ the feasible set is compact and thus the objective function over the feasible set is bounded, which concludes the proof. □

Supposing that the optimization problem (4.11) is bounded and $p = q = 2$, (4.11) can be rewritten as a generalized eigenvalue problem. Moreover, a closed-form solution parameterized by a generalized eigenvalue and eigenvector pair can be obtained.

Theorem 4.3.2. *Let* $p = q = 2$ *and suppose that* $\operatorname{Ker}(\mathcal{T}_r) \subseteq \operatorname{Ker}(\mathcal{T}_x)$. *The optimal attack policy for* (4.11) *is given by*

$$\mathbf{a}^\star = \frac{\delta}{\|\mathcal{T}_r \mathbf{v}^\star\|_2} \mathbf{v}^\star,$$

where $\mathbf{v}^\star$ *is the generalized eigenvector associated with* λ^*, *the largest generalized eigenvalue of the matrix pencil* $\left(\mathcal{T}_x^\top \mathcal{T}_x,\ \mathcal{T}_r^\top \mathcal{T}_r\right)$. *Moreover, the corresponding optimal value is given by* $\|\mathcal{T}_x \mathbf{a}^\star\|_2 = \sqrt{\lambda^*}\delta$.

Proof. The proof is similar to that of Theorem 3.5.7, and is thus omitted. □

Given the solution to (4.11) characterized by the previous result, the maximum impact with respect to (4.10) is given by

$$g_p(\mathcal{T}_x \mathbf{a}^\star) = \begin{cases} \sqrt{\lambda^*}\delta & \text{, if } \sqrt{\lambda^*}\delta \leq 1 \\ +\infty & \text{, otherwise.} \end{cases}$$

4.3.2 Minimum-Resource Attacks

Cyber security of control systems can also be quantified by assessing the number of resources needed by the adversary to perform a given set of attacks, without necessarily taking into account the attack impact, as formulated below.

Consider the set of attacks $\mathcal{A}_G$ such that $\mathbf{a} \in \mathcal{A}_G$ satisfies the goals of a given attack scenario. Recall that $a_k \in \mathbb{R}^{n_a}$ for all $k \in [k_0,\ k_f]$ and denote $\mathbf{a}_{(i),\,[k_0,\,k_f]} = \{a_{(i),k_0},\ \ldots,\ a_{(i),k_f}\}$ as the signal corresponding to the $i-$th attack resource, where $a_{(i),k}$ is the i-th entry of the vector a_k. Consider the function

$$h_p(\mathbf{a}) = [\|\mathbf{a}_{(1)}\|_p\ \ldots\ \|\mathbf{a}_{(n_a)}\|_p]^\top$$

with $1 \leq p \leq +\infty$. The number of resources employed in a given attack are $\|h_p(\mathbf{a})\|_0$, where $\|x\|_0$ denotes the number of non-zero elements of the vector x. For the set of attacks $\mathcal{A}_G$, the minimum-resource attacks are computed by solving the following optimization problem

$$\begin{aligned} \underset{\mathbf{a}}{\text{minimize}} \quad & \|h_p(\mathbf{a})\|_0 \\ \text{subject to} \quad & \left\|\mathcal{C}_\xi \mathbf{e} + \mathcal{D}_\xi \mathbf{a}\right\|_q \leq \delta, \\ & \mathbf{e} = \mathcal{O}_\xi \xi_0 + \mathcal{T}_\xi \mathbf{a}, \\ & \mathbf{a} \in \mathcal{A}_G. \end{aligned} \tag{4.12}$$

Although the set $\mathcal{A}_G$ may be chosen depending on the attack impact $g_p(\mathbf{n})$, i.e., $\mathcal{A}_G = \{\mathbf{a} :\ g_p(\mathbf{n}) = \|\mathcal{T}_x \mathbf{a}\|_p > \gamma\}$, this generally results in non-convex constraints that increase the computational complexity of the problem. As an example, the set $\mathcal{A}_G = \{\mathbf{a} :\ \|\mathcal{T}_x \mathbf{a}\|_\infty > \gamma\}$ is formulated as a set of linear constraints with binary variables in (4.21). However, $\mathcal{A}_G$ might not be directly related to the impact of the attack in terms of $g_p(\mathbf{n})$. For instance, the formulation (4.12) captures the security-index proposed for static systems in Section 4.2.1, where the adversary aims at corrupting a given measurement i without being detected. The security-index formulation is retrieved by having $\xi_0 = 0$, $N = 0$, $\delta = 0$, and

$$\mathcal{A}_G = \{\mathbf{a} \in \mathbb{R}^{n_a} :\ \mathbf{a}_{(i)} = 1\}.$$

However, for dynamic systems with $N > 0$, the specification of the attack scenario and corresponding set of attacks $\mathcal{A}_G$ is more involved. The same scenario where the adversary aims at corrupting a given channel i can be formulated by having $\delta = 0$ and $\mathcal{A}_G = \{\mathbf{a} :\ \|\mathbf{a}_{(i)}\|_p = \epsilon\}$. For positive values of δ, the feasibility of the problem depends on both δ and ϵ, which need to be carefully chosen.

4.3.3 Maximum-Impact Minimum-Resource Attacks

The previous formulations considered impact and resources independently when quantifying cyber security. Here the impact and resources are addressed simultaneously by considering the multi-objective optimization problem

$$\begin{aligned} \underset{\mathbf{a}}{\text{maximize}} \quad & [g_p(\mathbf{n}),\ -\|h_p(\mathbf{a})\|_0]^\top \\ \text{subject to} \quad & \left\|\mathcal{C}_\xi \mathbf{e} + \mathcal{D}_\xi \mathbf{a}\right\|_q \leq \delta, \\ & \mathbf{e} = \mathcal{O}_\xi \xi_0 + \mathcal{T}_\xi \mathbf{a}, \\ & \mathbf{n} = \mathcal{O}_\eta \eta_0 + \mathcal{T}_\eta \mathbf{a}. \end{aligned} \tag{4.13}$$

The vector-valued objective function indicates that the adversary desires to simultaneously maximize and minimize $g_p(\mathbf{n})$ and $\|h_p(\mathbf{a})\|_0$, respectively. Solutions to multi-objective problems are related to the concept of Pareto optimality (Marler and Arora, 2004) and correspond to the optimal trade-off manifold between the objectives. These solutions can be obtained through several techniques, for instance the bounded objective function method in which all but one of the objectives are posed as constraints, thus obtaining a scalar-valued objective function. Applying this method to (4.13) and constraining $\|h_p(\mathbf{a})\|_0$ yields

$$\begin{aligned} \underset{\mathbf{a}}{\text{maximize}} \quad & g_p(\mathbf{n}) \\ \\ \text{subject to} \quad & \left\|\mathcal{C}_\xi \mathbf{e} + \mathcal{D}_\xi \mathbf{a}\right\|_q \leq \delta, \\ & \mathbf{e} = \mathcal{O}_\xi \xi_0 + \mathcal{T}_\xi \mathbf{a}, \\ & \mathbf{n} = \mathcal{O}_\eta \eta_0 + \mathcal{T}_\eta \mathbf{a}, \\ & \|h_p(\mathbf{a})\|_0 < \epsilon, \end{aligned} \tag{4.14}$$

which can be interpreted as a maximum-impact resource-constrained attack policy. The Pareto frontier that characterizes the optimal trade-off manifold can be obtained by iteratively solving (4.14) for $\epsilon \in \{1,\ \dots,\ n_a\}$. This approach is illustrated in Section 4.6 for the quadruple-tank process.

4.4 Dynamical Case: Steady-State Analysis

Here we consider the steady-state of the system under attack. Let $\nu \in \mathbb{C}$ and define

$$\begin{aligned} G_{xa}(\nu) &= [I_n\ 0](\nu I - \mathbf{A})^{-1}\mathbf{B} + \mathbf{D}, \\ G_{ra}(\nu) &= C_e(\nu I - A_e)^{-1}B_e + D_e, \end{aligned}$$

which correspond to the transfer functions from a_k to x_k and r_k respectively. Considering exponential attack signals of the form $a_k = g\nu^k$ for fixed ν, denote $a(\nu) = g \in \mathbb{C}^{n_a}$, $x(\nu) = G_{xa}(\nu)a(\nu)$, and $r(\nu) = G_{ra}(\nu)a(\nu)$ as the phasor notation of a_k, x_k, and r_k, respectively. Since the analysis in this section is restricted to steady-state, we consider ν to be on the unit circle,

$$\nu \in \mathbb{S} \triangleq \{\nu \in \mathbb{C} :\ |\nu| = 1\}$$

, and thus $a(\nu)$ corresponds to sinusoidal signals of constant magnitude. Defining the frequency-domain safe set as $\mathcal{S}_\infty^p = \{x \in \mathbb{C}^n : \|x\|_p \leq 1\}$, the system under attack is said to be safe at steady-state if $x(\nu) = G_{xa}(\nu)a(\nu) \in \mathcal{S}_\infty^p$.

4.4.1 Maximum-Impact Attacks

For a given $\nu \in \mathbb{S}$, the steady-state attack impact is characterized by

$$g_p(x(\nu)) = \begin{cases} \|x(\nu)\|_p & \text{, if } x(\nu) \in \mathcal{S}_\infty^p \\ +\infty & \text{, otherwise.} \end{cases}$$

Similarly, recall the set of steady-state stealthy attacks $a(\nu)$ such that

$$r(\nu) \in \mathcal{U} \triangleq \{r \in \mathbb{C}^{n_r} : \|r\|_p \leq \delta\},$$

where $r(\nu) = G_{ra}(\nu)a(\nu)$.

The attack yielding the maximum impact can be computed by solving

$$\begin{aligned} \sup_{\nu \in \mathbb{S}} \quad & \underset{a(\nu)}{\text{maximize}} && g_p(G_{xa}(\nu)a(\nu)) \\ & \text{subject to} && \left\|G_{ra}(\nu)a(\nu)\right\|_p \leq \delta. \end{aligned} \tag{4.15}$$

The maximum impact over all stealthy attacks can be computed by replacing the objective function $g_p(G_{xa}(\nu)a(\nu))$ with $\|G_{xa}(\nu)a(\nu)\|_p$, solving

$$\begin{aligned} \sup_{\nu \in \mathbb{S}} \quad & \underset{a(\nu)}{\text{maximize}} && \|G_{xa}(\nu)a(\nu)\|_p \\ & \text{subject to} && \left\|G_{ra}(\nu)a(\nu)\right\|_q \leq \delta, \end{aligned} \tag{4.16}$$

and evaluating $g_p(G_{xa}(\nu)a(\nu))$ for the obtained solution. The conditions under which (4.16) admits bounded optimal values are characterized as follows.

Lemma 4.4.1. *The optimization problem* (4.16) *is bounded if and only if the relation*

$$\operatorname{Ker}\big(G_{ra}(\nu)\big) \subseteq \operatorname{Ker}\big(G_{xa}(\nu)\big)$$

holds for all $\nu \in \mathbb{S}$.

Proof. The proof follows the same reasoning as that of Lemma 4.3.1. □

The previous statement is related to the concept of invariant-zeros of dynamical systems (Tokarzewski, 2006).

Definition 4.4.1. *Consider a linear time-invariant system in discrete-time with the state-space realization* (A, B, C, D) *and the equation*

$$\begin{bmatrix} \lambda_z I - A & -B \\ C & D \end{bmatrix} \begin{bmatrix} x_0 \\ u_z \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \tag{4.17}$$

with $\lambda_z \in \mathbb{C}$ and $x_0 \neq 0$. For a given solution to the previous equation (λ_z, u_z, x_0), denote λ_z as the invariant-zero, u_z as the input-zero direction, and x_0 as the state-zero direction. Furthermore, the tuple (λ_z, u_z, x_0) is denoted as a zero-dynamics of the system (A, B, C, D).

The relation between Lemma 4.4.1 and invariant-zeros is formalized in the following result.

Lemma 4.4.2. *The optimization problem* (4.16) *is bounded if and only if either of the following hold:*

1. *the transfer function $G_{ra}(\nu)$ does not contain invariant-zeros on the unit circle;*

2. *all the invariant-zeros of the transfer function $G_{ra}(\nu)$ on the unit circle are also invariant-zeros of $G_{xa}(\nu)$, with the same input-zero direction.*

Proof. For the first statement, note that if $G_{ra}(\nu)$ does not contain invariant-zeros on the unit circle, then $\operatorname{Ker}\big(G_{ra}(\nu)\big) = \emptyset$ for $\nu \in \mathbb{S}$ and thus (4.16) is bounded. As for the second statement, suppose that $G_{ra}(\nu)$ contains an invariant-zero $\bar{\lambda}_z \in \mathbb{S}$ and recall that (A_e, B_e, C_e, D_e) is the state-space realization of $G_{ra}(\nu)$. For a non-zero state-zero direction x_0, (4.17) can be rewritten as

$$\begin{aligned}(\bar{\lambda}_z I - A_e)x_0 - B_e u_z &= 0,\\ C_e x_0 + D_e u_z &= 0.\end{aligned} \tag{4.18}$$

Since A_e is stable and $|\bar{\lambda}_z| = 1$, we have that $\bar{\lambda}_z I - A_e$ is invertible and thus (4.18) can be rewritten as $\big(C_e(\bar{\lambda}_z I - A_e)^{-1}B_e + D_e\big)\, u_z = G_{ra}(\bar{\lambda}_z)u_z = 0$. Hence we conclude that the input-zero direction u_z lies in the null-space of $G_{ra}(\bar{\lambda}_z)$. In this case, applying Lemma 4.4.1 shows that the problem is bounded if and only if u_z also lies in the null-space of $G_{xa}(\bar{\lambda}_z)$, which concludes the proof. □

Supposing that the optimization problem (4.16) is bounded and $p = 2$ and denoting G^H as the Hermitian conjugate of $G \in \mathbb{C}^{n\times m}$, (4.16) can be rewritten as a generalized eigenvalue problem.

Theorem 4.4.3. *Let $p = q = 2$ and suppose that* $\operatorname{Ker}\big(G_{ra}(\nu)\big) \subseteq \operatorname{Ker}\big(G_{xa}(\nu)\big)$ *for all $\nu \in \mathbb{S}$. The optimal maximum-impact attack policy is given by*

$$a^\star(\nu^\star) = \frac{\delta}{\|G_{ra}(\nu^\star)\mathbf{v}^\star\|_2}\mathbf{v}^\star,$$

where $\mathbf{v}^\star$ is the eigenvector associated with λ^*, the largest generalized eigenvalue of the matrix pencil $\big(G_{xa}^H(\nu)G_{xa}(\nu),\ G_{ra}^H(\nu)G_{ra}(\nu)\big)$ maximized over $\nu \in \mathbb{S}$. Moreover, the corresponding impact is given by $\|G_{xa}(\nu^\star)a^\star(\nu^\star)\|_2 = \sqrt{\lambda^*}\delta$.

Proof. The proof is similar to that of Theorem 3.5.7. □

Given the solution to (4.16) characterized by the previous result, the maximum impact with respect to (4.15) is given by

$$g_p(G_{xa}(\nu^\star)a^\star(\nu^\star)) = \begin{cases} \sqrt{\lambda^*}\delta & \text{, if } \sqrt{\lambda^*}\delta \leq 1 \\ +\infty & \text{, otherwise.} \end{cases}$$

Theorem 4.4.4. *Supposing $G_{ra}(\nu)$ is left-invertible for all $\nu \in \mathbb{S}$, the largest generalized eigenvalue of the matrix pencil $\left(G_{xa}^H(\nu)G_{xa}(\nu),\ G_{ra}^H(\nu)G_{ra}(\nu)\right)$, $\lambda^\star(\nu^\star)$, maximized over $\nu^\star \in \mathbb{S}$ corresponds to the $\mathcal{H}_\infty$-norm of $G_{xa}(\nu)G_{ra}^\dagger(\nu)$ with $G_{ra}^\dagger(\nu) = \left(G_{ra}^H(\nu)G_{ra}(\nu)\right)^{-1} G_{ra}^H(\nu)$.*

Proof. First observe that $\operatorname{Ker}\left(G_{ra}(\nu)\right) = \emptyset$, since $G_{ra}(\nu)$ is left-invertible for all $\nu \in \mathbb{S}$. Letting $\delta = 1$, without loss of generality, from Theorem 4.4.3 we then have that

$$\lambda^\star(\nu^\star) = \sup_{\nu\in\mathbb{S}} \underset{a(\nu):\, \|G_{ra}(\nu)a(\nu)\|_2=1}{\text{maximize}} \|G_{xa}(\nu)a(\nu)\|_2.$$

The proof concludes by noting that, since $G_{ra}(\nu)$ is left-invertible and $G_{xa}(\nu)$ and $G_{ra}(\nu)$ are stable, we have $a(\nu) = G_{ra}^\dagger(\nu)b(\nu)$ for some $b(\nu) \in \mathbb{C}^{n_r}$ and so $\lambda^\star(\nu^\star)$ can be rewritten as

$$\lambda^\star(\nu^\star) = \sup_{\nu\in\mathbb{S}} \max_{b(\nu):\, \|b(\nu)\|_2=1} \|G_{xa}(\nu)G_{ra}^\dagger(\nu)b(\nu)\|_2^2 \triangleq \|G_{xa}(\nu)G_{ra}^\dagger(\nu)\|_\infty.$$

□

4.4.2 Minimum-Resource Attacks

Consider the set of attacks $\mathcal{A}_G$ such that $a(\nu) \in \mathcal{A}_G$ satisfies the goals of a given attack scenario. For the set of attacks $\mathcal{A}_G$, the minimum-resource steady-state attacks are computed by solving the following optimization problem

$$\begin{aligned} \inf_{\nu\in\mathbb{S}} \quad \underset{a(\nu)}{\text{minimize}} \quad & \|a(\nu)\|_0 \\ \text{subject to} \quad & \left\|G_{ra}(\nu)a(\nu)\right\|_q \leq \delta, \\ & a(\nu) \in \mathcal{A}_G. \end{aligned}$$

As in the security-index formulation for a given channel i (Sandberg *et al.*, 2010), one can define $\mathcal{A}_G \triangleq \{a(\nu) \in \mathbb{C}^{n_a} :\ a_{(i)}(\nu) = 1\}$.

4.4.3 Maximum-Impact Minimum-Resource Attacks

Similarly as for the transient analysis, the impact and adversarial resources can be treated simultaneously in the multi-objective optimization problem

$$\begin{aligned} \sup_{\nu\in\mathbb{S}} \quad \underset{a(\nu)}{\text{maximize}} \quad & [g_p(G_{xa}(\nu)a(\nu)),\ -\|a(\nu)\|_0]^\top \\ \text{subject to} \quad & \left\|G_{ra}(\nu)a(\nu)\right\|_q \leq \delta. \end{aligned}$$

Using the bounded objective function method (Marler and Arora, 2004), the Pareto frontier can be obtained by iteratively solving the following problem for $\epsilon \in \{1, \ldots, n_a\}$

$$\begin{aligned} \sup_{\nu \in \mathbb{S}} \quad \underset{a(\nu)}{\text{maximize}} \quad & g_p(G_{xa}(\nu)a(\nu)) \\ \text{subject to} \quad & \left\| G_{ra}(\nu)a(\nu) \right\|_q \leq \delta, \\ & \|a(\nu)\|_0 < \epsilon. \end{aligned}$$

4.5 Computational Algorithms

In this section, different metrics are formulated as mixed-integer linear programming problems. First, we consider the minimum-resource attacks for static systems in Subsection 4.5.1, while the maximum-impact resource-constrained formulation for dynamical systems is considered later in Subsection 4.5.2.

4.5.1 Minimum-Resource Attacks on Static Systems

Consider the minimum-resource attacks for static systems (4.7) reproduced below

$$\begin{aligned} \rho_j \quad \triangleq \quad \underset{\Delta x \in \mathbb{R}^{n_x}}{\text{minimize}} \quad & \|C\Delta x\|_0 \\ \text{subject to} \quad & e_j^\top C\Delta x \neq 0. \end{aligned}$$

Because of the cardinality minimization, computing the security indices ρ_j can sometimes be hard. In fact, it can be established that problem (4.7) is NP-hard using techniques from (Tillmann and Pfetsch, 2012; McCormick, 1983). As a result, known exact solution algorithms for (4.7) are enumerative by nature. Three different typical exact algorithms include (a) enumeration on the support of $C\Delta x$, (b) finding the maximum feasible subsystem for an appropriately constructed system of infeasible inequalities (Jokar and Pfetsch, 2008), and (c) the big M method (Tsitsiklis and Bertsimas, 1997). In this section, the big M method is chosen because it is easily adapted to more complex problems, as performed for the security metric for dynamical systems in Subsection 4.5.2. Moreover, the resulting optimization problem can be modeled as a mixed integer linear programming problem and solved using available software such as CPLEX (IBM). The big M method sets up and solves the following optimization problem:

$$\begin{array}{rrcl} \underset{\Delta x,\, \gamma = [\gamma_1 \cdots \gamma_{n_y}]^\top}{\text{minimize}} & \sum_i \gamma_i & & \\ \text{subject to} & C\Delta x & \leq & M\gamma \\ & -C\Delta x & \leq & M\gamma \\ & e_j^\top C\Delta x & = & 1 \\ & \gamma_i & \in & \{0,1\} \quad \forall\, i = 1, \ldots, n_y. \end{array} \tag{4.19}$$

In (4.19), the inequalities are interpreted entry-wise and $0 < M < \infty$ is a user-defined constant scalar. Hence, the inequality constraints can be interpreted as the equivalent set of constraints

$$|e_i^\top C\Delta x| \leq M\gamma_i, \quad \forall\, i = 1, \ldots, n_y.$$

The latter constraint imposes that, for each i, the data corruption $\Delta y_i = e_i^\top C\Delta x$ is constrained in magnitude by $M\gamma_i$. Thus, having the variable $\gamma_i = 0$ yields $|\Delta y_i| \leq 0$, which indicates that the i-th measurement is not be attacked. Furthermore, since $\gamma_i \in \{0, 1\}$ and the cost function is $\sum\limits_i \gamma_i$, we observe that the optimization problem (4.19) aims at minimizing the number of corrupted measurements, i.e., $\|\Delta y\|_0$.

On the other hand, having $\gamma_i = 1$ results in $|\Delta y_i| \leq M$ which, for a large M, renders the attack to the i-th measurement Δy_i relatively free. In fact, if M is greater than the maximum entry of $C\Delta x^\star$ in absolute value, for some optimal solution $\Delta x^\star$ of (4.7), then the optimal solution to (4.19) is exactly an optimal solution to (4.7). Otherwise, solving (4.19) yields a suboptimal solution, optimal among all solutions Δx such that the maximum entry of $C\Delta x$ is less than or equal to M in absolute value. The procedure described in (Schrijver, 1986) can always find a sufficiently large M to ensure that the big M method indeed provides the optimal solution to (4.7). In addition, the physics and insights of the underlying application problem can also lead to a suitable M.

4.5.2 Maximum-Impact Resource-Constrained Attacks on Dynamical Systems

Consider the maximum-impact resource-constrained formulation from the transient analysis (4.14) reproduced below

$$\begin{aligned} \underset{\mathbf{a}}{\text{maximize}} \quad & g_p(\mathbf{n}) \\ \text{subject to} \quad & \left\|\mathcal{C}_\xi \mathbf{e} + \mathcal{D}_\xi \mathbf{a}\right\|_p \leq \delta, \\ & \|h_p(\mathbf{a})\|_0 \leq \epsilon, \\ & \mathbf{e} = \mathcal{O}_\xi \xi_0 + \mathcal{T}_\xi \mathbf{a}, \\ & \mathbf{n} = \mathcal{O}_\eta \eta_0 + \mathcal{T}_\eta \mathbf{a}. \end{aligned}$$

For $1 \leq p \leq \infty$, the constraint $\|h_p(\mathbf{a})\|_0 \leq \epsilon$ models the fact that the number of channels the adversary can compromise is upper bounded by epsilon. Using the big M method described in Subsection 4.5.1, by introducing the binary decision variables γ_i, one for each channel, and a given large positive scalar M_h used to model "infinity", the constraint can be modeled as follows:

$$\begin{array}{rcll} \mathbf{a}_{(i)} & \leq & M_h \gamma_i \mathbf{1} & \forall\, i = 1,\, \ldots,\, n_a \\ -\mathbf{a}_{(i)} & \leq & M_h \gamma_i \mathbf{1} & \forall\, i = 1,\, \ldots,\, n_a \\ \sum_{i=1}^{n_a} \gamma_i & \leq & \epsilon & \\ \gamma_i & \in & \{0, 1\} & \forall\, i = 1,\, \ldots,\, n_a, \end{array} \tag{4.20}$$

where $\mathbf{1}$ is a vector of ones of appropriate dimension. The constant M_h is typically chosen according to the physical limitation of the system. The binary decision variables γ_i serve to count the number of channels the adversary can compromise, similarly to the interpretation discussed in Subsection 4.5.1. That is, $\gamma_i = 1$ if and only if channel i can be compromised. Once a channel is compromised, the adversary is expected to be able to modify the time signal in that channel in any way he desires. This is modeled by the first two sets of constraints in (4.20).

In the constraint $\left\|C_\xi \mathbf{e} + D_\xi \mathbf{a}\right\|_p \leq \delta$, the p-norm is chosen to be the infinity norm, which models a constraint on the worst case output violation. This constraint can be modeled as

$$\begin{array}{rcl} C_\xi \mathbf{e} + D_\xi \mathbf{a} & \leq & \delta \mathbf{1} \\ -C_\xi \mathbf{e} - D_\xi \mathbf{a} & \leq & \delta \mathbf{1}. \end{array}$$

In the objective function $g_p(n)$, the safety set $\mathcal{S}^p$ is chosen to be a ∞-norm ball. That is, $\mathcal{C}_x \mathbf{n} \in \mathcal{S}^p$ if and only if $\|\mathcal{C}_x \mathbf{n}\|_\infty \leq M_\mathcal{S}$ for some given safety tolerance $M_\mathcal{S}$. This is to model the fact that if any component of $\mathcal{C}_x \mathbf{n}$ is too large, then the system is considered to be unsafe. Consequently, the adversary's goal is to maximize $g_p(\mathbf{n})$ so that at least one component of $\mathcal{C}_x \mathbf{n}$ is larger than the safety tolerance $M_\mathcal{S}$. In hypograph form (Boyd and Vandenberghe, 2004), maximizing $g_p(\mathbf{n})$ amounts to maximizing a slack variable φ with the additional constraint that $g_p(\mathbf{n}) \geq \varphi$. The latter constraint can be modeled as

$$\begin{array}{rcll} \mathcal{C}_x \mathbf{n} & \geq & +\varphi \mathbf{1} - M_{\mathcal{C}_x} \left(\mathbf{1} - \gamma^+\right) & \\ \mathcal{C}_x \mathbf{n} & \leq & -\varphi \mathbf{1} + M_{\mathcal{C}_x} (\mathbf{1} - \gamma^-) & \\ \gamma_i^+ + \gamma_i^- & \leq & 1 & \forall\, i \\ \sum_i \left(\gamma_i^+ + \gamma_i^-\right) & \geq & 1 & \\ \gamma_i^+ & \in & \{0, 1\} & \forall\, i \\ \gamma_i^- & \in & \{0, 1\} & \forall\, i. \end{array} \tag{4.21}$$

In (4.21), $M_{\mathcal{C}_x}$ is another given large number used to represent "infinity". For each i, when the binary decision variable $\gamma_i^+ = 1$, the i-th constraint of $\mathcal{C}_x \mathbf{n} \geq \varphi \mathbf{1} - M_{\mathcal{C}_x} \left(\mathbf{1} - \gamma^+\right)$ implies that the i-th component of $\mathcal{C}_x \mathbf{n}$ is greater than or equal to φ. On the other hand, if $\gamma_i^+ = 0$ then this constraint component can be ignored. A similar interpretation holds for the combination of γ^- and $\mathcal{C}_x \mathbf{n} \leq -\varphi \mathbf{1} + M_{\mathcal{C}_x}(\mathbf{1} - \gamma^-)$. Furthermore, the constraint $\gamma_i^+ + \gamma_i^- \leq 1$ models the fact that the i-th component of $\mathcal{C}_x\mathbf{n}$ cannot be both greater than φ and less than $-\varphi$, when $\varphi > 0$. Together with the above discussion, the constraint $\sum\limits_i \left(\gamma_i^+ + \gamma_i^-\right) \geq 1$ indicates that at least one component of $\mathcal{C}_x\mathbf{n}$ must be greater than or equal to γ in absolute value. Since the objective is to maximize φ, it holds that $\varphi = \|\mathcal{C}_x\mathbf{n}\|_\infty$ at optimality. Finally, to model the fact that, once the goal $\|\mathcal{C}\mathbf{n}\|_\infty > M_\mathcal{S}$ is achieved, the adversary no longer needs to maximize φ, an additional constraint can be imposed:

$$\gamma \leq M_\mathcal{S}.$$

In conclusion, the maximum-impact resource-constrained attack is modeled by the following mixed integer linear program:

$$\begin{aligned}
\underset{\mathbf{a},\varphi,\gamma,\gamma^+,\gamma^-}{\text{maximize}} \quad & \varphi \\
\text{subject to} \quad & \mathbf{e} = \mathcal{O}_\xi \xi_0 + \mathcal{T}_\xi \mathbf{a}, \\
& \mathbf{n} = \mathcal{O}_{\eta_0} \eta_0 + \mathcal{T}_\eta \mathbf{a}, \\
& (4.20), (4.5.2), (4.21), (4.5.2).
\end{aligned} \tag{4.22}$$

4.6 Numerical Examples

Numerical examples are presented to illustrate some of the proposed formulations for quantifying cyber security of control systems.

4.6.1 Electric Power Systems

Next, we present the results obtained by computing the minimum-resource security index for data deception attacks on the measurements of electric power systems.

The power network used in this example is depicted in Figure 4.1 and consists of 14 substations and the bus-branch model has 27 buses and 40 branches. Several measurements are available at each substation, which can be corrupted by the adversary. The system is modeled in Section 2.5.1, where formulas relating the state and measurements are given. In particular, here we consider the DC model of the power network, which is captured in the static model (4.4) reproduced below

$$\tilde{y} = Cx + \Gamma^y b^y = Cx + \Delta y.$$

Consider the security index ρ_j formulated as the combinatorial problem (4.7). For each measurement j, the corresponding value of ρ_j was computed by solving the mixed-integer linear programming problem (4.19). The result is presented in Figure 4.2. Given the default measurement configuration of the power network, the security metric ρ_j (the red full circles) yields quite heterogeneous results. Recalling that ρ_j is the minimum number of measurements needed to perform a stealthy attack on measurement j, we conclude that measurements with low ρ_j are relatively

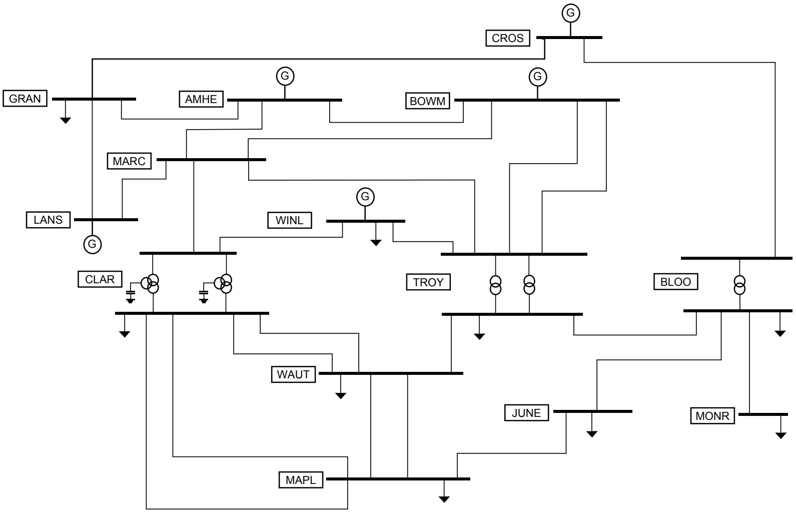


Figure 4.1: Power network considered in example.

easily attacked by coordinated attacks. Conversely, the measurements with large ρ_j are relatively protected, in the sense that stealthy false-data injection attacks corrupting them would require access to several measurements.

Recall that $\bar{\rho}_j$ is the security metric computed assuming that all possible measurements are being taken. Therefore, observing that $\bar{\rho}_j$ is larger than ρ_j, we conclude that increasing the redundancy of the system, by adding more measurements, increases the security level. However, note that this does not guarantee full protection, as all measurements with finite ρ_j still have finite $\bar{\rho}_j$.

Risk treatment approaches

One possible approach to decrease the risk of stealthy deception attacks is to encrypt the data and communication channels. Since a large part of today's power grid equipment is old, data encryption can be costly to implement because of the corresponding update of the equipment. Therefore, the following question is of great importance to measurement data integrity: given limited protection resources (the number of devices for data encryption), which measurements should be encrypted in order to maximize the benefits of the protection resources? The risk analysis outcome from computing the measurements' security indices may be used to sort the measurements in terms of their vulnerability and identify those that should be protected. In fact, a variant of the security index problem (4.7) can help provide

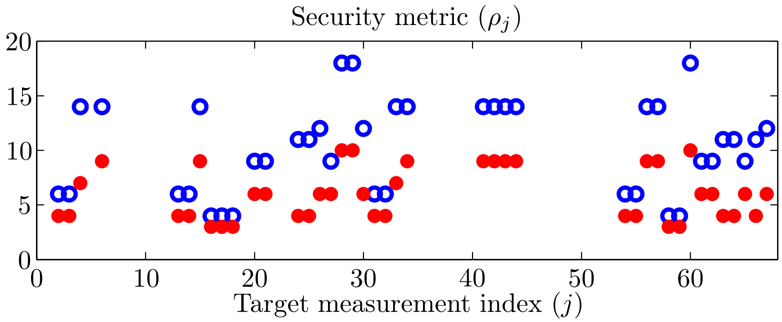


Figure 4.2: Security metrics for each measurement j: ρ_j (red full circles) was computed considering the default measurement configuration, while $\bar{\rho}_j$ (blue rings) was computed assuming that all possible measurements are taken. Both represent the minimum number of measurements needed to stealthily attack the target measurement j.

an answer to the previous question:

$$\begin{aligned} \underset{\Delta x \in \mathbb{R}^n}{\text{minimize}} \quad & \|C\Delta x\|_0 \\ \text{subject to} \quad & e_j^\top C\Delta x = 1, \\ & e_i^\top C\Delta x = 0, \forall i \in \mathcal{C}_p, \end{aligned}$$

where $\mathcal{C}_p$ is the index set of the encrypted measurements which cannot be attacked. By comparing the security indices for different index sets $\mathcal{C}_p$, it is possible to evaluate the effect of different protection strategies, and determine the best one to implement. For example, Vukovic *et al.* (2012) consider a lexicographic optimization of some security metrics which are based on the security index computation related to (4.6.1).

In the case where it is impractical to encrypt all measurements, it becomes critical to detect and isolate the measurements which are under attack. Effective attack isolation enables the damage control (e.g., removing attacked measurements for state estimation) to be performed in a timely fashion before the attack can lead to any incident with significant consequences. Sou *et al.* (2013a) present a distributed procedure for isolating the data attacks on power system transmission line power flow measurements, based on secure bus voltage magnitude measurements. The work by Kosut *et al.* (2011) develops a generalized likelihood ratio test to detect the presence of data attacks, based on the assumption that the normal measurements follow a known Gaussian distribution. Mechanisms to detect data attacks based on known-secure PMU measurements and known pattern of system states are presented in Giani *et al.* (2013).

4.6.2 Networked Control System Testbed

Next, we illustrate some of the proposed formulations for the quadruple-tank process (QTP) described in Section 2.5.2. The nonlinear plant model is linearized for a given operating point and sampled with a sampling period $T_s = 2\,s$. Recall that the state variable x_k corresponds to the water-levels in each tank, i.e. $x_k = [h_1\, h_2\, h_3\, h_4]^\top$. The QTP is controlled using a centralized LQG controller with integral action and a Kalman-filter-based anomaly detector is used so that alarms are triggered according to (4.1), for which we chose $\delta = 0.25$ for illustration purposes.

For the time-interval [0, 50], the maximum-impact minimum-resource attacks were computed for the process in minimum and non-minimum phase settings (with stable and unstable zeros, respectively) by choosing $p = q = 2$ and iteratively solving (4.14) with respect to ϵ. The respective impacts correspond to the energy of the state signal $\mathbf{x}$ for value of ϵ and are presented in Table 4.1, while the risk is depicted by the risk matrix plot in Figure 4.3a. As discussed in Section 2.3.3 for the risk matrix plot in Figure 2.10, the scenarios farther away from the origin have higher risk than those that are closer.

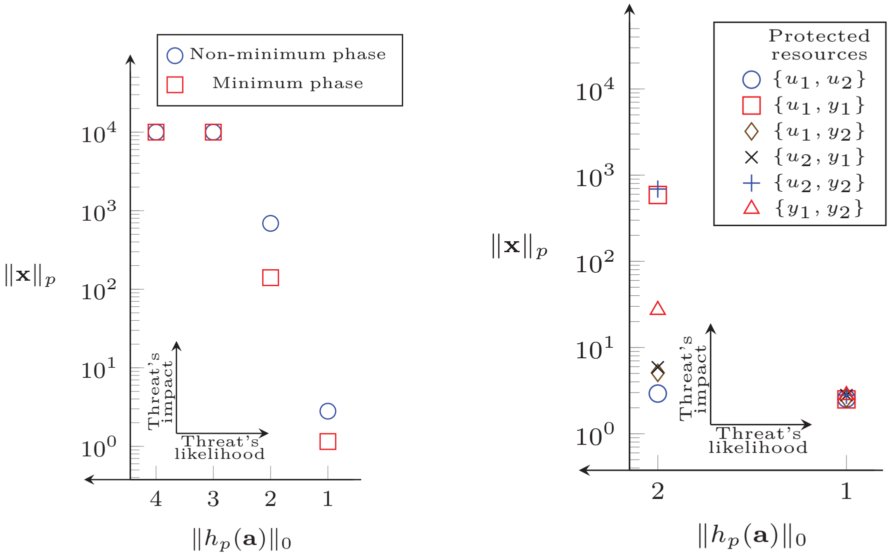
Table 4.1: Risk analysis results for the QTP. Each entry corresponds to the maximum impact $\|\mathbf{x}\|_p$ for a given number of corrupted channels, computed through (4.14) with $p = q = 2$ and $\delta = 0.15$.

	No. of compromised channels			
	4	**3**	**2**	**1**
Minimum phase	∞	∞	140.39	1.15
Non-minimum phase	∞	∞	689.43	2.80

As expected, due to the unstable zeros, the non-minimum phase system is less resilient than the minimum-phase one. In both settings, the attack impact can be made arbitrarily large by corrupting 3 or more channels, as explained next. Consider an adversary corrupting the two available outputs and one input, i.e. $\{y_1,\, y_2,\, u_i\}$ for $i = 1, 2$. Such an adversary may freely modify the input u_i while corrupting both outputs to remain stealthy, using for instance the replay attack illustrated in Section 3.6.2. Thus the adversary can drive the state out of the safe set while remaining stealthy.

The results in Table 4.1 indicate that the threats compromising 3 or more channels have high risk and should therefore be analyzed in more detail. The risk of such threats can be mitigated by protecting the data channels, which is performed in the next subsection.

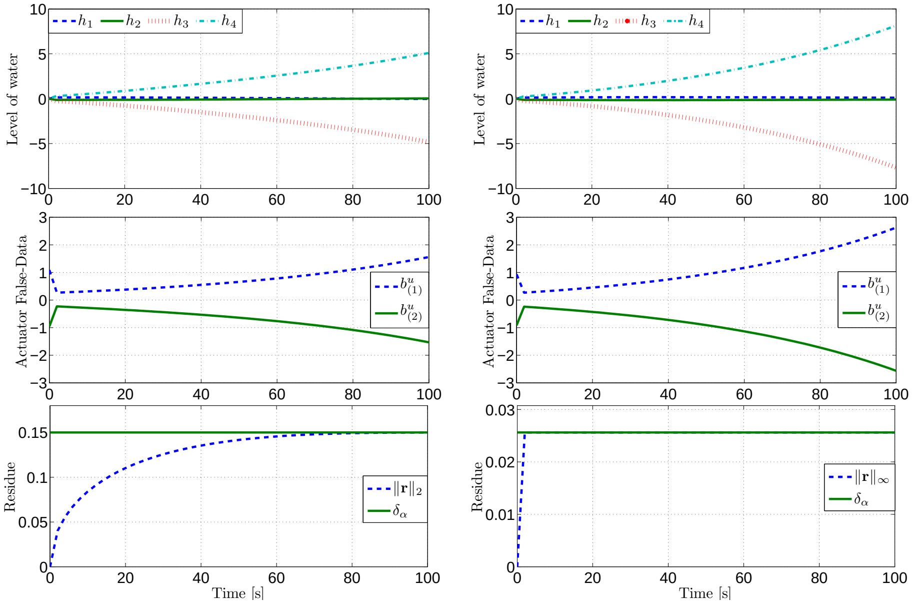
For illustration purposes, the maximum-impact attack signal for the non-minimum phase system with $\epsilon = 2$, $\delta = 0.15$, and $p = q = 2$ is presented in Figure 4.4a. Supposing that the adversary corrupts both actuators, the attack signal can be



(a) The risk matrix plot without protection.

(b) The risk matrix plot for the non-minimum phase case when different pairs of resources are protected.

Figure 4.3: The risk matrix plot for the QTP. The threat's likelihood is taken as a decreasing function of the number of compromised data channels, $\|h_p(\mathbf{a})\|_0$, and corresponds to the x-axis. The threat's impact on the y-axis is the p-norm of the state trajectory, $\|\mathbf{x}\|_p$. In (a), the risk analysis results for the minimum phase system (cross) and non-minimum phase (circle) from Table 4.1 are depicted and qualitatively classified. From (b) one concludes that, when pairs of resources can be protected in the non-minimum phase process, the most effective choice for risk treatment is to protect both actuator channels, $\{u_1, u_2\}$.



(a) Parameters: $p = q = 2$, $\delta = 0.15$.

(b) Parameters: $p = q = \infty$, $\delta = 0.025$.

Figure 4.4: Simulation results of the multi-objective problem (4.14) with $\epsilon = 2$ for the non-minimum phase system.

computed through Theorem 4.3.2. We highlight the similarity to the zero-dynamics attack signal used in the experiments reported in Section 3.6.2.

For the parameters $\epsilon = 2$, $\delta = 0.025$, and $p = q = \infty$, the maximum-impact attack signal was computed using the mixed-integer linear programming problem (4.22) and is shown in Figure 4.4b. In both cases, the optimal attack corrupts both actuator channels and ensures that no alarm is triggered, i.e. $\|\mathbf{r}\|_p \leq \delta$. Although the impact results in Table 4.1 do not consider the impact according to the safe set

$$\mathcal{S}_x = \{x \in \mathbb{R}^{n_x} : \|x\|_\infty \leq 5\},$$

the state trajectory does indeed leave the safe set in both cases. The attack signals illustrated in Figure 4.4 are related to the zero-dynamics of the QTP system, as illustrated in the zero-dynamics attack scenario in Section 3.6.

Risk treatment approaches

The risk analysis identifies the data channels that, when corrupted, may lead to a large impact on the system. The subsequent step in the risk management framework is the risk treatment stage, in which actions reducing the risk are chosen and implemented. A common approach to decrease the risk of threats is to deploy protective resources such as encryption, thus preventing the attacks from occurring. To assess the effectiveness of protecting a given set of data channels, $\mathcal{C}_p$, the optimization problem (4.14) may be modified as follows

$$\begin{aligned} \underset{\mathbf{a}}{\text{maximize}} \quad & \|\mathbf{x}\|_p \\ \text{subject to} \quad & \|\mathbf{r}\|_p \leq \delta, \\ & \|h_p(\mathbf{a})\|_0 < \epsilon, \\ & (4.8), (4.9), \\ & \mathbf{a}_{(i)} = 0, \quad \text{for all } i \in \mathcal{C}_p. \end{aligned} \tag{4.23}$$

The QTP example is now considered to illustrate the risk treatment step using channel encryption. The preventive action under study is the encryption of one pair of data channels, so that the risk is minimized. The optimization problem (4.23) is solved for each pair of data channels, and the corresponding risk matrices plots are depicted in Figure 4.3b.

Consider the results in Figure 4.3b for the case where two unprotected channels are corrupted ($\|h_p(\mathbf{a})\|_0 = 2$). We observe that the largest impact obtained from (4.23) occurs when $\{u_2,\, y_2\}$ and $\{u_1,\, y_1\}$ are protected. This means that attacking the unprotected channels ($\{u_1,\, y_1\}$ and $\{u_2,\, y_2\}$, respectively) yields a high impact. On the other hand, the smallest impact occurs when the channels $\{u_1,\, u_2\}$ are protected, meaning that attacking the outputs $\{y_1,\, y_2\}$ has a low impact. Therefore, we conclude that the pair of actuators $\{u_1,\, u_2\}$ should be protected to minimize the risk. Moreover, recalling the original risk matrix plot in Figure 4.3a, we observe that the maximum attack impact is substantially decreased by protecting $\{u_1,\, u_2\}$. Such a protection choice is expected, since the adversary can no longer inject an attack exciting the unstable zero-dynamics of the system when both actuators are protected. Furthermore, since the resources accessible to the adversary are y_1 and y_2, the adversary cannot have a direct impact on the physical system, but instead needs to affect the system through the feedback controller by corrupting the measurement signals.

Methods other than encryption have been proposed in the literature to reduce the risk of threats. Concerning replay attacks, (Chabukswar *et al.*, 2011) proposes the use of a hypothesis test as the anomaly detector and the injection of random zero-mean Gaussian noise with an optimally designed covariance in the control input channels. The injected noise increases the performance of the hypothesis test, since the noise statistics are assumed to be unknown to the adversary. Similarly, in Chapter 5 we propose the insertion of uncertainty in the adversary's model knowledge, by modifying the system dynamics and control and output channels. The effects of such actions on zero-dynamics attacks are also characterized in detail.

4.7 Summary

Several formulations for quantifying cyber security of networked control systems were proposed and formulated as constrained optimization problems, capturing trade-offs among adversary goals and constraints such as attack impact on the control system, attack detectability, and adversarial resources. Although the formulations are non-convex, some can be related to system-theoretic concepts such as invariant-zeros and modified $\mathcal{H}_\infty$-norm of the closed-loop system. The maximum-impact resource-constrained attack policy was also formulated as a mixed-integer linear program for a particular choice of parameters. The results were illustrated for the electric power network and quadruple-tank process.

Chapter 5

Revealing Stealthy Attacks in Networked Control Systems

In this chapter, we address stealthy data deception attacks that are constructed so that they cannot be detected based on control input and measurement data. In Chapter 4, we have identified the zero-dynamics attack as an instance of stealthy false-data injection attacks that may have high impact on the system, while remaining stealthy with respect to any time-invariant anomaly detector. As detailed in Chapter 3, a zero-dynamics attack has been performed on a networked control system testbed. The experiment showed that, although the attack is initially hard to detect, it is in fact detected when the system dynamics change due, for instance, to physical limitations. In the experiments on the quadruple-tank process, these changes occurred when the water-tanks became empty and the actuator reached saturation. Hence, changes in the system dynamics could be used to reveal stealthy false-data attacks. In essence, when prompted by the system operator, such changes create asymmetries in the information available to the adversary and operator. By exploiting this asymmetry, the operator is able to detect attacks that were previously undetectable. An example of this concept was used by Mo and Sinopoli (2009) to detect replay attacks, in which an auxiliary signal unknown to the attacker is used to excite the system.

Contributions and Related Work

In recent years, stealthy data deception attacks have been addressed from a system-theoretic perspective. Smith (2011) characterizes the set of attack policies for stealthy false-data injection attacks with detailed model knowledge and full access to all sensor and actuator channels, while Pasqualetti *et al.* (2011) described the set of stealthy false-data injection attacks for omniscient attackers with full-state information, but possibly compromising only a subset of the existing sensors and actuators.

The set of zero-dynamics attacks is considered in this chapter. As seen in Chapter 3, the zero-dynamics attack is an open-loop attack, in the sense that no online information is used to construct the attack. Hence, the attack policy is defined in terms of the available *a priori* information, namely the dynamical model of the system. In fact, this class of attacks was characterized in Chapter 3 using a property of the system known as zero-dynamics.

Using a geometric control framework, the system under a zero-dynamics attack is characterized as an autonomous dynamical system with a given initial condition. Furthermore, the attack detectability is cast as an observability property of the derived autonomous system. These two steps provide the basis of our results.

It is shown that zero-dynamics attacks may not be completely stealthy since they require the system to be at a non-zero initial condition. The effects of initial condition mismatch are then characterized and it is shown that they can be made arbitrarily small. The problem of changing the system structure to reveal the attacks is then considered. Specifically, we analyze how separately changing the outputs, system dynamics, and inputs affects the attacks' stealthiness. For each component, we characterize classes of changes that reveal attacks, as well as those that do not. Regarding changes on the system outputs, we provide an algorithm to reveal all attacks by incrementally adding new measurements. As for the inputs, we characterize the output effect of a scalar multiplicative perturbation to the inputs, assuming it remains unknown to the attacker. This particular perturbation can be interpreted as a coding or encryption scheme between the controller and actuator, having the scalar factor as their shared private key. Moreover, the corresponding contribution to the output energy is quantified as a function of the augmented system state, which can be used to determine a suitable scaling factor.

The outline of the chapter is as follows. The control system architecture and model under attack are described in Section 5.1. Section 5.2 follows with a geometric control characterization of zero-dynamics attacks and the effects of non-zero initial conditions are analyzed in Section 5.3. Different strategies to reveal zero-dynamics attacks are then proposed and analyzed in Section 5.4, followed by numerical examples illustrating our results. Summary and conclusions follow in Section 5.6.

5.1 Problem Formulation

For ease of reading, we recall the networked control system structure and zero-dynamics attack scenario presented in Chapter 3 and describe the main problem to be addressed.

The physical plant, feedback controller, and anomaly detector are modeled in a discrete-time state-space form, respectively, as

$$\mathcal{P}:\begin{cases} x_{k+1} &= Ax_k + B\tilde{u}_k \\ y_k &= Cx_k \end{cases} \tag{5.1}$$

$$\mathcal{F}:\begin{cases} z_{k+1} &= A_c z_k + B_c\tilde{y}_k \\ u_k &= C_c z_k + D_c\tilde{y}_k \end{cases}$$

$$\mathcal{D}:\begin{cases} s_{k+1} &= A_e s_k + B_e u_k + K_e\tilde{y}_k \\ r_k &= C_e s_k + D_e u_k + E_e\tilde{y}_k \end{cases}$$

where $x_k \in \mathbb{R}^{n_x}$, $z_k \in \mathbb{R}^{n_z}$, and $s_k \in \mathbb{R}^{n_s}$ are the state variables, $\tilde{u}_k \in \mathbb{R}^{n_u}$ the control actions applied to the process, $y_k \in \mathbb{R}^{n_y}$ the measurements from the sensors, and $r_k \in \mathbb{R}^{n_r}$ the residue vector. The sensor measurements and actuator data are transmitted through a communication network, which at the plant side correspond to y_k and $\tilde{u}_k$, respectively. At the controller side we denote the sensor and actuator data by $\tilde{y}_k \in \mathbb{R}^{n_y}$ and $u_k \in \mathbb{R}^{n_u}$, respectively.

The anomaly detector is collocated with the controller and therefore it only has access to $\tilde{y}_k$ and u_k to evaluate the behavior of the plant. In particular, given the time-interval $[k_0,\, k_f]$ and the residue signal $\mathbf{r}_{[k_0,\, k_f]}$, an alarm is triggered if the residue meets

$$\|\mathbf{r}_{[k_0,\, k_f]}\|_p \geq \delta,$$

where $\delta \geq 0$ is chosen according to a suitable trade-off between detection and false alarm rates and $p \geq 1$.

5.1.1 Attack Scenario: Data Deception

The attack scenario and adversary model are described in the remainder of this section.

Disruption and disclosure resources: In the present scenario, the attacker is able to inject false data in the actuator and measurement channels, which is captured by having

$$\begin{bmatrix} \tilde{u}_k \\ \tilde{y}_k \end{bmatrix} = \begin{bmatrix} u_k \\ y_k \end{bmatrix} + \begin{bmatrix} B_a \\ D_a \end{bmatrix} a_k,$$

where $a_k \in \mathbb{R}^{n_a}$ is the attack vector. However, the attacker cannot eavesdrop on the sensor and actuator data. Hence, the corresponding attack policy does not use any online data on the system and is further assumed to be computed *a priori*. Therefore, it corresponds to an open-loop type of policy.

Model knowledge: The attacker also has access to the detailed model of the plant $\mathcal{P} = (A, B, C)$, which is used to compute the attack policy.

Attack goals and constraints: Recall from Chapter 3 that the adversary aims at disrupting the system behavior while remaining stealthy. Next, we characterize the set of stealthy attacks considered in this chapter.

Stacking the states of the plant, controller, and anomaly detector as $\xi_k = [x_k^\top \; z_k^\top \; s_k^\top]^\top$, the closed-loop dynamics under attack can be written as

$$\begin{aligned}
\xi_{k+1} &= \underbrace{\begin{bmatrix} A+BD_cC & BC_c & 0 \\ B_cC & A_c & 0 \\ (B_eD_c+K_e)C & B_eC_c & A_e \end{bmatrix}}_{\mathbf{A}} \xi_k + \underbrace{\begin{bmatrix} BB_a+BD_cD_a \\ B_cD_a \\ (B_eD_c+K_e)D_a \end{bmatrix}}_{\mathbf{B}} a_k \\
r_k &= \underbrace{\begin{bmatrix} (D_eD_c+E_e)C & D_eC_c & C_e \end{bmatrix}}_{\mathbf{C}} \xi_k + \underbrace{(D_eD_c+E_e)D_a}_{\mathbf{D}} a_k \\
\tilde{y}_k &= \underbrace{\begin{bmatrix} C & 0 & 0 \end{bmatrix}}_{\mathbf{C}_y} \xi_k + D_a a_k.
\end{aligned} \tag{5.2}$$

Consider that the attack starts at $k = k_0$ and that the system is at the zero initial condition, i.e. $\xi_{k_0} = 0$. Denoting $\mathbf{a}_{[k_0,\, k_f]} = \{a_{k_0}, \ldots, a_{k_f}\}$ as the attack signal, the set of stealthy attacks are defined with respect to the decomposed system (5.2) as follows.

Definition 5.1.1. *The attack signal $\mathbf{a}_{[k_0,\, k_f]}$ is δ-stealthy with respect to the anomaly detector $\mathcal{D}$ if $\|\mathbf{r}_{[k_0,\, +\infty)}\|_p \leq \delta$.*

A particular subset of 0-stealthy attacks is characterized in the following lemma.

Lemma 5.1.1. *Consider the output $\tilde{y}_k$ of the closed-loop system* (5.2) *with $\xi_{k_0} = 0$. The attack signal $\mathbf{a}_{[k_0,\, k_f]}$ is 0-stealthy with respect to any output feedback controller $\mathcal{F}$ and anomaly detector $\mathcal{D}$ if $\tilde{y}_k = 0$ for all $k \geq k_0$.*

Proof. The proof follows directly from considering the subsystem composed of the feedback controller and anomaly detector and observing that $\tilde{y}_k$ is the only input to this subsystem. Hence, given the initial condition $\xi_{k_0} = 0$, having $\tilde{y}_k = 0$ for all $k \geq k_0$ results in a zero residual signal. □

Attack policy: The subset of 0-stealthy attacks satisfying the conditions in Lemma 5.1.1 results in trajectories of the system that do not affect $\tilde{y}_k$. Therefore, using only the plant model $\mathcal{P}$, such a set of attacks can be characterized as the attack signals that render the output $\tilde{y}_k$ identically zero. For linear systems, the 0-stealthy attack signals are related to the output zeroing problem or zero-dynamics studied in the control theory literature (Tokarzewski, 2006), which we revisit in the next section. In fact, this attack policy is used in the zero-dynamics attacks described in Chapter 3.

5.1.2 Revealing Attacks

In this chapter, the main goal is to devise methods to reveal 0-stealthy attacks characterized in Definition 5.1.1, by modifying the system dynamics. To that end, the following definition of revealed attacks is considered throughout this chapter.

Definition 5.1.2. *Consider the system under a* 0*-stealthy attack characterized by Lemma 5.1.1. The* 0*-stealthy attack signal* $\mathbf{a}_{[k_0,\, k_f]}$ *is said to be revealed if* $\tilde{y}_k \neq 0$ *for some* $k \geq k_0$.

The latter definition can be extended to also account for the anomaly detector, by stating that an attack is revealed when $r_k \neq 0$ for some $k \geq k_0$. In fact, based on similar arguments as in Lemma 5.1.1, the anomaly detector may be designed so that a revealed attack yielding $\tilde{y}_k \neq 0$ for some $k \geq k_0$ leads to a non-zero residue signal.

Next, we provide a geometric characterization of a system's zero-dynamics, which is instrumental to analyze how changes on the system's dynamics affect the stealthiness properties of the zero-dynamics attacks.

5.2 Geometric Control Characterization of Zero-Dynamics

Recalling Lemma 5.1.1, the zero-dynamics attacks can be analyzed by considering the plant dynamics due to the false-data injection attack. The set of zero-dynamics attacks with $D_a = 0$ and $B_a = B$ in (5.2) are now characterized under a geometric control framework (Basile and Marro, 1992).

Remark 5.2.1. *The case for* $D_a \neq 0$ *can be analyzed in a similar fashion when the set of system zeros is finite (Tokarzewski, 2006).*

Consider the linear time-invariant system $\mathcal{P} = (A, B, C)$. In general, the matrices B and C may have linearly dependent columns and rows, respectively. This means that there exists some redundancy in the actuators and sensors, in the sense that removing one actuator or sensor does not affect the controllability and observability properties of the system. While this concept of redundancy is explored in Chapter 7, dealing with it here would involve several additional technicalities. Therefore, for the sake of a clear presentation of our results, in this chapter we make the following assumptions.

Assumption 5.2.1. *The matrix* B *has full column-rank and* C *has full row rank. Moreover,* $\mathcal{P}$ *is the minimal realization of the system.*

We now introduce the necessary concepts from geometric control theory (Basile and Marro, 1992) to describe the zero-dynamics. Let $\mathscr{Z}$ and $\mathscr{X}$ be subspaces contained in $\mathbb{C}^{n_x}$. In the following, we denote $\mathscr{X} \subseteq \mathscr{Z}$ as the set inclusion of $\mathscr{X}$ by $\mathscr{Z}$, i.e. for all $x \in \mathscr{X}$, it holds that $x \in \mathscr{Z}$. Moreover, we denote $\mathscr{Y} = \mathscr{Z} + \mathscr{X}$ as the union of $\mathcal{Z}$ and $\mathcal{X}$, defined as $\mathcal{Y} = \{x \in \mathbb{C}^{n_x} : \ x \in \mathcal{X} \text{ or } x \in \mathcal{Z}\}$. Given a matrix $A \in \mathbb{C}^{n_x \times n_x}$ and a subspace $\mathcal{Z} \subseteq \mathbb{C}^{n_x}$, we define the subspace $A\mathcal{Z}$ as $A\mathcal{Z} \triangleq \{y \in \mathbb{C}^{n_x} : \exists x \in \mathcal{Z} : \ y = Ax\}$. Additionally, we say that $\mathcal{Z}$ is A-invariant if $A\mathcal{Z} \subseteq \mathcal{Z}$. Note that the eigenspace of A is the maximal A-invariant subspace, i.e. any A-invariant subspace is contained in the eigenspace of A.

First, controlled invariant subspaces are characterized as follows.

Lemma 5.2.1. *For a given non-empty subspace $\mathcal{Z}$ for which $A\mathcal{Z} \subseteq \mathcal{Z} + \operatorname{Im}(B)$ holds, there exists a matrix F such that $(A + BF)\mathcal{Z} \subseteq \mathcal{Z}$. Furthermore, $\mathcal{Z}$ is called an $\big(A, \operatorname{Im}(B)\big)$-controlled invariant subspace.*

The subset of controlled invariant subspaces contained in $\operatorname{Ker}(C)$ is the basis for characterizing the system's zero-dynamics, as summarized in the next statement.

Lemma 5.2.2. *There exist an initial condition $x_0 \neq 0$ and control input a_k such that $\tilde{y}_k = 0$ for all $k \geq 0$ if and only if there exists a non-empty $\big(A, \operatorname{Im}(B)\big)$-controlled invariant subspace $\mathcal{Z}$ contained in $\operatorname{Ker}(C)$, i.e., there exits $\mathcal{Z}$ satisfying $A\mathcal{Z} \subseteq \mathcal{Z} + \operatorname{Im}(B)$ and $\mathcal{Z} \subseteq \operatorname{Ker}(C)$.*

The set of all subspaces $\mathcal{Z}$ satisfying the conditions of Lemma 5.2.2 admits a maximum, $\mathcal{Z}^\star$, which we denote by the maximal output-nulling invariant subspace. A procedure to compute $\mathcal{Z}^\star$ can be found in Basile and Marro (1992).

Using the maximal output-nulling invariant subspace, one can compute all output-nulling input signals that generate identically zero output signals, i.e., $\tilde{y}_k = 0$ for all $k \geq 0$. In fact, the output-nulling inputs of the plant (5.1) can be characterized as the output of an autonomous dynamical system, as stated in the following theorem.

Theorem 5.2.3. *Consider the plant* (5.1) *with an initial condition x_0. For the initial condition $x_0 = \tilde{x}_0$, the input $a_k = F\tilde{x}_k$ with $\tilde{x}_{k+1} = (A + BF)\tilde{x}_k$, $(A + BF)\mathcal{Z}^\star \subseteq \mathcal{Z}^\star \subseteq \operatorname{Ker}(C)$, and $\tilde{x}_0 \in \mathcal{Z}^\star$ yields $x_k \in \mathcal{Z}^\star$ and $\tilde{y}_k = 0$ for all $k \geq 0$.*

The invariant-zeros of a system, characterized in Definition 4.4.1, are related to the matrix $A + BF$ and the subspace $\mathcal{Z}^\star$, as described next. Denote $\sigma\left(A|_{\mathcal{Z}}\right)$ as the eigenvalues of A whose associated eigenvectors belong to the subspace $\mathcal{Z}$, i.e. $\sigma\left(A|_{\mathcal{Z}}\right) \triangleq \left\{\lambda \in \mathbb{C} : \ \exists x \in \mathcal{Z} : \ (\lambda I - A)\, x = 0\right\}$. Given the previous definition, the invariant-zeros of the system $\mathcal{P}$ correspond to $\sigma\left((A + BF)|_{\mathcal{Z}^\star}\right)$, i.e. the eigenvalues of $A + BF$ whose associated eigenvectors belong to $\mathcal{Z}^\star$. In fact, note that the equation characterizing the invariant-zeros of (A, B, C), i.e.

$$\begin{bmatrix} \lambda_z I - A & -B \\ C & 0 \end{bmatrix} \begin{bmatrix} x_0 \\ u_z \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix},$$

is satisfied when the tuple (λ_z, u_z, x_0) is chosen such that $\lambda_z \in \sigma\left((A + BF)|_{\mathcal{Z}^\star}\right)$, $x_0 \in \mathcal{Z}^\star$ is the eigenvector of $A + BF$ associated with λ_z, and $u_z = Fx_0$.

The zero-dynamics attack policy readily follows from Theorem 5.2.3.

Corollary 5.2.4. *The zero-dynamics attack policy is characterized by*

$$\begin{aligned} \tilde{x}_{k+1} &= (A + BF)\tilde{x}_k \\ a_k &= F\tilde{x}_k, \end{aligned} \tag{5.3}$$

with $\tilde{x}_0 \in \mathcal{Z}^\star$ and F such that $(A + BF)\mathcal{Z}^\star \subseteq \mathcal{Z}^\star$.

Note that the zero-dynamics characterized in Corollary 5.2.4 require the initial condition to be non-zero and belong to $\mathcal{Z}^\star$. Such a requirement contradicts the definition of 0-stealthy attacks in Lemma 5.1.1, where the initial condition of the system component under attack is the origin. The effect of having non-zero initial conditions is addressed in the next section.

5.3 Effects of Non-Zero Initial Conditions

Note that the zero-dynamics do not match the definition of 0-stealthy attacks, since a non-zero initial condition for the plant (5.1) is required. However, in some cases the effects of the initial condition may be made arbitrarily small, as discussed below.

Using Corollary 5.2.4, the system under a zero-dynamics attack is described by

$$\begin{aligned} \begin{bmatrix} x_{k+1} \\ \tilde{x}_{k+1} \end{bmatrix} &= \begin{bmatrix} A & BF \\ 0 & A + BF \end{bmatrix} \begin{bmatrix} x_k \\ \tilde{x}_k \end{bmatrix} \\ \tilde{y}_k &= \begin{bmatrix} C & 0 \end{bmatrix} \begin{bmatrix} x_k \\ \tilde{x}_k \end{bmatrix} \end{aligned} \tag{5.4}$$

with $\tilde{x}_0 \in \mathcal{Z}^\star$. For $x_0 = \tilde{x}_0$ it directly follows that $\tilde{y}_k = 0$ for all $k \geq 0$. Introducing the error variable $e_k = x_k - \tilde{x}_k$, the previous system may be rewritten as

$$\begin{aligned} \begin{bmatrix} e_{k+1} \\ \tilde{x}_{k+1} \end{bmatrix} &= \begin{bmatrix} A & 0 \\ 0 & A + BF \end{bmatrix} \begin{bmatrix} e_k \\ \tilde{x}_k \end{bmatrix} \\ \tilde{y}_k &= \begin{bmatrix} C & 0 \end{bmatrix} \begin{bmatrix} e_k \\ \tilde{x}_k \end{bmatrix} \end{aligned} \tag{5.5}$$

with $\tilde{x}_0 \in \mathcal{Z}^\star$ and $e_0 = x_0 - \tilde{x}_0$. The next result readily follows.

Theorem 5.3.1. *For a zero initial condition $x_0 = 0$, a zero-dynamics attack generated by $\tilde{x}_0 \in \mathcal{Z}^\star$ yields the output characterized by*

$$\begin{aligned} e_{k+1} &= Ae_k \\ \tilde{y}_k &= Ce_k \end{aligned}$$

with $e_0 = -\tilde{x}_0$.

The previous result allows us to characterize conditions under which the output and residue signal energy caused by zero-dynamics attacks can be made arbitrarily small. These conditions are related to the stable eigenvalues of A, where the complex number $\lambda \in \mathbb{C}$ is said to be unstable if $|\lambda| \geq 1$ and stable if $|\lambda| < 1$. Moreover, we consider the coordinate transform $e_k = Tv_k$, where $T = [T_s \; T_u]$ is a basis for the eigenspace of A, with $T_s \in \mathbb{C}^{n_x \times n_v}$ and $T_u \in \mathbb{C}^{n_x \times n_x - n_v}$ being associated with the stable and unstable eigenvalues of A, respectively. The dynamics under the coordinate transform are described by $v_{k+1} = \Lambda v_k$, where Λ is the Jordan block matrix of A containing its eigenvalues. Given the structure of T, Λ can be written as

$$\Lambda = \begin{bmatrix} \Lambda_s & 0 \\ 0 & \Lambda_u \end{bmatrix},$$

where Λ_s and Λ_u contain the stable and unstable eigenvalues of A, respectively. Furthermore, the output of the autonomous system in Theorem 5.3.1 may be characterized as

$$\begin{aligned} v_{k+1} &= \Lambda v_k \quad , \; v_0 = T^{-1} e_0 \\ \tilde{y}_k &= CTv_k. \end{aligned} \tag{5.6}$$

Using the previous definitions, first we derive the results for the open-loop system, which are then directly applied to the closed-loop dynamics.

Corollary 5.3.2. *Consider the open-loop system* (5.1)*. The output of a zero-dynamics attack generated by $\tilde{x}_0 \in \mathcal{Z}^\star$ with $x_0 = 0$ has finite energy if and only if $[0_{n_v} \; I_{n_x - n_v}]T^{-1}\tilde{x}_0 = 0$.*

Proof. From Theorem 5.3.1, we have that the autonomous system (5.6) characterizes the output of (5.1) under a zero-dynamics attack generated by $\tilde{x}_0 \in \mathcal{Z}^\star$ and with $x_0 = 0$.

Recall that the system (5.6) is assumed to be observable, since (A, B, C) is a minimal realization. Thus, by definition of observability, any initial condition affects the output of the autonomous system (5.6). Furthermore, initial conditions exciting unstable modes lead to unbounded output energy. On the other hand, initial conditions that only excite stable modes lead to state trajectories that decay asymptotically to zero, thus having finite output energy. Therefore, we conclude that the output energy of (5.1) is finite if and only if $e_0 = -\tilde{x}_0$ only excites stable modes of (5.6). The proof concludes by observing that initial conditions satisfying $[0_{n_v} \; I_{n_x - n_v}]T^{-1}\tilde{x}_0 = 0$ only excite stable modes of (5.6). □

Now we analyze the case where $\tilde{x}_0$ does not excite unstable eigenvectors of A, i.e. $\tilde{x}_0$ satisfies the equality $[0_{n_v} \; I_{n_x - n_v}]T^{-1}\tilde{x}_0 = 0$. Supposing that $\tilde{x}_0$ only excites stable eigenvalues of A, the output of (5.6) may be characterized as

$$\begin{aligned} v_{s_{k+1}} &= \Lambda_s v_{s_k} \\ \tilde{y}_k &= CT_s v_{s_k} \end{aligned} \tag{5.7}$$

where $v_k = [v_{s_k}^\top \; v_{u_k}^\top]^\top$ with $v_{s_0} = [I_{n_v} \; 0_{n_x - n_v}]T^{-1}\tilde{x}_0$ and $v_{u_0} = [0_{n_v} \; I_{n_x - n_v}]T^{-1}\tilde{x}_0 = 0$. This leads to the following result.

Corollary 5.3.3. *Consider a zero-dynamics attack generated by $\tilde{x}_0 \in \mathcal{Z}^\star$ with $\tilde{x}_0$ such that $[0_{n_v} \; I_{n_x - n_v}]T^{-1}\tilde{x}_0 = 0$ and let $x_0 = 0$. The output energy of the attack is given by $\|\mathbf{y}\|_2^2 = \tilde{x}_0^\top \bar{Q}\tilde{x}_0$ where*

$$\bar{Q} = T^{-\top} \begin{bmatrix} I_{n_v} \\ 0_{n_x - n_v} \end{bmatrix} Q_s \begin{bmatrix} I_{n_v} & 0_{n_x - n_v} \end{bmatrix} T^{-1}$$

and $Q_s \succeq 0$ is the solution to

$$\Lambda_s^\top Q_s \Lambda_s - Q_s + T_s^\top C^\top C T_s = 0.$$

Proof. Using the transform T, decompose the open-loop system matrix A in its stable and unstable components, Λ_s and Λ_u, respectively. Since $\tilde{x}_0$ does not excite unstable components, $\|\mathbf{y}\|_2^2$ can be computed from the stable component (5.7). Denoting $Q_s \succeq 0$ as the observability Gramian of (5.7), the output energy is computed as $\|\mathbf{y}\|_2^2 = v_{s_0}^\top Q_s v_{s_0}$. The proof concludes by recalling that Q_s can be computed through the Lyapunov equation above and using the equality $v_{s_0} = [I_{n_v} \; 0_{n_x - n_v}]T^{-1}\tilde{x}_0$. □

The former analysis can be directly extended to the closed-loop system (5.2) through the following results.

Theorem 5.3.4. *Consider the closed-loop system (5.2), which is assumed to be stabilized by a suitable controller, and define $\varepsilon_k = [e_k^\top \; z_k^\top \; s_k^\top]^\top$. For a zero initial condition $\xi_0 = 0$, a zero-dynamics attack generated by $\tilde{x}_0 \in \mathcal{Z}^\star$ yields the residue characterized by*

$$\begin{aligned} \varepsilon_{k+1} &= \mathbf{A}\varepsilon_k \\ r_k &= \mathbf{C}\varepsilon_k, \end{aligned}, \quad \varepsilon_0 = \begin{bmatrix} -\tilde{x}_0 \\ 0 \\ 0 \end{bmatrix}.$$

Proof. The proof follows directly from combining the closed-loop dynamics (5.2) with the zero-dynamics attack in (5.5). □

Corollary 5.3.5. *Consider a zero-dynamics attack generated by $\tilde{x}_0 \in \mathcal{Z}^\star$. The residue signal energy of the attack is given by $\|\mathbf{r}\|_2^2 = \varepsilon_0^\top \mathbf{Q}\varepsilon_0$, where $\varepsilon_0 = [-\tilde{x}_0 \, 0 \, 0]^\top$ and $\mathbf{Q} \succeq 0$ is the solution to*

$$\mathbf{A}^\top \mathbf{Q}\mathbf{A} - \mathbf{Q} + \mathbf{C}^\top \mathbf{C} = 0.$$

Proof. The proof follows with similar arguments as in Corollaries 5.3.2 and 5.3.3 and using the fact that the closed-loop system $\mathbf{A}$ is stable. □

From Corollary 5.3.5, we conclude that the residue signal energy caused by zero-dynamic attacks can be made arbitrarily small by selecting a sufficiently small initial condition $\tilde{x}_0 \in \mathcal{Z}^\star$ to generate the attack. Such attacks are particularly dangerous if the initial condition $\tilde{x}_0$ excites an unstable eigenvalue of $A + BF$, as illustrated in the numerical example in Section 5.5. This motivates us to consider schemes to reveal these attacks.

5.4 Revealing Zero-Dynamics Attacks

In this section, we discuss possible methods to reveal the zero-dynamics attacks characterized in Section 5.2.

Given Definition 5.1.2, a 0-stealthy attack is revealed if it generates a non-zero output signal. As seen in the previous section, all zero-dynamics attacks generate a non-zero output signal, due to the non-zero initial condition that is used to compute the attack signal. Therefore, as per Definition 5.1.2, all zero-dynamics attacks are revealed. However, recalling Corollaries 5.3.2 and 5.3.5, there exist certain conditions under which the residue and output generated by zero-dynamics attacks can be made arbitrarily small. This is particularly relevant when the corresponding zero-dynamics are unstable, since the magnitude of the attack signal increases exponentially. Motivated by these arguments, we tackle the problem of revealing zero-dynamics attacks while assuming that they are indeed 0-stealthy attacks. In other terms, in this section we let the system's initial condition and the zero-dynamics initial condition be the same, i.e. $x_0 = \tilde{x}_0$.

As per Definition 5.1.2, a zero-dynamics attack is revealed if the corresponding attack signal $\mathbf{a}_{[k_0,\,k_f]}$ no longer matches the zero-dynamics of the system. As it is well-known in the control literature (Skogestad and Postlethwaite, 1996), the system zeros cannot be changed by state- or output-feedback policies. However, state-feedback laws $u_k = Kx_k$ can indeed modify the input-zero direction, and thus the zero-dynamics, of the closed-loop system $\tilde{\mathcal{P}} = (A + BK, B, C)$ so that (5.3) is no longer an output-nulling input of the resulting system $\tilde{\mathcal{P}}$.

A more general approach is to modify the system $\mathcal{P} = (A, B, C)$ in a certain way to obtain $\tilde{\mathcal{P}} = (\tilde{A}, \tilde{B}, \tilde{C})$, so that the attack signal (5.3) is no longer an output-nulling input of the resulting system

$$\begin{aligned}\begin{bmatrix} x_{k+1} \\ \tilde{x}_{k+1} \end{bmatrix} &= \begin{bmatrix} \tilde{A} & \tilde{B}F \\ 0 & A+BF \end{bmatrix} \begin{bmatrix} x_k \\ \tilde{x}_k \end{bmatrix} \\ \tilde{y}_k &= \begin{bmatrix} \tilde{C} & 0 \end{bmatrix} \begin{bmatrix} x_k \\ \tilde{x}_k \end{bmatrix}. \end{aligned} \tag{5.8}$$

This formulation includes the aforementioned state-feedback approach, as we see by having $\tilde{A} = A + BK$. Since (5.8) is an autonomous system, the concept of unobservability is tightly related to the stealthiness of attacks.

Definition 5.4.1. *The autonomous system*

$$\begin{aligned} x_{k+1} &= Ax_k \\ y_k &= Cx_k \end{aligned}$$

is unobservable *if there exists a non-empty subspace $\mathcal{X} \subseteq \mathbb{C}^{n_x}$ such that all initial conditions $x_0 \in \mathcal{X}$ yield $y_k = 0$ for all $k \geq 0$. Moreover, the subspace $\mathcal{X}$ is called an* unobservable subspace*. The largest unobservable subspace $\mathcal{X}^\star$ is defined as the maximal A-invariant subspace contained in* $\ker C$*. Let $\{\lambda_i\}$ and $\{x_i\}$ be the set of all eigenvalues and corresponding normalized eigenvectors of A satisfying*

$$\begin{bmatrix} \lambda_i I - A \\ C \end{bmatrix} x_i = \begin{bmatrix} 0 \\ 0 \end{bmatrix},$$

which relates to the PBH observability test (Zhou et al., 1996, Theorem 3.4). The subspace $\mathcal{X}^\star$ can be computed $\mathcal{X}^\star = span(\{x_i\})$.

Considering the autonomous system (5.8) describing the plant under zero-dynamics attacks, note that the concept of unobservability in Definition 5.4.1 is quite similar to the properties of 0-stealthy attacks described in Definition 5.1.1, in the sense that both yield a zero output. This relation is leveraged in the next result to characterize when zero-dynamics attacks are revealed.

Lemma 5.4.1. *Every zero-dynamics attack is revealed if and only if the system* (5.8) *is observable for all $x_0 = \tilde{x}_0 \in \mathcal{Z}^\star$.*

Proof. Consider the system (5.8) with the initial condition $w_0 = [\tilde{x}_0^\top \; \tilde{x}_0^\top]^\top$, where $\tilde{x}_0 \in \mathcal{Z}^\star$. By Definition 5.4.1, a given subspace $\mathcal{X}_d \subseteq \mathbb{C}^{2n_x}$ is observable if and only if all initial conditions $w_0 \in \mathcal{X}_d$ yield $\tilde{y}_k \neq 0$ for some $k \geq 0$. Given Definition 5.1.2, w_0 being observable implies that the corresponding attack is revealed, since $\tilde{\mathbf{y}} \neq 0$. □

Attacks remaining stealthy after the perturbation can also be characterized using similar arguments.

Corollary 5.4.2. *Consider a zero-dynamics attack generated by $x_0 \in \mathcal{Z}^\star$. The former attack remains stealthy after the perturbation if and only if $w_0 = [x_0^\top \; x_0^\top]^\top$ belongs to the unobservable subspace of the system* (5.8)*.*

Proof. Suppose x_0 is an eigenvector of $A + BF$, without loss of generality, and consider the augmented system before the perturbation as in (5.4). Since the state trajectories of (5.4) generated by the attack are contained in span(w_0), the state when the perturbation occurs can be written as $\tilde{w}_0 = \alpha w_0$, for a given $\alpha \in \mathbb{C}$. The remaining of the proof follows from Definition 5.1.2 and the notion of unobservable subspace in Definition 5.4.1. □

A less restrictive condition for revealing the set of zero-dynamics attacks associated with unstable zeros follows from the above theorem.

Corollary 5.4.3. *Every unstable zero-dynamics attack is revealed if and only if the system* (5.8) *is detectable for all* $x_0 = \tilde{x}_0 \in \mathcal{Z}^\star$.

Using the observability concepts presented in this subsection, next we propose schemes to reveal the zero-dynamics attacks by separately changing A, B, or C. In fact, the results from Lemma 5.4.1 and Corollary 5.4.2 are instrumental for the discussions that follow below.

5.4.1 Modifying the Output Matrix C

Here, we consider modifications on the output matrix C to reveal zero-dynamics attacks. In particular, we consider that a new output matrix $\tilde{C}$ is obtained by adding and removing measurements. The following result directly follows from Theorem 5.2.3.

Lemma 5.4.4. *All the zero-dynamics attacks associated with a given* $\tilde{x}_0 \in \mathcal{Z}^\star$ *remain stealthy with respect to* $\tilde{\mathcal{P}} = (A, B, \tilde{C})$ *if and only if* $\mathcal{Z}^\star \subseteq \operatorname{Ker}\left(\tilde{C}\right)$.

Note that obtaining $\tilde{C}$ by removing measurements from C yields $\operatorname{Ker}(C) \subseteq \operatorname{Ker}\left(\tilde{C}\right)$. Hence, the latter result shows that only removing measurements does not reveal any attack, since the following relation holds $\mathcal{Z}^\star \subseteq \operatorname{Ker}(C) \subseteq \operatorname{Ker}\left(\tilde{C}\right)$. Moreover, attacks are revealed by adding measurements if and only if the dimension of $\mathcal{Z}^\star \cap \operatorname{Ker}\left(\tilde{C}\right)$ is reduced. The next result characterizes under what conditions do there exist zero-dynamics attacks after modifying the output matrix.

Theorem 5.4.5. *There exists a* $\tilde{x}_0 \in \mathcal{Z}^\star$ *generating a zero-dynamics attack to* $\tilde{\mathcal{P}} = (A, B, \tilde{C})$ *if and only if there exists a non-empty* $(A + BF)$*-invariant subspace* $\mathcal{X}$ *that is contained in* $\mathcal{Z}^\star \cap \operatorname{Ker}\left(\tilde{C}\right)$.

Proof. Consider a non-empty subspace $\mathcal{X} \subseteq \mathcal{Z}^\star$, from which an initial condition $\tilde{x}_0$ is chosen to generate an open-loop zero-dynamics attack. From Theorem 5.2.3, we have that the attack generated by $\tilde{x}_0 \in \mathcal{X}$ is stealthy if and only if $\mathcal{X}$ is $(A + BF)$-invariant and $\mathcal{X} \subseteq \operatorname{Ker}\left(\tilde{C}\right)$. The latter condition and $\mathcal{X} \subseteq \mathcal{Z}^\star$ can be replaced by the equivalent condition $\mathcal{X} \subseteq \mathcal{Z}^\star \cap \operatorname{Ker}\left(\tilde{C}\right)$, which conclude the proof. □

The previous results indicate that one should modify the measurement matrix such that the dimension of $\mathcal{X} \subseteq \mathcal{Z}^\star \cap \operatorname{Ker}\left(\tilde{C}\right)$ is reduced as much as possible. In particular, having $\dim(\mathcal{X}) < \dim(\mathcal{Z}^\star)$ indicates that a subset of the zero-dynamics attacks has been revealed, while $\mathcal{X} = \emptyset$ implies that none of the zero-dynamics attacks remain stealthy.

Based on these arguments, Algorithm 5.1 can be used to incrementally deploy measurements that reveal zero-dynamics attacks.

Algorithm 5.1 Algorithm to deploy additional measurements revealing zero-dynamics attacks.

```
 1: Initialize $\mathcal{M} \leftarrow \{C_i\}$ as the set of additional measurements available;
 2: $j \leftarrow 0$;
 3: $\mathcal{X}_0 \leftarrow \mathcal{Z}^\star$;
 4: repeat
 5:     for all $C_i \in \mathcal{M}$ do
 6:         $\mathcal{Y}_i \leftarrow \mathcal{X}_j \cap \mathrm{Ker}\,(C_i)$;
 7:     end for
 8:     Choose $C_i \in \mathcal{M}$ such that $\dim(\mathcal{Y}_i)$ is minimized;
 9:     Compute $\mathcal{X}_{j+1}$ as the maximal $(A + BF)$-invariant contained in $\mathcal{Y}_i$;
10:     $j \leftarrow j + 1$;
11: until $\mathcal{X}_j = \emptyset$ or $\mathcal{X}_{j-1} = \mathcal{X}_j$
```

To better understand the rational of Algorithm 5.1, consider the first iteration $j = 0$, where $\mathcal{X}_0$ corresponds to $\mathcal{Z}^\star$, i.e. the maximal $(A+BF)$-invariant contained in $\mathrm{Ker}\,(C)$, from which initial condition $x_0 = \tilde{x}_0 \in \mathcal{Z}^\star$ are taken to generate attacks. In line 6, for each measurement candidate C_i that may be added to the system, the subspace $\mathcal{Y}_i = \mathcal{X}_0 \cap \mathrm{Ker}\,(C_i)$ is computed. Denoting

$$\tilde{C}_i = \begin{bmatrix} C \\ C_i \end{bmatrix}$$

as the candidate measurement set composed of C and the candidate C_i, note that we have $\mathrm{Ker}\left(\tilde{C}_i\right) = \mathrm{Ker}\,(C) \cap \mathrm{Ker}\,(C_i)$. Moreover, the subspace $\mathcal{X}_0 = \mathcal{Z}^\star$ is contained in $\mathrm{Ker}\,(C)$, which yields $\mathcal{X}_0 \cap \mathrm{Ker}\,(C) = \mathcal{X}_0$. Thus, using the two latter relations, we conclude that the subspace $\mathcal{Y}_i$ actually corresponds to $\mathcal{X}_0 \cap \mathrm{Ker}\left(\tilde{C}_i\right)$. We conclude that, at the first iteration of line 8, Algorithm 5.1 chooses the measurement candidate that most reduces the dimension of $\mathcal{Z}^\star \cap \mathrm{Ker}\left(\tilde{C}_i\right)$, for all candidates $\tilde{C}_i$. Once the measurement candidate is chosen, $\mathcal{X}_1$ is computed in line 9 as the maximal $(A + BF)$-invariant contained in $\mathcal{X}_0 \cap \mathrm{Ker}\left(\tilde{C}_i\right)$ and the next iteration begins. As stated in Theorem 5.4.5, all attacks are revealed if $\mathcal{X}_1$ is empty. Moreover, note that $\mathcal{X}_1$ is contained in $\mathcal{Y}_i$, which means that the dimension of $\mathcal{X}_1$ is reduced at least as much as the dimension of $\mathcal{Y}_i$, when selecting C_i.

The previous discussion illustrates how Algorithm 5.1 is a greedy algorithm that, at each iteration j, chooses the measurement candidate C_i that most reduces the dimension of $\mathcal{Y}_i = \mathcal{X}_j \cap \mathrm{Ker}\,(C_i)$, where $\mathcal{X}_j$ is constructed at the previous iteration $j - 1$ as the subspace generating stealthy zero-dynamics attacks with respect to the perturbed system. Moreover, the dimension of $\mathcal{X}_{j+1}$, the maximal $(A + BF)$-invariant contained in $\mathcal{Y}_i$, is reduced at least as much as the dimension of $\mathcal{Y}_i$, when selecting C_i. This property agrees with the interpretation of Theorem 5.4.5, which states that all attacks are revealed if $\mathcal{X}_{j+1}$ is empty. Furthermore, the proposed algorithm requires the addition of at most $\dim(\mathcal{Z}^\star)$ new measurements.

5.4.2 Modifying the System Matrix A

Perturbations to the system dynamics as $\tilde{A} = A + \Delta A$ are now considered, resulting in the system $\tilde{\mathcal{P}} = (\tilde{A}, B, C)$. Recall that such an approach includes the case when a state-feedback law K is used to design the perturbed matrix $\tilde{A} = A + BK$. The following result provides conditions under which an attack remains stealthy for such perturbations.

Theorem 5.4.6. *There exists a vector $\tilde{x}_0 \in \mathcal{Z}^\star$ generating a stealthy attack to $\tilde{\mathcal{P}} = (\tilde{A}, B, C)$ if and only if there exists a non-empty $(A + BF)$-invariant subspace $\mathcal{X}$ that is contained in $\mathcal{Z}^\star \cap \operatorname{Ker}(\Delta A)$.*

Proof. Let $\tilde{x}_0 \in \mathcal{Z}^\star$ and recall that $w_0 = [\tilde{x}_0^\top \, \tilde{x}_0^\top]^\top$ belongs to the unobservable subspace of the augmented system (5.4). From Corollary 5.4.2, the attack remains stealthy if and only if w_0 is also in the unobservable subspace of the perturbed system (5.8). Using the PBH observability test (Zhou *et al.*, 1996, Theorem 3.4), this means that there exist a set of complex numbers $\{\lambda_i\}$ and vectors $\{w_i\}$ such that

$$\begin{bmatrix} \lambda_i I - \tilde{A} & -BF \\ 0 & \lambda_i I - (A + BF) \\ C & 0 \end{bmatrix} w_i = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \tag{5.9}$$

and w_0 belongs to the subspace spanned by the vectors $\{w_i\}$. Without loss of generality, suppose the unobservable subspace is one-dimensional. Next, we show that the vector $w_0 = [\tilde{x}_0^\top \, \tilde{x}_0^\top]^\top$ satisfies (5.9) for some complex number λ if and only if $\tilde{x}_0 \in \mathcal{Z}^\star$ is an eigenvector of $A + BF$ satisfying $\Delta A \tilde{x}_0 = 0$.

Recall from Corollary 5.2.4 that $C\tilde{x}_0 = 0$ holds for any $\tilde{x}_0 \in \mathcal{Z}^\star$, which addresses the third equation. Moreover, recall that the matrix F is designed so that the second equation is satisfied if and only if λ is an invariant-zero of the system belonging to $\sigma\left((A + BF)|_{\mathcal{Z}^\star}\right)$ and $\tilde{x}_0 \in \mathcal{Z}^\star$ is the corresponding eigenvector of $A + BF$.

For $\lambda \in \sigma\left((A + BF)|_{\mathcal{Z}^\star}\right)$ and $\tilde{x}_0 \in \mathcal{Z}^\star$ being the associated eigenvector, the first equation can be rewritten as $0 = \left(\lambda I - (\tilde{A} + BF)\right)\tilde{x}_0 = \Delta A \tilde{x}_0$. Therefore, we conclude that $w_0 = [\tilde{x}_0^\top \, \tilde{x}_0^\top]^\top$ belongs to the unobservable subspace of (5.8) if and only if $\tilde{x}_0 \in \mathcal{Z}^\star$ is an eigenvector of $A + BF$ and $\Delta A \tilde{x}_0 = 0$.

Denote $\mathcal{X}$ as the subspace spanned by all vectors $\tilde{x}_0 \in \mathcal{Z}^\star$ that are eigenvectors of $A+BF$ satisfying $\Delta A \tilde{x}_0 = 0$. Thus, $\mathcal{X}$ is contained in $\mathcal{Z}^\star$ and in the eigenspace of $A+BF$. Consequently, $\mathcal{X}$ is $(A+BF)$-invariant. Moreover, requiring that $\Delta A \tilde{x}_0 = 0$ holds for all $\tilde{x}_0 \in \mathcal{X} \subseteq \mathcal{Z}^\star$ is equivalent to have $\mathcal{X} \subseteq \mathcal{Z}^\star \cap \operatorname{Ker}(\Delta A)$, which concludes the proof. □

Recall from Theorem 5.2.3 that the zero-dynamics attacks generate state trajectories that belong to $\mathcal{Z}^\star$. The above result states that perturbations ΔA that are not excited by the state-trajectories of the system under attack cannot reveal the corresponding zero-dynamics attacks. The next result follows directly from Theorem 5.4.6 and characterizes perturbations ΔA that do not reveal any attack.

Corollary 5.4.7. *All the zero-dynamics attacks associated with a given $\tilde{x}_0 \in \mathcal{Z}^\star$ remain stealthy with respect to $\tilde{\mathcal{P}} = (\tilde{A}, B, C)$ if and only if $\mathcal{Z}^\star \subseteq \operatorname{Ker}(\Delta A)$.*

The previous results indicate that ΔA should be designed so that $\mathcal{X} \nsubseteq \operatorname{Ker}(\Delta A)$ for all $(A+BF)$-invariant subspaces $\mathcal{X} \subseteq \mathcal{Z}^\star$, thus revealing all the zero-dynamics attacks. Based on this reasoning, Algorithm 5.2 can be used to incrementally perform perturbations to the system matrix A that reveal zero-dynamics attacks.

Algorithm 5.2 Algorithm to deploy system matrix perturbations revealing zero-dynamics attacks.

```
Initialize $\mathcal{M} \leftarrow \{\Delta A_i\}$ as the set of possible system changes;
$j \leftarrow 0$;
$\Delta A^0 \leftarrow 0$;
$\mathcal{X}_0 \leftarrow \mathcal{Z}^\star$;
repeat
  for all $\Delta A_i \in \mathcal{M}$ do
    $\mathcal{Y}_i \leftarrow \mathcal{X}_j \cap \operatorname{Ker}\left(\Delta A^j + \Delta A_i\right)$;
  end for
  Choose $\Delta A_i \in \mathcal{M}$ such that $\dim(\mathcal{Y}_i)$ is minimized;
  $\Delta A^{j+1} \leftarrow \Delta A^j + \Delta A_i$;
  Compute $\mathcal{X}_{j+1}$ as the maximal $(A+BF)$-invariant contained in $\mathcal{X}_j \cap \operatorname{Ker}\left(\Delta A^{j+1}\right)$;
  $j \leftarrow j+1$;
until $\mathcal{X}_j = \emptyset$ or $\mathcal{X}_{j-1} = \mathcal{X}_j$
```

The interpretation of the latter algorithm is quite similar to that of Algorithm 5.1. In fact, Algorithm 5.2 is a greedy algorithm that, at each iteration j, selects the perturbation $\Delta A_i \in \mathcal{M}$ that most reduces the dimension of $\mathcal{X}_j \cap \operatorname{Ker}\left(\Delta A^j + \Delta A_i\right)$, where $\mathcal{X}_j$ is constructed at the iteration $j-1$ as the subspace generating stealthy zero-dynamics attacks with respect to $(A + \Delta A^j, B, C)$.

Note that the proposed algorithm converges in at most $\dim(\mathcal{Z}^\star)$ steps. Furthermore, all the zero-dynamics attacks become revealed if and only if the subspace $\mathcal{X}_j$ is empty.

5.4.3 Modifying the Input Matrix B

Here, we consider modifications on the input matrix B to reveal zero-dynamics attacks. A new input matrix $\tilde{B}$ is obtained by adding and removing actuators, or perturbing the matrix B by adding ΔB.

First, we consider the addition of secure actuators that may be used in state- or output-feedback controllers. The results in Section 5.4.2 can be applied to the case of state-feedback by having $\Delta A = \tilde{B}K$. As for the output-feedback case, the following result directly follows from the definition of 0-stealthy attacks.

Lemma 5.4.8. *Suppose secure actuators are added to $\mathcal{P}$, i.e. $\tilde{B} = [B\, B_i]$, and let the system $\tilde{\mathcal{P}} = (A, \tilde{B}, C)$ be controlled by an output-feedback controller $\mathcal{F} = (A_c, B_c, C_c, D_c)$. Then, all the zero-dynamics attacks on $\mathcal{P}$ remain stealthy with respect to $\tilde{\mathcal{P}}$.*

Proof. Let the secure input signals u^i_k be computed by $\mathcal{F}_i = (A_c, B_c, C^i_c, D^i_c)$ and considering the closed-loop system under attack

$$\begin{bmatrix} x_{k+1} \\ z_{k+1} \\ \tilde{x}_{k+1} \end{bmatrix} = \begin{bmatrix} A + B_i D^i_c C & B_i C^i_c & BF \\ B_c C & A_c & 0 \\ 0 & 0 & A + BF \end{bmatrix} \begin{bmatrix} x_k \\ z_k \\ \tilde{x}_k \end{bmatrix}$$

$$\tilde{y}_k = \begin{bmatrix} C & 0 & 0 \end{bmatrix} \begin{bmatrix} x_k \\ z_k \\ \tilde{x}_k \end{bmatrix}$$

with $x_0 = \tilde{x}_0 \in \mathcal{Z}^\star \subseteq \operatorname{Ker}(C)$ and $z_0 = 0$. Since $Cx_0 = 0$, by induction we have $z_k = z_0 = 0$ and $x_k = \tilde{x}_k \in \mathcal{Z}^\star$. Thus, we conclude that $\tilde{y}_k = 0$ for all $k \geq 0$, which completes the proof. □

The former statement shows that only adding inputs does not reveal any attack when output-feedback controllers are used. On the other hand, revealing zero-dynamics attacks by removing actuators also reduces the controllability of the system. A less intrusive approach is to change the actuator gains, i.e., have $\tilde{B} = BW$ and $\tilde{u}_k = W^{-1}u_k$ where W is an invertible matrix unknown to the attacker. This can be interpreted as a coding or encryption scheme performed by the actuator and controller with W as their shared private key. Assuming W is unknown by the attacker, we then have the following result.

Theorem 5.4.9. *There exists a vector $\tilde{x}_0 \in \mathcal{Z}^\star$ generating a stealthy attack to $\tilde{\mathcal{P}} = (A, BW, C)$ if and only if there exists a non-empty $(A+BF)$-invariant subspace $\mathcal{X}$ that is contained in $\mathcal{Z}^\star \cap \operatorname{Ker}\big(B(W-I)F\big)$.*

Proof. Without loss of generality, let $\tilde{x}_0 \in \mathcal{Z}^\star$ be an eigenvector of $A+BF$ generating a zero-dynamics attack. Recall from the proof of Theorem 5.4.6 that the attack is stealthy with respect to the perturbed system if and only if $w_0 = [\tilde{x}_0^\top\, \tilde{x}_0^\top]^\top$ is in the unobservable subspace of the perturbed system (5.8). This in turn holds if and only if there exists a complex number λ such that

$$\begin{bmatrix} \lambda I - A & -BWF \\ 0 & \lambda I - (A+BF) \\ C & 0 \end{bmatrix} \begin{bmatrix} \tilde{x}_0 \\ \tilde{x}_0 \end{bmatrix} = 0.$$

Following similar arguments as in the proof of Theorem 5.4.6, we conclude that the second and third equations are satisfied for $\lambda \in \sigma\big((A+BF)|_{\mathcal{Z}^\star})\big)$, with $\tilde{x}_0 \in \mathcal{Z}^\star$ being an eigenvector of $A+BF$ associated with λ. For $\tilde{x}_0 \in \mathcal{Z}^\star$ and $\lambda \in \sigma\big((A+BF)|_{\mathcal{Z}^\star})\big)$, the first equation can be rewritten as

$$\big(\lambda I - (A+BF) - B(W-I)F\big)\,\tilde{x}_0 = -B(W-I)F\tilde{x}_0 = 0.$$

Hence, the attack is stealthy if and only if $B(W-I)F\tilde{x}_0 = 0$.

Denote $\mathcal{X} \subseteq \mathcal{Z}^\star$ as the subspace spanned by all vectors $\tilde{x}_0 \in \mathcal{Z}^\star$ that are eigenvectors of $A+BF$. Recalling that $\mathcal{X}$ is $(A+BF)$-invariant, the proof concludes by noting that having $B(W-I)F\tilde{x}_0 = 0$ for all $\tilde{x}_0 \in \mathcal{X} \subseteq \mathcal{Z}^\star$ is equivalent to have $\mathcal{X} \subseteq \mathcal{Z}^\star \cap \operatorname{Ker}\big(B(W-I)F\big)$. □

Theorem 5.4.9 has a similar interpretation as that of Theorem 5.4.6: perturbations $\Delta B = B(W-I)F$ that are not excited by the state-trajectories of the system under attack cannot reveal the corresponding zero-dynamics attacks. Consequently, the next result readily follows.

Corollary 5.4.10. *All the zero-dynamics attacks on $\mathcal{P}$ remain stealthy with respect to $\tilde{\mathcal{P}} = (A, BW, C)$ if and only if $\mathcal{Z}^\star \subseteq \operatorname{Ker}\big(B(W-I)F\big)$.*

A sufficient condition for zero-dynamics attacks to be revealed with such perturbations follows directly from the previous theorem.

Corollary 5.4.11. *All the zero-dynamics attacks are revealed if*

$$\mathcal{Z}^\star \cap \operatorname{Ker}\big(B(W-I)F\big) = \emptyset.$$

The condition in Corollary 5.4.11 and the assumption that the system is observable can be used to provide a method for choosing W. First, we derive the following result.

Lemma 5.4.12. *Assume that (A, C) is observable. For any matrix F such that $\mathcal{Z}^\star$ is $(A+BF)$-invariant, it holds that $\mathcal{Z}^\star \cap \operatorname{Ker}(BF) = \mathcal{Z}^\star \cap \operatorname{Ker}(F) = \emptyset$.*

Proof. Recall that $\mathcal{Z}^\star$ is $(A+BF)$-invariant and suppose that $\mathcal{Z}^\star \cap \operatorname{Ker}(BF) \neq \emptyset$ i.e., there exists $\tilde{x}_0 \in \mathcal{Z}^\star$ such that $BF\tilde{x}_0 = 0$. This then implies that $\tilde{x}_0$ is A-invariant and generates an unobservable state trajectory, which is a contradiction since the system is observable. The proof concludes by observing that $\operatorname{Ker}(BF) = \operatorname{Ker}(F)$, since B has full column-rank. □

Using the above lemma, we conclude that a matrix W revealing all stealthy attacks can be constructed as $W = I + \bar{W}$ where $\bar{W}$ is a non-singular matrix. In fact, since B having full column rank yields $\operatorname{Ker}\big(B(W-I)F\big) = \operatorname{Ker}\big(B\bar{W}F\big) = \operatorname{Ker}(F)$, such a choice of W results in $\mathcal{Z}^\star \cap \operatorname{Ker}\big(B(W-I)F\big) = \emptyset$ and satisfies the condition in Corollary 5.4.11 to reveal all zero-dynamics attacks.

In particular, a possible weight for revealing zero-dynamics attacks is $W = \alpha I$ with $\alpha \neq 1$ being a scalar design parameter. We now analyze the effects of such a perturbation on the output energy of the system. Introducing the variable $\hat{x}_k = \alpha^{-1}x_k$, the perturbed system (5.8) can be rewritten as

$$\begin{aligned}\begin{bmatrix}\hat{x}_{k+1}\\ \tilde{x}_{k+1}\end{bmatrix} &= \begin{bmatrix}A & BF\\ 0 & A+BF\end{bmatrix}\begin{bmatrix}\hat{x}_k\\ \tilde{x}_k\end{bmatrix}\\ \tilde{y}_k &= \begin{bmatrix}\alpha C & 0\end{bmatrix}\begin{bmatrix}\hat{x}_k\\ \tilde{x}_k\end{bmatrix}\end{aligned} \tag{5.10}$$

with $\hat{x}_0 = \alpha^{-1}\tilde{x}_0$ and $\tilde{x}_0 \in \mathcal{Z}^\star$. The output of (5.10) is characterized as follows.

Theorem 5.4.13. *Suppose the augmented system* (5.8) *is at the state $\tilde{x}_0 = x_0 = \mathcal{Z}^\star$ when the perturbation $W = \alpha I$ is performed. After the perturbation the output is described by*

$$\begin{aligned}e_{k+1} &= Ae_k\\ \tilde{y}_k &= \alpha Ce_k\end{aligned}$$

with $e_0 = (\alpha^{-1} - 1)\tilde{x}_0$.

Proof. The proof comes from introducing the variable $e_k = \hat{x}_k - \tilde{x}_k$ and rewriting (5.10) with respect to e_k and $\tilde{x}_k$. □

Note that the output energy after the perturbation is dependent only on the open-loop dynamics, the initial condition $\tilde{x}_0$, and the scaling α. The results in Section 5.3 can be directly applied to characterize the output and residue signal energy when $W = \alpha I$, as summarized in the following statements.

Corollary 5.4.14. *Suppose the augmented system* (5.8) *is at the state $\tilde{x}_0 = x_0 = \mathcal{Z}^\star$ when the perturbation $W = \alpha I$ is performed. After the perturbation, the output energy is finite if and only if $\tilde{x}_0$ does not excite unstable eigenvalues of A.*

For attacks satisfying the conditions of Corollary 5.4.14, the finite output energy may be computed as follows, by using the coordinate transform $e_k = Tv_k$ as in (5.6).

Corollary 5.4.15. *Suppose that the perturbation $W = \alpha I$ is performed when the augmented system under a zero-dynamics attack* (5.8) *is at the state $\tilde{x}_0 = x_0 = \mathcal{Z}^\star$, which does not excite unstable modes of A, i.e. $\tilde{x}_0$ satisfies the equation $[0_{n_v}\ I_{n_x - n_v}]T^{-1}\tilde{x}_0 = 0$. After the perturbation, the output energy is given by*

$\|y\|_2^2 = \tilde{x}_0^\top \bar{Q} \tilde{x}_0$ *where*

$$\bar{Q} = T^{-\top} \begin{bmatrix} I_{n_v} \\ 0_{n_x - n_v} \end{bmatrix} Q_s \begin{bmatrix} I_{n_v} & 0_{n_x - n_v} \end{bmatrix} T^{-1}$$

and Q_s is the solution to

$$\Lambda_s^\top Q_s \Lambda_s - Q_s + \alpha^2 T_s^\top C^\top C T_s = 0.$$

Similarly to the above result, the output energy can also be characterized for the stable closed-loop system as follows.

Corollary 5.4.16. *Consider the closed-loop system* (5.2)*, which is assumed to be stable. Suppose the closed-loop system under a zero-dynamics attack is at the state $\xi_0 = [x_0^\top \ 0 \ 0]^\top$ with $x_0 = \tilde{x}_0 = \mathcal{Z}^\star$ when the perturbation $W = \alpha I$ is performed. After the perturbation, the residue signal energy is given by $\|\mathbf{r}\|_2^2 = \varepsilon_0^\top \mathbf{Q} \varepsilon_0$, where $\varepsilon_0 = [-\tilde{x}_0^\top \ 0 \ 0]^\top$ and $\mathbf{Q} \succeq 0$ is the solution to*

$$\mathbf{A}^\top \mathbf{Q} \mathbf{A} - \mathbf{Q} + \alpha^2 \mathbf{C}^\top \mathbf{C} = 0.$$

Note that, for stable closed-loop systems, Corollary 5.4.16 establishes that the perturbation $W = \alpha I$ results in a finite residue energy, even when the zero-dynamics are unstable. Moreover, the output energy is parameterized by the constant α, which is a design parameter.

5.5 Numerical Examples

To better illustrate the results from the previous sections, we provide an example of a zero-dynamics attack on a process control system: the quadruple-tank process described in Section 2.5.2. In the simulations, we consider the linearized model (2.6), at a given operating point, which is sampled with a period of $T_s = 0.5s$. The resulting discrete-time system is given by (5.1) with

$$A = \begin{bmatrix} 0.975 & 0 & 0.042 & 0 \\ 0 & 0.977 & 0 & 0.044 \\ 0 & 0 & 0.958 & 0 \\ 0 & 0 & 0 & 0.956 \end{bmatrix}, \quad B = \begin{bmatrix} 0.0515 & 0.0016 \\ 0.0019 & 0.0447 \\ 0 & 0.0737 \\ 0.0850 & 0 \end{bmatrix},$$

$$C = \begin{bmatrix} 0.2 & 0 & 0 & 0 \\ 0 & 0.2 & 0 & 0 \end{bmatrix}.$$

The corresponding maximal $(A, \mathrm{Im}\,(B))$-controlled invariant subspace contained in $\mathrm{Ker}\,(C)$, $\mathcal{Z}^\star$, is spanned by the columns of $Z^\star$, which is shown below together

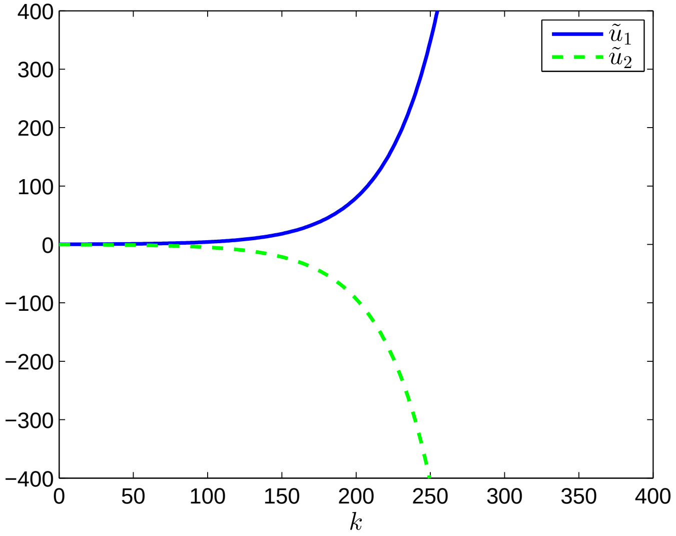


Figure 5.1: Unstable zero-dynamics attack applied to the system from $k = 0$, generated by $\tilde{x}_0 = \epsilon[0\;0\;-0.72\;0.69]^\top$.

with a suitable F:

$$Z^\star = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ -1 & 0 \\ 0 & 1 \end{bmatrix}, \quad F = \begin{bmatrix} 0 & 0 & -0.8057 & 0.0302 \\ 0 & 0 & 0.0349 & -0.9844 \end{bmatrix}.$$

The system $\mathcal{P} = (A, B, C)$ has two zeros, $\lambda = 0.89$ and $\lambda = 1.03$, and A has only stable eigenvalues. The unstable zero-dynamics corresponding to $\lambda = 1.03$ are excited by $\tilde{x}_0 = \epsilon[0\;\;0\;\;-0.72\;\;0.69]^\top$, where $\epsilon > 0$ is chosen so that the output energy is sufficiently small. The respective attack signal is depicted in Figure 5.1. This attack is considered in the examples below.

The interpretation of the attack signal with respect to the physical plant is as follows. Recall that the state of the system can be interpreted as the water-level deviations from equilibrium at each tank, i.e. $x_{(i),k} = h_{(i),k}$. Hence, the initial condition $\tilde{x}_0 = \epsilon[0\;\;0\;\;-0.72\;\;0.69]^\top$ indicates that only the water-levels of the 3rd and 4th water tanks deviate from equilibrium. Since $\tilde{x}_0$ is the eigenvector of $A + BF$ associated with the unstable eigenvalue $\lambda = 1.03$, the autonomous system $\tilde{x}_{k+1} = (A + BF)\tilde{x}_k$ has $\tilde{x}_k = \lambda^k \tilde{x}_0$ for all $k \geq 0$. Therefore, at any given time $k \geq 0$, only the water-levels of tanks 3 and 4 deviate from equilibrium. Recalling that only the water-levels of tanks 1 and 2 are measured, as seen in the output matrix C, we conclude that the attack results in a zero output signal.

With respect to the physical plant, the attack signal can be interpreted as a coordinated behavior of the water pumps that only affects the water-levels of tanks 3 and 4, while leaving the levels of tanks 1 and 2 unchanged. This interpretation helps to illustrate the reasoning behind different solutions to detect the attack. For instance, adding one additional sensor measuring either tank 3 or tank 4 would reveal the attacks, as described next.

5.5.1 Modifying the Output Matrix C

Suppose that the following additional measurements can be used to reveal zero-dynamics attacks:

$$\begin{aligned} C_3 &= \begin{bmatrix} 0 & 0 & 0.2 & 0 \end{bmatrix}, \\ C_4 &= \begin{bmatrix} 0 & 0 & 0 & 0.2 \end{bmatrix}, \end{aligned}$$

where C_3 and C_4 measure the water-level of tanks 3 and 4, respectively. Consider Algorithm 5.1 proposed in Section 5.4.1. The first iteration is initialized with $\mathcal{X}_0 = \mathcal{Z}^\star$ and we see that adding C_3 yields $\mathcal{Y}_3 = \mathcal{X}_0 \cap \operatorname{Ker}(C_3) = \operatorname{span}([0\ 0\ 0\ 1]^\top)$. The next step is to compute $\mathcal{X}_1$, i.e. the maximal $(A+BF)$-invariant subspace contained in $\mathcal{Y}_3$. Since $\mathcal{Y}_3$ is one-dimensional, $\mathcal{X}_1$ must be either empty or equal to $\mathcal{Y}_3$. Note that $\mathcal{Y}_3$ is not $(A+BF)$-invariant, since the vector $[0\ 0\ 0\ 1]^\top \in \mathcal{Y}_3$ is not an eigenvector of $(A+BF)$. Therefore, we conclude that $\mathcal{X}_1$ is empty and, thus, all the zero-dynamics attacks to $\mathcal{P}$ are revealed. In fact $\tilde{\mathcal{P}} = (A, B, \tilde{C})$ with $\tilde{C} = [C^\top\ C_3^\top]^\top$ has no zeros. In this particular example, adding C_4 instead of C_3 would also reveal all the zero-dynamics attacks. Note that, while Algorithm 5.1 is ensured to converge in at most $\dim(\mathcal{Z}^\star) = 2$ iterations, a single iteration was enough to reveal all zero-dynamics attacks.

5.5.2 Modifying the System Matrix A

From Corollary 5.4.7, we have that any system perturbation of the type

$$\Delta A = \begin{bmatrix} \Delta & 0 \end{bmatrix}$$

with $\Delta \in \mathbb{R}^{4\times 2}$ leaves all the zero-dynamics attacks stealthy, since $\Delta A Z^\star = 0$. In terms of the physical plant, such perturbations are only driven by the states of tanks 1 and 2, i.e., $\Delta A x_k = \Delta[x_{(1),k}\ x_{(2),k}]^\top$. Recalling that the attack does not affect the water-levels of tanks 1 and 2, we conclude that the attack would remain undetected. In fact, note that $(A+\Delta A+BF)\mathcal{Z}^\star = (A+BF)\mathcal{Z}^\star$, which says that the zero-dynamics of $\mathcal{P}$ and $\tilde{\mathcal{P}}$ are identical. Therefore, such perturbations should be avoided. On the other hand, the zero-dynamics change for perturbations of the type

$$\Delta A = \begin{bmatrix} 0 & \Delta \end{bmatrix}.$$

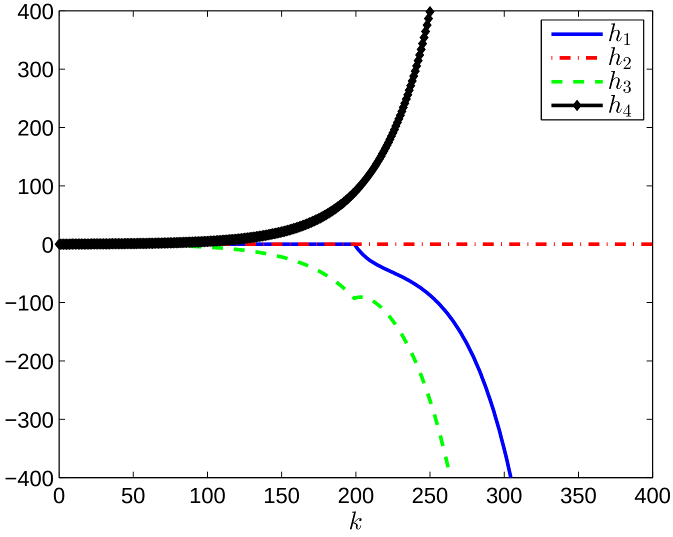


Figure 5.2: State trajectories of the system under attack with attack detection. The zero-dynamics attack starts at $k = 0$ with an initial condition mismatch. The states of tanks 1 and 2 remain close to zero until the system matrix A is perturbed at $k = 200$. After the perturbation, the state of tank 1 significantly changes, which reveals the attack.

For instance, adding an extra connection from tank 3 to tank 1 corresponds to

$$\Delta A = \begin{bmatrix} 0 & 0 & 0.0397 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & -0.0402 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}.$$

The outcome of such a perturbation can be seen in Figure 5.2 and Figure 5.3. The attack begins at $k = 0$ with an initial conditions mismatch, leading to a small increase in the output energy as initially seen in Figure 5.3. The change to the system dynamics occurs at $k = 200$ and one immediately observes a perturbation in the state trajectory. The extra coupling between tanks 3 and 1 changes the zero-dynamics of the system and thus the current attack signal affects the water level of tank 1. As a result the attack is revealed in the output, as illustrated in Figure 5.3.

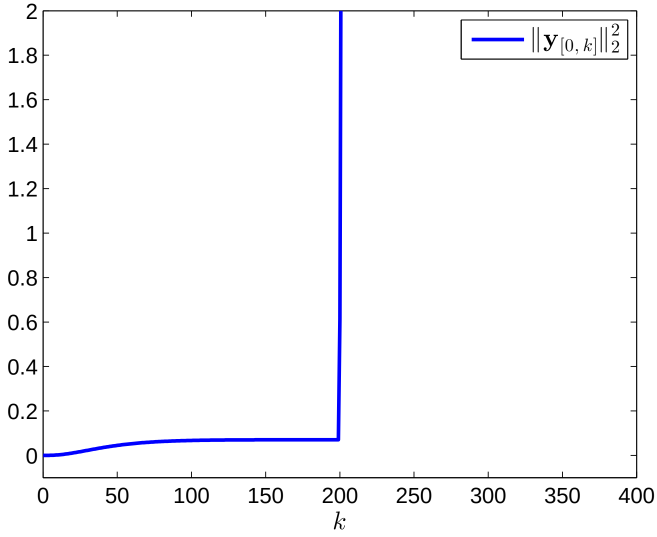


Figure 5.3: Output energy of the system under attack with attack detection. The zero-dynamics attack starts at $k = 0$ with an initial condition mismatch. The mismatch results in a small increase in the output energy. The system matrix A is perturbed at $k = 200$, by connecting tank 3 to tank 1, which results in a steep increase of the output energy and reveals the attack.

5.5.3 Modifying the Input Matrix B

Consider the case where the uniform input scaling $W = 0.987I$ is applied to the system. From the results in Section 5.4.3, all the zero-dynamics are revealed, since $\mathrm{Ker}\,(BF) = \mathrm{Ker}\,\big((1-\alpha)BF\big)$ and $\mathcal{Z}^\star \cap \mathrm{Ker}\,(BF) = \emptyset$. Moreover, as stated in Corollary 5.4.14, the scaling results in a finite energy output since A is stable. The output energy resulting from the attack an input scaling is depicted in Figure 5.4. As before, the attack begins at $k = 0$ with a mismatch in the initial condition, resulting in a finite output energy. The input scaling is applied at $k = 200$, which again results in a finite increment of the output energy since A is stable, as depicted in Figure 5.4.

5.6 Summary

The problem of revealing zero-dynamics attacks on control system was tackled. First, we studied the effect of initial condition mismatch in terms of the resulting increase in the output energy. We concluded that for the subset of attacks exciting unstable zero-dynamics, this effect can be made arbitrarily small while still

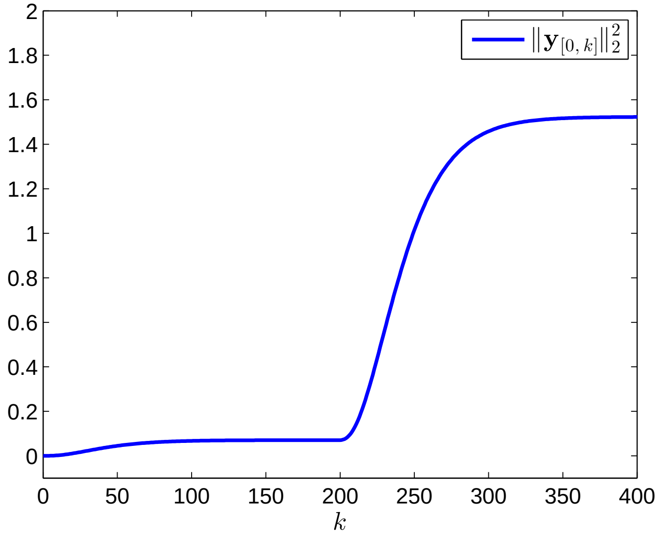


Figure 5.4: Output energy of the system under attack with attack detection. The zero-dynamics attack starts at $k = 0$ with an initial condition mismatch. The mismatch results in a small increase in the output energy. The output energy significantly increases after introducing the input scaling $BW = 0.987B$ at $k = 200$, which reveals the attack.

affecting the system performance. Then, we addressed the problem of revealing zero-dynamics attacks by modifying the system structure in terms of the respective outputs, inputs, and dynamics. For changes in each component, we provided necessary and sufficient conditions for all attacks to be revealed. Furthermore, we provided an algorithm to incrementally add measurements and thus reveal attacks. We also proposed a coordinated scaling of the inputs by the actuator and controller. For this particular change, we quantified the resulting increase in output energy in terms of the initial condition and scaling factor. Both these changes on the inputs and outputs are able to reveal attacks while not affecting the system performance when no attack is present.

Chapter 6

Distributed Fault Detection and Isolation in Networked Systems

Increasing the cyber security by adding encryption and authentication schemes helps to prevent some attacks by making them harder to succeed. However, it would be a mistake to rely solely on such methods, as it is well-known that the overall system is not secured simply because some of its components are. One way to enhance resiliency of networked control systems is to design control algorithms that are robust to the effects of certain categories of faults and attacks (Lynch, 1997; Amin *et al.*, 2009; Hou *et al.*, 2009; Sundaram and Hadjicostis, 2011). Another way is to develop monitoring schemes to detect anomalies in the system caused by attacks and faults (Pasqualetti *et al.*, 2007) and mitigate these threats upon detection. The latter approach in general allows faster and more effective responses to anomalies as opposed to the former, since properties of the fault such as location and fault signal can be obtained. Moreover, monitoring schemes can also improve the state-awareness of the system (Rieger, 2010).

Automatic detection of system faults is of growing importance as the size and complexity of systems rapidly increase. Most of the available literature on model-based fault detection and isolation (FDI) focuses on centralized systems where the FDI scheme has access to all the available measurements and the objective is to detect and isolate faults occurring in any part of the system (Isermann, 2004; Ding, 2008; Chen and Patton, 1999). Distributed implementations are more suitable than centralized for large-scale interconnected dynamical systems such as power networks and multi-agent systems due to its lower complexity and less use of network resources (Siljak, 1991). Traditional FDI schemes may not be applied to distributed systems, since not all measurements are available in every node. However, in large-scale networked systems such as electric power systems, even benign disturbances such as model changes or unmeasured signals may hinder the detection of faults. Additionally, a global model of the system may not be available, or the large size of the system may lead to computationally intractable monitoring schemes. Hence in order to meet the demands of resilient control system components, monitoring schemes need to be architectured and designed to provide scalable solutions suitable for large-scale highly uncertain networked systems.

In this chapter, we address the design of distributed FDI for large-scale networked systems that resilient to model changes and external faults, not requiring the exact global model of the network to be known to the nodes.

6.1 Contributions and Related Work

Power networks are large-scale spatially distributed dynamical systems. Being a critical infrastructure, they possess strict safety and reliability constraints (Shahidehpour *et al.*, 2005). Monitoring the state of the system is essential to guarantee safety. Currently this is typically done in a centralized control center through a single state estimator. The core methodology for state estimation of power systems dates from 1970 (see Schweppe and Wildes, 1970; Abur and Exposito, 2004). Due to the low sampling frequency of the sensors in these systems a steady-state approach is taken, which only allow for an over-constrained operation of the system to ensure reliability. Furthermore dynamic faults such as generator electro-mechanical oscillations may pass undetected by schemes based on steady-state models and measurements, possibly leading to cascade failures.

In recent years, measurement units with higher sampling rate have been developed, e.g., Phasor Measurement Units (PMU), opening the way to dynamic state estimators and observer-based fault detection schemes taking in account the dynamics of the system. As discussed in Section 2.2.1, there are various ways to detect and isolate a fault in a dynamical system (Massoumnia and Verghese, 1989; Chen and Patton, 1999; Isermann, 2004; Ding, 2008). A recent survey of different techniques can be found in Hwang *et al.* (2010). One approach is to use the system model to design a set of parity equations. In the case of dynamical systems, such parity equations can be obtained by exploiting the temporal correlation among state, input, and output variables for a given time-horizon. This approach was used in Han *et al.* (2005) to design a centralized FDI scheme insensitive to certain model changes and disturbances. Our approach is similar, but relies on an observer-based approach and results in a distributed FDI scheme.

Centralized observer-based FDI approaches have been well studied and some of these methods have been proposed for power systems (Scholtz and Lesieutre, 2008; Aldeen and Crusca, 2006; Demetriou, 2005). However, distributed FDI for systems comprised of a network of autonomous nodes is still in its infancy. Recently, a distributed FDI scheme for a network of interconnected first-order systems was proposed by Pasqualetti *et al.* (2012). The authors analyzed limitations on fault detectability and isolability in a system theoretic perspective. Distributed schemes for power networks were also developed. Pasqualetti *et al.* (2013) studied centralized and distributed fault detection schemes for networked systems modeled by differential-algebraic equations. Using swing-equation models of power networks, Nishino and Ishii (2014) proposed distributed fault detection schemes for power

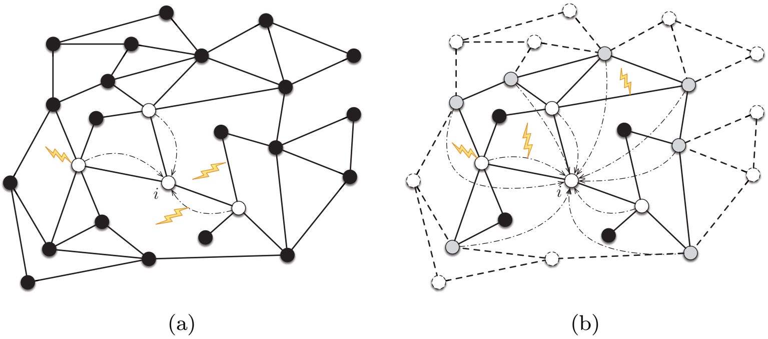


Figure 6.1: The networked system with faults, where nodes correspond to dynamical subsystems and undirected edges represent coupled dynamics between nodes. In distributed FDI schemes, node i aims at detecting and isolating faults on the solid white nodes and edges incident to them. Scenario (a) depicts the case where node i has access to measurements from its neighbors, represented by directed edges, and knows the entire network model. In scenario (b), node i only knows a local model of the network, where the dashed nodes and edges are unknown to node i. Moreover, node i receives measurements from the solid white and gray nodes.

networks using power flow and PMU measurements. In Ding *et al.* (2008), a bank of decentralized observers is built where each observer contains the model of the entire system and receives both measurements from the local subsystem and information transmitted from other observers. In both contributions, the exact model of the system is assumed to be known. Distributed FDI schemes using uncertain models were proposed in Ferrari *et al.* (2009). However, these schemes require bounded interconnections between the subsystems and knowledge of these bounds. A similar approach was followed by Zhang and Zhang (2012) and applied to nonlinear power system models, but in addition to bounded model uncertainty they required also communication between neighboring FDI filters.

This chapter tackles the problem of distributed FDI for large-scale interconnected systems with respect to different fault models. The networked system with different fault types are illustrated in Figure 6.1. The networked system is composed of interconnected individual subsystems, represented by nodes. Each node has access to local measurements from nodes in its vicinity, represented by directed edges. As an example, the measurements available to node i are depicted in Figure 6.1. The interconnections between subsystems are represented by undirected edges between nodes and model either physical couplings, as in the case of power networks, or distributed control laws computed based on the local measurements, which are present, for instance, in mobile multi-agent systems. Faults may affect the network through the nodes, undirected edges, and directed edges. Given the system model and local measurements, distributed FDI aims at having each node of the network detecting and isolating faults in its vicinity, as illustrated in Figure 6.1.

First we tackle the problem of distributed FDI with respect to faulty nodes and faulty edges. In particular, we consider schemes based on Unknown Input Observers (UIO) and, given the local measurements and system model as depicted in Figure 6.1a, we derive results on the existence of UIOs at each node for the different fault models.

As our second contribution, we consider the case where the UIOs are designed based on uncertain network models. More precisely, the model uncertainty is caused by the removal of edges or nodes with respect to the nominal model. The proposed distributed FDI scheme is shown to be somewhat resilient to network changes that are external to a node's local subsystem, i.e. that occur on the dashed nodes or edges in Figure 6.1b. Additionally, we propose a novel distributed FDI scheme based on local models and an augmented set of measurements from the local subsystem, as illustrated in Figure 6.1b. As opposed to approaches similar to Ferrari *et al.* (2009); Zhang and Zhang (2012), bounding the subsystems' interactions is not required. Instead, by using the additional measurements, the local FDI filter can be decoupled from faults and model changes in the external subsystems and it can detect and isolate faults in the neighboring nodes.

Our third contribution is to address the complexity reduction of the distributed FDI scheme. More precisely, leveraging on our second contribution, we outline the minimum amount of model information and measurements that are sufficient for a node to achieve FDI using only its local measurements and models. In particular, our results show that using the local model from a node's 2-hop neighborhood and the corresponding measurements may not be optimal. The proposed scheme has reduced computational complexity and required model knowledge compared to the schemes such as Pasqualetti *et al.* (2012); Sundaram and Hadjicostis (2011), which use the global system's model. Moreover, we propose a method to reduce the number of monitoring nodes while ensuring that all nodes are being monitored. Importantly, we do not assume that the monitoring nodes exchange information with each other.

The outline is as follows. In Section 6.2, we describe the system and fault models and define the problem of distributed FDI. Section 6.3 begins by recalling the existing FDI tools, which are then used to design distributed solutions to detect and isolate faulty nodes and edges, respectively. In Section 6.4, we show how to distributedly detect faults when the network model is uncertain using two different methods. The first method adapts the detection thresholds of the original distributed FDI, while the second consists of a novel distributed FDI method based on local models that not only requires less computations than the one presented in Section 6.3, but also is capable of handling uncertain network models. In Section 6.5 we propose methods to further reduce the computational burden of the methods proposed FDI schemes. Some numerical examples are given in Section 6.6. A summary of the chapter is given in Section 6.7.

6.2 Problem Formulation

Consider a network of N interconnected dynamical systems and let $\mathcal{G}(\mathcal{V},\mathcal{E})$ be the underlying graph of this network, where $\mathcal{V} \triangleq \{i\}_{i=1}^{N}$ is the vertex set and $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ is the edge set of the graph. Denote $\mathcal{A} \in \mathbb{R}^{N\times N}$ as the weighted adjacency matrix with nonnegative entries. The undirected edge $\{i,j\}$ is incident to vertices i and j if nodes i and j share a communication link, in which case the corresponding entry in the adjacency matrix $[\mathcal{A}]_{ij}$ is positive. The degree of node i is $\deg(i) \triangleq \mathcal{A}\mathbf{1}_N = \sum_{j\in\mathcal{N}_i} [\mathcal{A}]_{ij}$, where the entries of $\mathbf{1}_N \in \mathbb{R}^N$ are equal to 1, $\mathcal{N}_i \triangleq \big\{j \in \mathcal{V} : \{i,j\} \in \mathcal{E}\big\}$ is the neighborhood set of i with $N_i \triangleq |\mathcal{N}_i|$, and the degree matrix of $\mathcal{G}$ is $\Delta \triangleq \mathrm{diag}(\deg(1),\ldots,\deg(N))$. The Laplacian of $\mathcal{G}$ is defined as $\mathcal{L}(\mathcal{G}) \triangleq \Delta - \mathcal{A}$. Consider a subset of the vertex set $\tilde{\mathcal{V}} \subseteq \mathcal{V}$ and a subset of the edge set $\tilde{\mathcal{E}} \subseteq \mathcal{E}$. The subgraph of $\mathcal{G}$ induced by $\tilde{\mathcal{V}}$ and $\tilde{\mathcal{E}}$ is denoted as $\tilde{\mathcal{G}} \triangleq \mathcal{G}(\tilde{\mathcal{V}},\tilde{\mathcal{E}})$. Moreover, assume that the state of each node is given by $x_i(t) \in \mathbb{R}^2$.

We call the set $\mathcal{N}_i^{\ell} \subset \mathcal{V}$ *the ℓ-hop neighbor set of node i* where $v \in \mathcal{N}_i^{\ell}$ if there is a path of length at most ℓ between i and v. Defining $\mathcal{V}_i^{\ell} \triangleq \{i\} \cup \mathcal{N}_i^{\ell}$, we call the subgraph $\mathcal{G}_i^{\ell}\left(\mathcal{V}_i^{\ell},\mathcal{E}_i^{\ell}\right) \subseteq \mathcal{G}(\mathcal{V},\mathcal{E})$ *the ℓ-hop neighborhood graph of node i* where $\{v,u\} \in \mathcal{E}_i^{\ell}$ if $\{v,u\} \in \mathcal{E}$ and $u,v \in \mathcal{N}_i^{\ell}$. For the case where $\ell = 1$, we drop the superscript for the ease of notation. We call the graph $\mathcal{P}_i(\mathcal{V}_{P_i},\mathcal{E}_{P_i}) \subseteq \mathcal{G}(\mathcal{V},\mathcal{E})$, where $\mathcal{V}_{P_i} = \{i\} \cup \mathcal{N}_i \cup \overline{\mathcal{N}}_i$, and $\mathcal{E}_{P_i} = \mathcal{E}_i \cup \overline{\mathcal{E}}_i$, *the proximity graph of node i* where $\{v,u\} \in \mathcal{E}_i$ if $\{v,u\} \in \mathcal{E}$ and $u,v \in \mathcal{N}_i$. Moreover, $\overline{\mathcal{N}}_i$ is the set of all the nodes in the network that are not in $\mathcal{N}_i$ but share a link with at least one of the nodes in $\mathcal{N}_i$, and $\overline{\mathcal{E}}_i$ is the set of all edges incident to at least one of the nodes in $\mathcal{N}_i$ that are not in $\mathcal{E}_i$. Examples for the notation above are given in Figure 6.2.

Consider the linear time-invariant networked system described by

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bv(t) + Ef(t), \\ y_i(t) &= C_i x(t) + D_i f(t), \quad \forall\, i \in \mathcal{V}, \end{aligned} \tag{6.1}$$

where $x(t) \in \mathbb{R}^n$ is the global state vector containing all the agents' states, $v(t) \in \mathbb{R}^N$ is a known input vector, $y_i(t) \in \mathbb{R}^{m_i}$ is the set of measurements available at node i, and $f(t) \in \mathbb{R}^p$ is an unknown vector of faults affecting the system. We are interested in the problem of distributed fault detection and isolation, as described below.

Definition 6.2.1 (Distributed fault detection and isolation). *Consider the system* (6.1) *and suppose each node i has a model of the system and a local set of measurements $y_i(t)$ to design a FDI scheme. A fault $f(t) \not\equiv 0$ is said to be* detected *if at least one node $i \in \mathcal{V}$ decides that there exists an active fault in the network. Furthermore, a fault is said to be* isolated *if there exists a set of nodes that detect the fault and identify the faulty components, i.e. identify the non-zero elements of $f(t)$.*

The main aim of this work is to leverage the structural properties of the networked system (6.1) to characterize under what conditions the problem of dis-

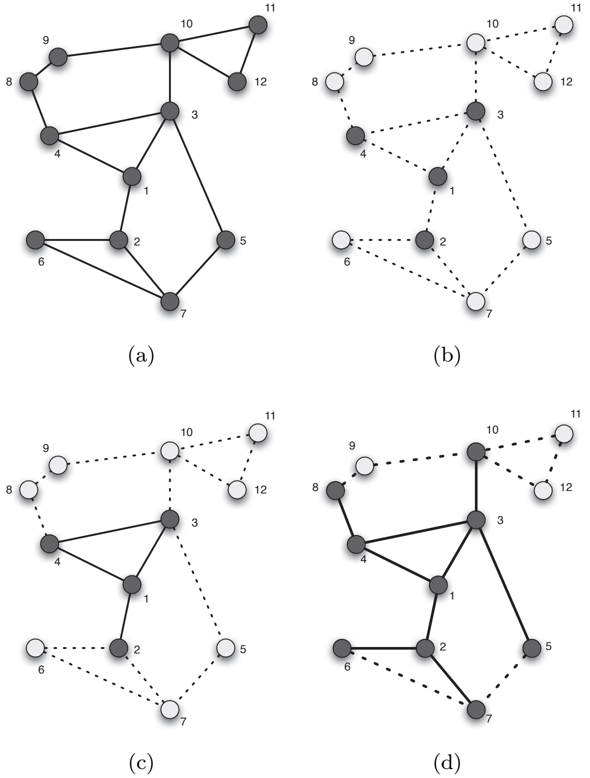


Figure 6.2: (a) A network with 12 nodes. (b) The set of one-hop neighbors of node 1, $\mathcal{N}_1$, are nodes $\{2, 3, 4\}$ and are coloured darker. (c) The one-hop neighborhood graph of node 1, $\mathcal{G}_1$, is the set of dark nodes connected by solid lines. (d) The graph represented by dark nodes that are connected to each other by solid lines is the proximity graph of node 1, i.e., $\mathcal{P}_1$.

tributed fault detection and isolation can be solved. In particular, we focus on the networked second-order systems, while similar results for networked first-order systems can be obtained (see, for instance, Pasqualetti *et al.*, 2012). For this case, the state of each node, $x_i(t) = [\xi_i(t)\ \zeta_i(t)]^\top$, $\xi_i(t)$, and $\zeta_i(t) \in \mathbb{R}$, is governed by

$$\begin{aligned}\dot{\xi}_i(t) &= \zeta_i(t)\\ \dot{\zeta}_i(t) &= u_i(t) + v_i(t) + f_i(t),\end{aligned}$$

where $\xi_i(t)$ and $\zeta_i(t)$ are the scalar states, $v_i(t)$ is the i-th entry of the external reference input $v(t)$, $u_i(t)$ is a scalar distributed control input capturing the interactions between neighboring nodes, and $f_i(t)$ is an unknown fault affecting node i. Additionally, each agent i has access to its own states and receives measurements of its neighbors' states, possibly corrupted by faults. Denoting

$$x(t) = [\xi_1(t)\ \dots\ \xi_N(t)\ \zeta_1(t)\ \dots\ \zeta_N(t)]^\top$$

as the global system state, the measurement vector with corrupted measurements is described as

$$\begin{aligned} y_i(t) &= C_i x(t) + C_i \sum_{j\in\mathcal{N}_i} \Big(l_j f_{ij}^{\xi}(t) + l_{N+j} f_{ij}^{\zeta}(t)\Big)\\ &= \begin{bmatrix} \xi_i(t) & \xi_{j_1}(t) & \dots & \xi_{j_{N_i}}(t) & \zeta_i(t) & \zeta_{j_1}(t) & \dots & \zeta_{j_{N_i}}(t) \end{bmatrix}^\top\\ &+ \begin{bmatrix} 0 & f_{ij_1}^{\xi}(t) & \dots & f_{ij_{N_i}}^{\xi}(t) & 0 & 0 & \dots & 0 \end{bmatrix}^\top\\ &+ \begin{bmatrix} 0 & 0 & \dots & 0 & 0 & f_{ij_1}^{\zeta}(t) & \dots & f_{ij_{N_i}}^{\zeta}(t) \end{bmatrix}^\top, \end{aligned} \tag{6.3}$$

where $j_k \in \mathcal{N}_i$ for all $k = 1, \dots, N_i$, $l_i \in \mathbb{R}^{2N}$ is the i-th column of I_{2N}, and $C_i = [\bar{C}_i^\top\ \bar{C}_i^\top]^\top$, with $\bar{C}_i \in \mathbb{R}^{|\mathcal{V}_i^1|\times N}$ being a full row rank matrix where each of the rows have all zero entries except for one entry at the j-th position that corresponds to those nodes that are in $\mathcal{V}_i^1 = \{i\} \cup \mathcal{N}_i$. The variables $f_{ij}^{\xi}(t)$ and $f_{ij}^{\zeta}(t)$ for $j \in \mathcal{N}_i$ denote measurement corruptions on ξ_j and ζ_j, respectively.

The distributed control input $u_i(t)$ is given by the linear control law on $y_i(t)$:

$$u_i(t) = \sum_{j\in N_i} (w_{ij} + f_{ij}^{w}(t)) \Big[(\xi_j(t) + f_{ij}^{\xi}(t) - \xi_i(t)) + \mu(\zeta_j(t) + f_{ij}^{\zeta}(t) - \zeta_i(t)) \Big] - \kappa_i \zeta_i(t), \tag{6.4}$$

where $w_{ij} = w_{ji} > 0$ are the edge weights, $\kappa_i, \mu \geq 0$ for $i, j = 1, \dots, N$, and $f_{ij}^{w}(t) = f_{ji}^{w}(t)$ is an unknown fault affecting the weight of the edge $\{i, j\}$.

The overall dynamics of the networked system under the control law (6.4) are described by (6.1) with

$$A = \begin{bmatrix} 0_N & I_N \\ -\mathcal{L} & -\mu\mathcal{L} - \overline{K} \end{bmatrix}, \quad B = \begin{bmatrix} 0_N \\ I_N \end{bmatrix}.$$

The matrix $\mathcal{L}$ is the weighted Laplacian matrix associated with the network where w_{ij} is the weight of edge $\{i, j\}$, and $\overline{K} = \mathrm{diag}(\kappa_1, \dots, \kappa_N)$.

The control law described by (6.4) with $f(t) \equiv 0$ is a generalized form of the two following well-known control laws:

$$\begin{aligned} u_i^1(t) &= -\kappa_i \zeta_i(t) + \sum_{j \in \mathcal{N}_i} w_{ij}(\xi_j(t) - \xi_i(t)), \\ u_i^2(t) &= \sum_{j \in \mathcal{N}_i} w_{ij} \left[(\xi_j(t) - \xi_i(t)) + \mu(\zeta_j(t) - \zeta_i(t)) \right]. \end{aligned}$$

Analysis of these control laws and design rules for κ_i, w_{ij}, and μ may be found in Ren and Atkins (2007); Qin *et al.* (2012).

Remark 6.2.1. *Under both these control laws with $f(t) \equiv 0$, for all $i, j \in \mathcal{V}$ we have $|\xi_i - \xi_j| \to 0$ and $|\zeta_i - \zeta_j| \to 0$ exponentially fast (Ren and Atkins, 2007; Qin et al., 2012). Furthermore, we denote the consensus equilibria as $\bar{x} = [\bar{\xi} \; \bar{\zeta}]^\top \otimes \mathbf{1}_N$ with $\bar{\xi} = \lim_{t \to +\infty} \xi_i(t)$ and $\bar{\zeta} = \lim_{t \to +\infty} \zeta_i(t)$, where $\otimes$ denotes the Kronecker product.*

The introduced networked system can represent many practical systems, which may lead to different fault models and interpretations. In this chapter, we consider two application examples, namely mobile multi-agent systems and electric power networks. For a mobile multi-agent system (Ren and Atkins, 2007), each node i represents a vehicle, where the variables ξ_i and ζ_i can be interpreted as the corresponding position and velocity, respectively. In this case, the node fault $f_i(t)$ corresponds to faults in the vehicle dynamics, which can represent, for instance, an obstacle immobilizing the vehicle. In mobile multi-agent systems, the edges map to communication or sensing links between the vehicles. For such systems, each node implements the control law by obtaining state measurements from the neighbors. Therefore, given the graph representation of the system, faults in the measurements appear as *sensing faults* on edges, corresponding to the signals $f_{ij}^{\xi}(t)$ and $f_{ij}^{\zeta}(t)$.

In the context of synchronous power systems (Kundur, 1994), each node i maps to a generator or motor, with ξ_i and ζ_i being the corresponding phase and frequency, respectively. In this case, node faults $f_i(t)$ may represent electro-mechanical disturbances affecting the electrical machines, e.g. a sudden change in the mechanical power supplied to the generators. For electric power networks represented by graphs, the edges model physical transmission lines between electrical machines. In this case, the control law corresponds to the model of the physical coupling between the nodes, thus being part of the physical system itself. Moreover, faults on the edges represent are actually faults on the transmission lines. In this work, we consider that such faults correspond to changes in the transmission line parameters. In particular, the edge weights $w_{ij} = w_{ji}$ may be affected by a fault and become $w_{ij} + f_{ij}^w(t) = w_{ji} + f_{ji}^w(t)$, which correspond to *parameter faults.*

Given the previous interpretations of the networked system (6.1) and corresponding node and edge faults, we define faulty nodes and faulty edges as follows.

Definition 6.2.2. *A node $i \in \mathcal{V}$ is faulty if $f_i(t) \not\equiv 0$. The system affected by the fault $f(t) = f_i(t)$ is modeled by (6.1) with $E = b_i$ and $D_i = 0$, where b_i is the i-th column of B.*

Definition 6.2.3. *An edge* $\{i,j\} \in \mathcal{E}$ *is faulty if any of the signals* $f_{ij}^{w}(t)$, $f_{ji}^{w}(t)$, $f_{ij}^{\xi}(t)$, $f_{ji}^{\xi}(t)$, $f_{ij}^{\zeta}(t)$, *and* $f_{ji}^{\zeta}(t)$ *are not identically zero. Moreover, we classify edge faults as either sensing faults or parameter faults.*

(i) A fault on edge $\{i,j\}$ *is a sensing fault from* j *to* i *if any of the signals* $f_{ij}^{\xi}(t)$ *and* $f_{ij}^{\zeta}(t)$ *are not identically zero and* $f_{ij}^{w}(t) \equiv 0$. *The system affected by the fault* $f(t) = [f_{ij}^{\xi}(t)\ f_{ij}^{\zeta}(t)]^\top$ *is modeled by* (6.1) *with* $E = b_i[w_{ij}\ \mu w_{ij}]$ *and* $D_i = C_i[l_j\ b_j]$, *where* l_j *is the* j*-th column of* I_{2N}.

(ii) A fault on edge $\{i,j\}$ *is a parameter fault if the signals* $f_{ij}^{\xi}(t)$, $f_{ij}^{\zeta}(t)$, $f_{ji}^{\xi}(t)$, *and* $f_{ji}^{\zeta}(t)$ *are identically zero and* $f_{ij}^{w}(t) = f_{ji}^{w}(t) \not\equiv 0$. *The system affected by the fault* $f(t) = \delta_{ij}(t) f_{ij}^{w}(t)$ *with* $\delta_{ij}(t) = \xi_j(t) - \xi_i(t) + \mu(\zeta_j(t) - \zeta_i(t))$ *is modeled by* (6.1) *with* $E = b_i - b_j$ *and* $D_i = 0$.

Given the control input (6.4) and local measurements from its neighbors (6.3), node i cannot compute each neighbor's input. Therefore, FDI based solely on individual models (6.2) is infeasible, as the neighbors trajectories cannot be estimated. However, the control inputs and corresponding trajectories can be estimated by using the global model of the networked system (6.1), as described next.

6.3 Distributed Fault Detection and Isolation

In this section we revisit some of the results on FDI using UIOs. The methodology is presented for the case of faulty nodes, but applies straightforwardly to the other scenarios.

Recall the problem of distributed FDI as per Definition 6.2.1, where each node i monitors its neighborhood to detect and isolate faulty components. For each node $i = 1, \dots, N$, consider a model of the form:

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bv(t) + \sum_{k\in\mathcal{N}_i} E_k f_k(t), \\ y_i(t) &= C_i x(t) + \sum_{k\in\mathcal{N}_i} D_{i,k} f_k(t), \end{aligned} \tag{6.7}$$

where, recalling Definition 6.2.2, a faulty node k is modeled by $E_k = b_k$ and $D_{i,k} = 0$. For the ease of notation, we assume that there is at most one faulty node. Note that this assumption is not essential and can be relaxed. In particular, one may take any combination of simultaneous faults and consider it as a higher-dimensional fault signal. For instance, a simultaneous fault on nodes j and k could be modeled using (6.7) by replacing $E_k f_k(t)$ with $[E_k\ E_j][f_k(t)\ f_j(t)]^\top$.

As suggested in Chen and Patton (1999), a possible method of detecting and isolating the faults is to use the so called generalized observer scheme (GOS), where we construct a bank of observers generating a structured set of residuals such that each residual is decoupled from one and only one fault, but being sensitive to all other faults. To achieve distributed FDI using this scheme, each node $i \in \mathcal{V}$ constructs a bank of N_i UIOs. In particular, for each neighbor $k \in \mathcal{N}_i$, an observer decoupled from E_k and $D_{i,k}$ is implemented, as described next. Given the model (6.7), let $\hat{x}_k^i(t)$ denote the state estimate decoupled from a faulty node k and calculated by node i using the state observer

$$\begin{aligned} \dot{z}_k^i(t) &= F_k^i z_k^i(t) + T_k^i B v(t) + K_k^i y_i(t) \\ \hat{x}_k^i(t) &= z_k^i(t) + H_k^i y_i(t), \end{aligned} \tag{6.8}$$

where $z_k^i(t) \in \mathbb{R}^{2N}$ is the observer's state.

An unknown input observer (UIO) decoupled from a faulty node k is defined as follows (Ding, 2008).

Definition 6.3.1. *Consider the dynamical system* (6.7) *and the observer* (6.8)*. The observer is a* UIO decoupled from a faulty node k *if* $\lim_{t\to+\infty} \|x(t) - \hat{x}_k^i(t)\| = 0$ *for any fault* $f_k(t)$*.*

For the observer (6.8) to be an UIO, the observer matrices should be designed to achieve decoupling from the faulty node k and should ensure the stability of the observer. By choosing the matrices $F_k^i, T_k^i, K_k^i, H_k^i$ to satisfy the conditions

$$\begin{aligned} &F_k^i = (A - H_k^i C_i A - K_k'^i C), \quad T_k^i = (I - H_k^i C_i) \\ &K_k^i = K_k'^i + K_k''^i, \quad K_k''^i = F_k^i H_k^i, \quad (H_k^i C_i - I)E_k = 0, \end{aligned} \tag{6.9}$$

where F_k^i is Hurwitz and recalling the model (6.7), we have the estimation error dynamics

$$\dot{e}_k^i(t) = F_k^i e_k^i(t) - T_k^i \sum_{m \in \mathcal{N}_i \setminus \{k\}} E_m f_m(t) \tag{6.10}$$

with $e_k^i(t) = x(t) - \hat{x}_k^i(t)$. Clearly, the error dynamics (6.10) do not depend on $f_k(t)$ and are stable, thus complying with Definition 6.3.1. In general, the UIO existence condition are as follows (Chen and Patton, 1999).

Proposition 6.3.1. *For the system* (6.7)*, there exists an UIO decoupled from a faulty node k in the sense of Definition 6.3.1 if and only if the following conditions hold*

$$\begin{aligned} &rank(C_i E_k) = rank(E_k) \\ &rank \begin{bmatrix} sI - A & E_k \\ C_i & 0 \end{bmatrix} = n + rank(E_k), \end{aligned} \tag{6.11}$$

for all $s \in \mathcal{C}$ *with non-negative real parts.*

Remark 6.3.1. *The UIO existence conditions* (6.11) *correspond to the necessary and sufficient conditions for asymptotic estimation of the unknown input* $f_k(t)$*.*

Consider the fault signal estimate $\hat{f}_k^i(t) = V(\dot{y}_i(t) - CA\hat{x}_k^i(t))$ *with* $V = (C_i E_k)^\dagger$ *as the pseudo-inverse of* $C_i E_k$. *From (Ding, 2008, Theorem 14.4), when* $y(t)$ *and* $\dot{y}(t)$ *are available, the necessary and sufficient conditions for* $\lim_{t\to+\infty} |f_k(t) - \hat{f}_k^i(t)| = 0$ *are the same as the UIO existence conditions in Proposition 6.3.1.*

The UIO error dynamics (6.10) are driven by the j-th fault, for some $j \neq k$, if $T_k^i E_j \neq 0$. In fact, having $T_k^i E_j \neq 0$ for all $j \in \mathcal{N}_i \backslash \{k\}$, for all $k \in \mathcal{N}_i$, plays an important role in the detection and isolation logic later described. This condition can be incorporated in the UIO design, as stated by the following result.

Proposition 6.3.2. *Given the system* (6.7)*, suppose the UIO existence conditions* (6.11) *hold for a given* $k \in \mathcal{N}_i$. *There exists an UIO decoupled from a faulty node* k *with* $T_k^i E_j \neq 0$ *for all* $j \in \mathcal{N}_i \backslash \{k\}$ *if* $\text{rank}(C_i [E_k \, E_j]) = \text{rank}([E_k \, E_j]) > \text{rank}(E_k)$, *for all* $j \in \mathcal{N}_i \backslash \{k\}$.

Proof. The desired UIO must satisfy (6.9) and $T_k^i E_j \neq 0$ for all $j \in \mathcal{N}_i \backslash \{k\}$. Recalling that $T_k^i = (I - H_k^i C_i)$, we then have that $T_k^i E_k = 0$ and $T_k^i E_j \neq 0$ must hold. The rank condition in the proposition's statement ensures that $H_k^i = E_k \left((C_i E_k)^\top C_i E_k\right)^{-1} (C_i E_k)^\top$ satisfies $T_k^i E_k = 0$ and $T_k^i E_j \neq 0$ for all $j \in \mathcal{N}_i \backslash \{k\}$, since E_k and E_j are orthogonal. The rest of the proof follows directly from the UIO design method detailed in Chen and Patton (1999), which constructs an UIO satisfying (6.9) with H_k^i as chosen above. □

Given the conditions in Proposition 6.3.1, we observe that the rank condition in Proposition 6.3.2 holds when there exist UIOs for all $k \in \mathcal{N}_i$ and every pair of fault directions E_k and E_j with $j \neq k$ is linearly independent. Since the latter holds for both node and edge faults, in the remainder of the chapter we focus only on the UIO existence conditions from Proposition 6.3.1. In particular, we derive results of existence and nonexistence of UIOs for the interconnected system (6.1) under different fault models by using the conditions of Proposition 6.3.1.

For the moment, suppose that there exists a bank of UIOs at node i, where each UIO is decoupled from a faulty node $k \in \mathcal{N}_i$. The bank of UIOs computes a set of state estimates $\hat{x}_j^i(t)$ for $j \in \mathcal{N}_i$ given the model of the system (6.7), which is assumed to be accurate. Intuitively, recalling that noise is neglected, a mismatch between the estimated and actual state trajectory of the system would indicate the presence of faults in the system. In fact, node i can detect faults by analyzing the difference between the estimated outputs $\hat{y}_j^i(t) = C_i \hat{x}_j^i(t)$ for all $j \in \mathcal{N}_i$ and the actual measurements $y_i(t)$, which are denoted as residual signals.

Definition 6.3.2. *The signal* $r_j^i(t) \triangleq y_i(t) - C_i \hat{x}_j^i(t) = C_i e_j^i(t)$ *is a residual if* $\|r_j^i(t)\| = 0$ *is equivalent to* $\|f_k(t)\| = 0$ *for all* $k \neq j \in \mathcal{N}_i$.

Note that the residual dynamics of $r_k^i(t)$ are driven by the j-th fault if $T_k^i E_j \neq 0$, which can be ensured for $j \in \mathcal{N}_i \backslash \{k\}$ through Proposition 6.3.2. Therefore, according to Definition 6.3.2, having $\|r_k^i(t)\| > 0$ indicates that there exists a fault in the network other than $f_k(t)$. Additionally, since $r^i_j(t)$ is computed by an UIO decoupled from $f_j(t)$, if the only active fault is $f_j(t)$ we have $\|r^i_j(t)\| = 0$ and $\|r^i_k(t)\| > 0$ for all $k \neq j$. Motivated by this reasoning, we consider the following detection and isolation logic for fault $f_j(t)$ monitored by node i:

$$\begin{aligned} \left\|r^i_j(t)\right\| &< \Theta^i_j \\ \left\|r^i_k(t)\right\| &\geq \Theta^i_k \ , \forall k \neq j, \end{aligned} \tag{6.12}$$

where $\Theta^i_j > 0$ are isolation thresholds. These thresholds should be chosen according to trade-offs between sensitivity to faults, robustness to unmodeled dynamics and noise, misdetection rate, and false alarm rate, among others. Since choosing these thresholds is not within the scope of this work, the reader is referred to Ding (2008) for further discussions.

Using Algorithm 6.3 a faulty node j can be detected and isolated by all the nodes in $\mathcal{N}_j$. However, all the other nodes in the network $i \notin \mathcal{N}_j$ can only detect the existence of a faulty node in the network, which occurs when $\left\|r^i_k(t)\right\| \geq \Theta^i_k \ \forall k \in \mathcal{N}_i$, while the identity of the faulty node is unknown to them. For the ease of notation we drop the superscript i from the variable names for the rest of this chapter.

Algorithm 6.3 Distributed FDI of Faulty Nodes at Node i

```
for $k \in \mathcal{N}_i$ do
    Generate $r^i_k(t)$.
end for
if $\exists j : \left\|r^i_j(t)\right\| < \Theta^i_j$ and $\left\|r^i_k(t)\right\| \geq \Theta^i_k \ \forall k \in \mathcal{N}_i \neq j$ then
    Node $j$ is faulty.
else if $\left\|r^i_k(t)\right\| \geq \Theta^i_k \ \forall k \in \mathcal{N}_i$ then
    There exists a faulty node $\ell \in \mathcal{V} \setminus \mathcal{N}_i$.
else if $\left\|r^i_k(t)\right\| < \Theta^i_k \ \forall k \in \mathcal{N}_i$ then
    There is no faulty node in the network.
end if
```

In the remainder of this section, we address the problem of distributed fault detection and isolation of faulty nodes and faulty edges using the approach outlined here. In particular, we adapt the model (6.7) and distributed FDI scheme to the case of either node or edge faults and, for each case, we study the existence conditions of UIOs in terms of the network graph and available measuremetns.

6.3.1 Distributed FDI for Faulty Nodes

The distributed FDI scheme to detect and isolate faulty nodes is outlined in Algorithm 6.3. However, to solve the distributed FDI problem for faulty nodes using Algorithm 6.3, there needs to exist a bank of UIOs for each node $i \in \mathcal{V}$ satisfying the isolability condition in Proposition 6.3.2. For the case of faulty nodes, the problem of distributed FDI using UIOs can be stated as follows.

Problem 6.3.1. *Consider the networked system* (6.1) *and faulty nodes as in Definition 6.2.2. The answer to the following question is sought:*

> *Consider the node j to be faulty, and let node i be a neighbor of j. Does there exist an UIO for node i that is decoupled from the faulty node j?*

The answer to Problem 6.3.1 is provided next by proving the existence of matrices $F_k^i, T_k^i, K_k^i, H_k^i$ satisfying (6.9) for the system (6.7) with node faults and local measurements (6.3) for all $i \in \mathcal{V}$.

Theorem 6.3.3. *Consider the networked system* (6.7) *with a fault at node k. In the sense of Definition 6.3.1, there exists an UIO decoupled from the faulty node k for node i if the graph $\mathcal{G}$ is connected and nodes k and i are neighbors.*

Proof. For a faulty node $k \in \mathcal{N}_i$, the system dynamics and measurement equations are given by (6.7) with $E_k = b_k$ and $D_{i,k} = 0$. Observing that the measurements at node i are not corrupted, next we show that the UIO existence conditions in Proposition 6.3.1 are satisfied. It follows that the first rank condition in Proposition 6.3.1 holds because

$$\operatorname{rank}(C_i E_k) = \operatorname{rank}(E_k^\top E_k) = \operatorname{rank}(E_k),$$

where the first equality follows from the fact node i measures the states its neighboring nodes, including the faulty node k.

As for the second rank condition in (6.11), consider the 1-hop neighborhood graph of node i, $\mathcal{G}_i$, with $\mathcal{V}_i = \{i\} \cup \mathcal{N}_i$ and $V_i = |\mathcal{V}_i|$. Denote $\tilde{\mathcal{G}}_i$ as the subgraph induced by the vertex set $\tilde{\mathcal{V}}_i = \mathcal{V} \backslash \mathcal{V}_i$, with $\tilde{V}_i = |\tilde{\mathcal{V}}_i|$. Without loss of generality, the nodes may be rearranged so that the Laplacian of $\mathcal{G}$ and E_k can be written as

$$\mathcal{L} = \begin{bmatrix} \mathcal{L}_i & \ell_i \\ \ell_i^\top & \tilde{\mathcal{L}}_i \end{bmatrix}, \quad E_k = \begin{bmatrix} 0_N \\ l_k \\ 0_{\tilde{V}_i} \end{bmatrix}$$

where $\ell_i \in \mathbb{R}^{V_i \times \tilde{V}_i}$ and the vector $l_k \in \mathbb{R}^{V_i}$ is the k-th column of I_{V_i}. The second rank condition in (6.11) becomes

$$\operatorname{rank} \underbrace{\begin{bmatrix} sI_{V_i} & 0_{V_i \times \tilde{V}_i} & -I_{V_i} & 0_{V_i \times \tilde{V}_i} & 0_{V_i} \\ 0_{\tilde{V}_i \times V_i} & sI_{\tilde{V}_i} & 0_{\tilde{V}_i \times V_i} & -I_{\tilde{V}_i} & 0_{\tilde{V}_i} \\ \mathcal{L}_i & \ell_i & \alpha_1(s) & \mu\ell_i & l_k \\ \ell_i^\top & \tilde{\mathcal{L}}_i & \mu\ell_i^\top & \alpha_2(s) & 0_{\tilde{V}_i} \\ I_{V_i} & 0_{V_i \times \tilde{V}_i} & 0_{V_i \times V_i} & 0_{V_i \times \tilde{V}_i} & 0_{V_i} \\ 0_{V_i \times V_i} & 0_{V_i \times \tilde{V}_i} & I_{V_i} & 0_{V_i \times \tilde{V}_i} & 0_{V_i} \end{bmatrix}}_{P} = 2N + 1,$$

where $\alpha_1(s) = sI_{V_i} + \mu\mathcal{L}_i + \bar{K}_i$ and $\alpha_2(s) = sI_{\tilde{V}_i} + \mu\tilde{\mathcal{L}}_i + \tilde{K}_i$.

Observing that the first and third column blocks are linearly independent of the rest and applying some row and column operations we have

$$\operatorname{rank}(P) = \operatorname{rank}\begin{bmatrix} -\frac{1}{\mu}I_{\tilde{V}_i} & -(1+\mu s)I_{\tilde{V}_i} & 0_{\tilde{V}_i} \\ \ell_i & 0_{V_i\times\tilde{V}_i} & l_k \\ 0_{\tilde{V}_i\times\tilde{V}_i} & -\alpha(s) & 0_{\tilde{V}_i} \end{bmatrix} + 2V_i,$$

with $\alpha(s) = \mu s^2 I_{\tilde{V}_i} + \mu s(\tilde{\mathcal{L}}_i + \tilde{K}_i) + \tilde{\mathcal{L}}_i$. It follows from Barooah and Hespanha (2006) that $\tilde{\mathcal{L}}_i$ is positive definite if $\mathcal{G}$ is connected. Since $\mu > 0$ and $\tilde{K}_i$ are positive definite, we conclude that $\alpha(s)$ is invertible for $s \in \mathbb{C}$ with non-negative real part. Therefore the first and second column blocks are independent of each other and the third column block, which concludes the proof. □

In particular, the existence conditions of Proposition 6.3.1 reduce to having the graph $\mathcal{G}$ connected and $k \in \mathcal{N}_i$. Therefore we make the following assumption for the remaining of this chapter.

Assumption 6.3.1. *The network graph $\mathcal{G}$ is connected.*

6.3.2 Distributed FDI for Faulty Edges

In this subsection we extend the distributed FDI scheme to the case of faulty edges as in Definition 6.2.3. Similarly to the detection and isolation scheme outlined for node faults in Section 6.3.1, faults on edges may also be detected and isolated using banks of UIOs. This subsection analyzes the existence of suitable UIOs that may be used to detects faulty edges. In particular, the following problem is addressed.

Problem 6.3.2. *Consider the networked system* (6.1) *and faulty edges as in Definition 6.2.3. The answers to the following two questions are sought:*

1. *Consider the edge between nodes j and k to be faulty, and let node i be a neighbor of both j and k. Does there exist an UIO for node i that is decoupled from the faulty edge $\{j,k\}$?*
2. *Does there exist an UIO for node i that is decoupled from a faulty edge incident to node i?*

First we consider the problem of distributed detection and isolation of those faults that appear as corruptions in the communication or sensing links between pairs of neighbors characterized by Definition 6.2.3.(i). Later the detection and isolation of edge parameter faults described in Definition 6.2.3.(ii) is tackled.

To address the problem of distributed detection and isolation of faulty edges, in addition to the bank of observers monitoring the fault in the neighbor nodes of a given node i to detect misbehaving nodes, we construct a bank of observers for those pairs of nodes neighboring to i that share the same edge. Hence at each node i, in addition to the observers for system models described by (6.7), observers for the following systems are constructed for all $\{j, k\} \in \mathcal{E}_i$:

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bv(t) + E_{jk}f_{jk}(t) + E_{kj}f_{kj}(t) \\ y_i(t) &= C_i x(t) + D_{i,jk}f_{jk}(t) + D_{i,kj}f_{kj}(t) \end{aligned} \tag{6.13}$$

where $f_{jk}(t) = [f^{\xi}_{jk}(t)\ f^{\zeta}_{jk}(t)]^{\top}$, $E_{jk} = b_j[w_{jk}\ \mu w_{jk}]$, $D_{i,ij} = C_i[l_j\ b_j]$, and $D_{i,jk} = 0$ for $j \neq i$. Similarly as before, let $\hat{x}_{jk}(t)$ denote the estimate of the states for this system model and define the UIO decoupled from a faulty edge $\{j, k\}$ and the respective residual signal as follows.

Definition 6.3.3. *Consider the dynamical system* (6.13) *and the observer* (6.8)*. The observer is a* UIO decoupled from a faulty edge $\{j, k\}$ *if* $\lim_{t\to+\infty} \|x(t) - \hat{x}^i_{jk}(t)\| = 0$ *for any fault signals* $f_{jk}(t)$ *and* $f_{kj}(t)$*.*

Definition 6.3.4. *The signal* $r_{jk}(t) \triangleq y_i(t) - C_i\hat{x}_{jk}(t)$ *is a residual if* $\|r_{jk}(t)\| = 0$ *is equivalent to* $\|f_{\bar{j}\bar{k}}(t)\| = \|f_{\bar{k}\bar{j}}(t)\| = 0$ *for all* $\{\bar{j}, \bar{k}\} \neq \{j, k\} \in \mathcal{E}_i$*.*

As seen in (6.13), the corrupted data sent along the faulty edge affects the dynamics of the node at the receiving end. In fact, comparing with the formulation in Pasqualetti *et al.* (2012), such false data appear in the dynamics as two concurrent faulty nodes. However, note that the measurements $y_i(t)$ may also be affected by the edge fault. The following proposition establishes the existence of such observers for the system described above and addresses the first question posed in Problem 6.3.2.

Theorem 6.3.4. *Consider the networked system* (6.13) *with a sensing fault at the edge* $\{j, k\}$ *and* $j, k \neq i$*. In the sense of Definition 6.3.3, there exists an UIO decoupled from the faulty edge* $\{j, k\}$ *for node* i *if the graph* $\mathcal{G}$ *is connected and node* i *is a neighbor of both* j *and* k*.*

Proof. For node $i \in \mathcal{N}_j \cap \mathcal{N}_k$, the system dynamics and measurement equations are given by (6.13) with $E_{jk} = b_j[w_{jk}\ \mu w_{jk}]$ and $D_{i,jk} = 0$. Observing that the measurements at node i are not corrupted and defining $f^e_{jk}(t) = w_{jk}f^{\xi}_{jk}(t) + \mu w_{jk}f^{\zeta}_{jk}(t)$, the model can be rewritten as two simultaneous node faults:

$$\begin{aligned} \dot{x}(t) &= Ax(t) + E_{\{j,k\}}[f^e_{jk}(t)\ f^e_{kj}(t)]^{\top} \\ y_i(t) &= C_i x(t), \end{aligned}$$

with $E_{\{j,k\}} = [b_j\ b_k]$. Next we show that the UIO existence conditions in Proposition 6.3.1 are satisfied. It follows that the first rank condition in Proposition 6.3.1 holds because

$$\operatorname{rank}(C_i E_{\{j,k\}}) = \operatorname{rank}(E^{\top}_{\{j,k\}}E_{\{j,k\}}) = \operatorname{rank}(E_{\{j,k\}}),$$

where $\operatorname{rank}(C_i E_{\{j,k\}}) = \operatorname{rank}(E^{\top}_{\{j,k\}}E_{\{j,k\}})$ follows from the fact node i measures the states of nodes j and k that are affected by the fault.

As for the second rank condition in (6.11), it is the same as the case where two concurrent faults occur in the system, so the proof is similar to that of Theorem 6.3.3. Consider the 1-hop neighborhood graph of node i, $\mathcal{G}_i$, with $\mathcal{V}_i = \{i\} \cup \mathcal{N}_i$ and $V_i = |\mathcal{V}_i|$. Denote $\tilde{\mathcal{G}}_i$ as the subgraph induced by the vertex set $\tilde{\mathcal{V}}_i = \mathcal{V} \backslash \mathcal{V}_i$, with $\tilde{V}_i = |\tilde{\mathcal{V}}_i|$. Without loss of generality, the nodes may be rearranged so that the Laplacian of $\mathcal{G}$ and $E_{\{j,k\}}$ can be written as

$$\mathcal{L} = \begin{bmatrix} \mathcal{L}_i & \ell_i \\ \ell_i^\top & \tilde{\mathcal{L}}_i \end{bmatrix}, \quad E_{\{j,k\}} = \begin{bmatrix} 0_{N\times 2} \\ l_{jk} \\ 0_{\tilde{V}_i \times 2} \end{bmatrix}$$

where $\ell_i \in \mathbb{R}^{V_i \times \tilde{V}_i}$ and the columns of $l_{jk} \in \mathbb{R}^{V_i \times 2}$ are the columns of I_{V_i} corresponding to nodes j and k. The second rank condition in (6.11) becomes

$$\operatorname{rank} \underbrace{\begin{bmatrix} sI_{V_i} & 0_{V_i\times\tilde{V}_i} & -I_{V_i} & 0_{V_i\times\tilde{V}_i} & 0_{V_i\times 2} \\ 0_{\tilde{V}_i\times V_i} & sI_{\tilde{V}_i} & 0_{\tilde{V}_i\times V_i} & -I_{\tilde{V}_i} & 0_{\tilde{V}_i\times 2} \\ \mathcal{L}_i & \ell_i & \alpha_1(s) & \mu\ell_i & l_{jk} \\ \ell_i^\top & \tilde{\mathcal{L}}_i & \mu\ell_i^\top & \alpha_2(s) & 0_{\tilde{V}_i\times 2} \\ I_{V_i} & 0_{V_i\times\tilde{V}_i} & 0_{V_i\times V_i} & 0_{V_i\times\tilde{V}_i} & 0_{V_i\times 2} \\ 0_{V_i\times V_i} & 0_{V_i\times\tilde{V}_i} & I_{V_i} & 0_{V_i\times\tilde{V}_i} & 0_{V_i\times 2} \end{bmatrix}}_{P} = 2N+2,$$

where $\alpha_1(s) = sI_{V_i} + \mu\mathcal{L}_i + \bar{K}_i$ and $\alpha_2(s) = sI_{\tilde{V}_i} + \mu\tilde{\mathcal{L}}_i + \tilde{K}_i$.

Observing that the first and third column blocks are linearly independent of the rest and applying some row and column operations we have

$$\operatorname{rank}(P) = \operatorname{rank} \begin{bmatrix} -\frac{1}{\mu} I_{\tilde{V}_i} & -(1+\mu s)I_{\tilde{V}_i} & 0_{\tilde{V}_i\times 2} \\ \ell_i & 0_{V_i\times\tilde{V}_i} & l_{jk} \\ 0_{\tilde{V}_i\times\tilde{V}_i} & -\alpha(s) & 0_{\tilde{V}_i\times 2} \end{bmatrix} + 2V_i,$$

with $\alpha(s) = \mu s^2 I_{\tilde{V}_i} + \mu s(\tilde{\mathcal{L}}_i + \tilde{K}_i) + \tilde{\mathcal{L}}_i$. It follows from Barooah and Hespanha (2006) that $\tilde{\mathcal{L}}_i$ is positive definite if $\mathcal{G}$ is connected. Since $\mu > 0$ and $\tilde{K}_i$ are positive definite, we conclude that $\alpha(s)$ is invertible for $s \in \mathbb{C}$ with non-negative real part. Therefore the first and second column blocks are independent of each other and the third column block, which concludes the proof. □

Moreover we have the following result stating that, for any node i, an observer decoupled from a faulty edge incident to i cannot be constructed. It addresses the second question posed in Problem 6.3.2.

Proposition 6.3.5. *Consider the networked system* (6.13) *with a sensing fault at the edge* $\{i,j\}$*. In the sense of Definition 6.3.3, there does not exist an UIO decoupled from the faulty edge* $\{i,j\}$ *for node* i*.*

Proof. Consider a faulty edge $\{i, j\}$ incident to node i with a sensing fault. Recalling (6.13), the system dynamics and measurement equations can be rewritten as

$$\begin{aligned}\dot{x}(t) &= Ax(t) + Bv(t) + E_{\{i,j\}} f_{\{i,j\}}(t)\\ y_i(t) &= C_i x(t) + D_{i,\{i,j\}} f_{\{i,j\}}(t)\end{aligned}$$

where $f_{\{i,j\}}(t) = [f_{ij}^\top(t)\ f_{ji}^\top(t)]^\top$, $E_{\{i,j\}} = [E_{ij}\ E_{ji}]$ and $D_{i,\{i,j\}} = [D_{i,ij}\ 0]$. From Ding (2008, Chapter 2) we recall that the following rank condition should hold for the existence of UIOs:

$$\operatorname{rank}\begin{bmatrix} D_{i,\{i,j\}} & C_i E_{\{i,j\}} \\ 0 & D_{i,\{i,j\}} \end{bmatrix} = \operatorname{rank}(D_{i,\{i,j\}}) + \operatorname{rank}\begin{bmatrix} E_{\{i,j\}} \\ D_{i,\{i,j\}} \end{bmatrix},$$

where the second term equals 5. Given $C_i E_{\{i,j\}}$ and $D_{i,\{i,j\}}$, the first term of the latter rank condition can be written as

$$\operatorname{rank}\begin{bmatrix} C_i l_j & C_i b_j & C_i b_i w_{ij} & C_i b_i \mu w_{ij} \\ 0 & 0 & C_i l_j & C_i b_j \end{bmatrix} \le 4,$$

since each column-block is a column vector. Since the rank condition is not fulfilled, there does not exist an UIO for this system. □

Although in the case of bidirectional sensing faults in edges there is no UIO for the nodes to which the faulty edge is incident to, the following result shows that this is not the case for unidirectional faults, i.e., for the case where either $f_{ij}(t)$ or $f_{ji}(t)$ is identically zero. We formalize this case in what follows.

Proposition 6.3.6. *Consider the networked system* (6.13) *with a sensing fault at the edge* $\{i, j\}$. *In the sense of Definition 6.3.3, if the graph* $\mathcal{G}$ *is connected, for node* i *there exists an UIO decoupled from*

1. *The sensing fault from node* j *to node* i, $f_{ij}(t)$, *when* $f_{ji}(t) \equiv 0$.
2. *The sensing fault from node* i *to node* j, $f_{ji}(t)$, *when* $f_{ij}(t) \equiv 0$.

Proof. In the first case, the dynamical system with respect to node i and the faulty edge $\{i, j\}$ is described by (6.13) with $E_{ij} = b_i[w_{ij}\ \mu w_{ij}]$, $E_{ji} = 0$, $D_{ij} = C_i[l_j\ b_j]$, and $D_{ji} = 0$. Now consider that the measurements corresponding to node j have been removed, yielding the following system

$$\begin{aligned}\dot{x}(t) &= Ax(t) + Bv(t) + E_{ij} f_{ij}(t),\\ \tilde{y}_i(t) &= \tilde{C}_i x(t),\end{aligned}$$

which corresponds to the model of a single node fault at node i and measurements from $\mathcal{V}_i^1 \backslash \{j\}$. From Theorem 6.3.3, it follows that an UIO exists for this system.

In the second case, the dynamical system with respect to node i is described by

$$\begin{aligned}\dot{x}(t) &= Ax(t) + Bv(t) + E_{ji}f_{ji}(t)\\ y_i(t) &= C_i x(t)\end{aligned}$$

which also corresponds to a single node fault at node j and, similarly to the previous case, the corresponding UIO exists. □

In the following we consider faulty edges with parameter faults, as described in Definition 6.2.3.(ii). For detecting and isolating these faults at each node i, in addition to the observers for system models described by (6.7), observers for the following systems are constructed at each node i for all $\{j,k\} \in \mathcal{E}_i$:

$$\begin{aligned}\dot{x}(t) &= Ax(t) + Bv(t) + E_{jk}f_{jk}(t)\\ y_i(t) &= C_i x(t)\end{aligned} \tag{6.14}$$

where $E_{jk} = b_j - b_k$ and $f_{jk}(t) = \delta_{jk}(t)f^w_{jk}(t)$. The existence of UIO's for (6.14) is a consequence of the results establishing the existence of UIO's for faulty nodes and will not be stated here for brevity.

Under the assumption that a single fault occurs at any given time, the following algorithm may be implemented at each node to simultaneously detect and isolate faulty nodes and edges.

Algorithm 6.4 Distributed FDI of Faulty Nodes and Edges at Node i

```
for $j \in \mathcal{N}_i$ do
  Generate $r_j(t)$.
end for
for $\{j,k\} \in \mathcal{E}_i$ do
  Generate $r_{jk}(t)$.
end for
if $\exists k : \|r_k(t)\| < \Theta_k$ and $\|r_j(t)\| \geq \Theta_j,\ \forall j \in \mathcal{N}_i \neq k$ then
  Node $k$ is faulty.
end if
if $\exists\{\bar{j},\bar{k}\} : \left\|r_{\bar{j}\bar{k}}(t)\right\| < \Theta_{\{\bar{j},\bar{k}\}}$ and $\|r_j(t)\| \geq \Theta_j,\ \forall j \in \mathcal{N}_i \neq k$ and $\|r_{jk}(t)\| \geq \Theta_{\{j,k\}},\ \forall\{j,k\} \in \mathcal{E}_i \neq \{\bar{j},\bar{k}\}$ then
  Edge $\{\bar{j},\bar{k}\}$ is faulty.
end if
if $\|r_j(t)\| \geq \Theta_j\ \forall j \in \mathcal{N}_i$ and $\|r_{jk}(t)\| \geq \Theta_{\{j,k\}}\ \forall\{j,k\} \in \mathcal{E}_i$ then
  There exists a faulty node or edge in $\mathcal{G}\backslash\mathcal{G}_i$.
end if
if $\|r_j(t)\| < \Theta_j\ \forall j \in \mathcal{N}_i$ and $\|r_{jk}(t)\| < \Theta_{\{j,k\}}\ \forall\{j,k\} \in \mathcal{E}_i$ then
  There is no faulty node or edge in the network.
end if
```

6.4 Distributed FDI with Imprecise Network Models

As described earlier, to construct a bank of observers achieving distributed FDI given the local measurements (6.3), the knowledge of the system matrix A is needed. In this section we study the case where, after having designed observers under a known network model and interconnection graph, some edges and nodes are removed. The edge and node removal may correspond to either unexpected changes in the system, or the removal of faulty edges and nodes. In both scenarios, it is desirable to maintain the detection and isolation capabilities of the distributed FDI scheme despite the model changes. Later in this section we show that a distributed FDI scheme does not require the full knowledge of the network. Now we are ready to pose the following problem.

Problem 6.4.1. *Consider a network and a bank of observers as described in section 6.3.1 and 6.3.2. Suppose the network loses l edges. What are the necessary and sufficient conditions ensuring that node i can detect faults in the network using the bank of observers and Algorithm 6.4?*

Note that removing a node corresponds to removing all the edges incident to it, thus the case of node removal is covered by the previous problem.

6.4.1 Distributed FDI with Global Models

We first address Problem 6.4.1 when the global model (6.7) is used to design the UIOs. Consider the case where we design a bank of UIO's to estimate the states of the neighbors of node i and recall that we have the following observer error and residual dynamics

$$\begin{aligned} \dot{e}_k(t) &= F_k e_k(t) - T_k \sum_{m \in \mathcal{N}_i \setminus \{k\}} E_m f_m(t) \\ r_k(t) &= C_i e_k(t). \end{aligned}$$

Introduce $\mathcal{E}_{loss} \subseteq \mathcal{E}$ as the subset of edges removed from the network. Recalling the system dynamics (6.7), under edge removal the new system and output matrices A_ℓ and $C_{i\ell}$, respectively, are given by

$$\begin{aligned} A_\ell &= A + \Delta A, \\ C_{i\ell} &= C_i + \Delta C_i. \end{aligned}$$

The matrices ΔA and ΔC_i are perturbation matrices corresponding to the lost edges. More precisely, $\Delta A = \begin{bmatrix} 0_N & 0_N \\ \mathcal{L}_{loss} & \mu \mathcal{L}_{loss} \end{bmatrix}$, where $\mathcal{L}_{loss}$ is the Laplacian matrix corresponding to the graph $\mathcal{G}_{loss}(\mathcal{V}, \mathcal{E}_{loss})$. Moreover, all the entries of ΔC_i are zero except those entries that correspond to a neighbor of i whose shared edge with i is in $\mathcal{E}_{loss}$, which are all equal to -1. We have the following assumption.

Assumption 6.4.1. *The network remains connected after losing the edges $\mathcal{E}_{loss}$.*

Using the existing parameters of the UIO (computed under the assumption of no edge loss), the error dynamics are characterized by

$$\begin{aligned}\dot{e}_k(t) =&F_k e_k(t) + \Delta A x(t) + H_k C_i \Delta A x(t) + H_k \Delta C_i \Delta A x(t)\\ &- K_k \Delta C_i x(t) - T_k \sum_{m\in\mathcal{N}_i\backslash\{k\}} E_m f_m(t).\end{aligned}$$

If the removed links had not been connecting i to any of its neighbors, we have $\Delta C_i = 0$. It is easy to check that then the error dynamics become

$$\dot{e}_k(t) =F_k e_k(t) + (I + H_k C_i)\Delta A x(t) - T_k \sum_{m\in\mathcal{N}_i\backslash\{k\}} E_m f_m(t). \tag{6.15}$$

The error dynamics described by (6.15), in the presence of no faults for $m \in \mathcal{V}\backslash\{k\}$, $f_m(t) \equiv 0$, are

$$\dot{e}_k(t) =F_k e_k(t) + (I + H_k C_i)\Delta A x(t). \tag{6.16}$$

Assume for the moment that the known input $v(t)$ is zero. Recall from Remark 6.2.1 that, if the network is connected, $x(t)$ converges exponentially to $[\bar{\xi}\ \bar{\zeta}]^\top \otimes \mathbf{1}_{2N}$ when there is no fault. Given the structure of ΔA and recalling that $\mathcal{L}\mathbf{1}_N = 0$ for any Laplacian matrix $\mathcal{L} \in \mathbb{R}^{N\times N}$, it follows that $\Delta A x(t)$ goes exponentially fast to zero when there is no fault in the network. Therefore, since F_k is Hurwitz, the error dynamics described by (6.16) are stable. Consequently $r_k(t) = C_i e_k(t)$ goes to zero when there is no fault in the system, although the UIO parameters are designed for a different interconnection network. However, if the input $v(t)$ does not drive the system to consensus, i.e. $\|x_i(t) - x_j(t)\|$ does not go to zero as t goes to infinity, then $\Delta A x(t)$ does not generically converge to zero when there is no fault, and neither does the residual $r_k(t)$.

On the other hand, if any of the removed edges had been connecting i to one of its neighbors, the error dynamics may not even converge to zero when there is no fault. In particular, suppose there are no faults and that the system has reached an equilibrium so that $\Delta A x(t) = 0$, yielding the error dynamics

$$\dot{e}_k(t) =F_k e_k(t) - K_k \Delta C_i x(t).$$

Since in general $K_k \Delta C_i x(t)$ is not identically zero at the equilibrium, we conclude that the error does not converge to zero and thus $r_k(t)$ is not a suitable residual, as it violates Definition 6.3.2. Hence, the bank of observers should be redesigned taking into account the updated network model. Formally, we have the following result that addresses Problem 6.4.1.

Theorem 6.4.1. *Consider a monitoring node i in an arbitrary connected network described by* (6.1) *and a bank of UIO's for this network. Using Algorithm 6.3 and* *the existing bank of observers, node i can detect the presence of a faulty node after the loss of ℓ edges if and only if all the following conditions are satisfied: (1) the network remains connected, (2) $v(t)$ is such that $\|x_i(t) - x_j(t)\| \to 0$ as $t \to \infty$, i.e. it drives the system to consensus, and (3) $\mathcal{N}_i$ is the same as in the original network.*

Proof. Consider the original graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$ and let $k \in \mathcal{N}_i$. Suppose ℓ edges in set $\tilde{\mathcal{E}}$ are lost and the corresponding subgraph to these edges is denoted by $\tilde{\mathcal{G}}(\mathcal{V}, \tilde{\mathcal{E}})$. Since node i cannot detect faults in network components that are not connected to it, a necessary condition is that the subgraph $\tilde{\mathcal{G}}$ remains connected, which corresponds to condition (1). Regarding condition (2), another necessary condition is that $v(t)$ drives the system to consensus, thus ensuring that $\Delta Ax(t)$ does converge to zero. Additionally, having $\Delta C_i = 0$, or equivalently $k \in \tilde{\mathcal{N}_i}$ for all $k \in \mathcal{N}_i$, as in (3), is also a necessary condition. Otherwise, in general the residuals do not converge to zero.

Now suppose all the necessary conditions (1)–(3) hold. When there is no fault in the network, $e_k(t)$ goes to zero and as a result $\|r_k(t)\|$ goes to zero as well. For the faulty case, $\|r_k(t)\|$ will generically not converge to zero for $k \in \mathcal{N}_i$. Hence, using Algorithm 6.3 one can detect if there is a fault in the network or not. □

Note that the faulty node cannot be isolated using the condition given by (6.12) when the network model is imprecise. Moreover, detection is also not feasible when the system is not driven to consensus by $v(t)$. These limitations follow from the fact that $\Delta Ax(t)$ does not go to zero because, in general, $x(t)$ does not reach consensus under the fault $f_k(t)$. Thus, the error of the UIO monitoring the neighbor node k converges to a ball around zero with a nonzero radius. Hence, none of the residuals goes to zero so (6.12) cannot be used to isolate the faulty node.

A possible way to overcome such limitations is to use additional measurements from outside each node's neighborhood and design the bank of UIOs using local models of the system that are not not affected by changes in other parts of the network. In particular, we consider the following problem.

Problem 6.4.2. *For a given node i, consider a subgraph of the network $\tilde{\mathcal{G}}_i$ containing the 1-hop neighborhood graph $\mathcal{G}_i$. Let any state measurement within $\tilde{G}_i$ be available to node i. The following questions are considered:*

1. *For which subgraphs can node i design a bank of UIOs and implement Algorithm 6.3 to detect and isolate faults in any of its neighbors?*

2. *Given the set of subgraphs for which an UIO-based FDI scheme exists, which subgraph $\tilde{\mathcal{G}}_i$ minimizes the number of edges in $\tilde{\mathcal{G}}_i$ and required state measurements?*

In what follows we propose a method to address the problem of isolating the faulty nodes and edges in the network, and tackle Problem 6.4.2.

6.4.2 Distributed FDI with Local Models

Consider a fault-free network $\mathcal{G}(\mathcal{V},\mathcal{E})$ with the system dynamics $\dot{x}(t) = Ax(t) + Bv(t)$. Define $\hat{\mathcal{G}}_i$ as a subgraph containing the proximity subgraph of node i, $\mathcal{P}_i \subseteq \hat{\mathcal{G}}_i \subseteq \mathcal{G}(\mathcal{V},\mathcal{E})$. Let $\mathcal{B}(\hat{\mathcal{V}}_i) \subseteq \hat{\mathcal{V}}_i$ be the boundary vertex set such that $\ell \in \mathcal{B}(\hat{\mathcal{V}}_i)$, if $\{\ell, \bar{\ell}\} \in \mathcal{E}$ and $\bar{\ell} \notin \hat{\mathcal{V}}_i$.

The dynamics of the subsystem associated with $\hat{\mathcal{G}}_i$ are

$$\dot{\phi}^i(t) = A^i_{\hat{\mathcal{G}}}\phi^i(t) + \psi^i(t) + B^i_{\hat{\mathcal{G}}}v^i_{\hat{\mathcal{G}}}(t), \tag{6.17}$$

where $\phi_i = [\xi_i \ \xi_{i_1} \ \ldots \ \xi_{i_{|\hat{\mathcal{V}}_i|}} \ \zeta_i \ \zeta_{i_1} \ \ldots \ \zeta_{i_{|\hat{\mathcal{V}}_i|}}]$, $i_m \in \hat{\mathcal{V}}_i$. Particularly i_1 to $i_{|\mathcal{N}_i|}$ are associated with the nodes in $\mathcal{N}_i$. Moreover, $A^i_{\hat{\mathcal{G}}}$ is the matrix associated with the network with $\hat{\mathcal{G}}_i$ as its graph, $\psi^i(t)$ is an unknown vector with zero entries except for the entries corresponding to nodes $j \in \mathcal{B}(\hat{\mathcal{V}}_i)$ that represents the interaction of the rest of the network with the subnetwork of interest. Additionally, $v^i_{\hat{\mathcal{G}}}(t)$ is an input vector in this subnetwork known to i, and $B^i_{\hat{\mathcal{G}}}$ is the input matrix associated with these inputs. We have the following straightforward result for $\psi^i(t)$.

Proposition 6.4.2. *In the network induced by the proximity graph of node i as described by* (6.17)*, $\psi^i(t)$ goes to zero exponentially fast for $v(t) \equiv 0$.*

Proof. The proof is a direct consequence of the exponential stability of (6.1) to the consensus equilibrium and the distributed control law (6.4). □

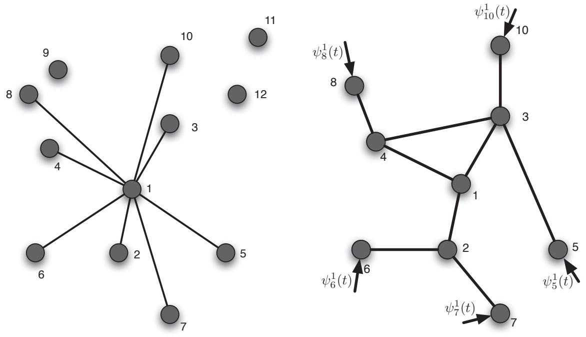
The bank of UIOs at i can be designed for the subnetwork with $\hat{\mathcal{G}}_i$ as its graph and dynamics described by (6.17). An example of such a subnetwork for the network of Figure 6.2 when $\hat{\mathcal{G}}_i = \mathcal{P}_i$ is given in Figure 6.3 (b).

In the case where there is no fault in the network and $v(t) \equiv 0$, the unknown parts of the real network enter the equation dynamics as exponentially decaying signals. As before, in this case the detection of a fault can be determined using the bank of UIOs for $\hat{\mathcal{G}}_i$. Moreover, isolation can be achieved by choosing an appropriate threshold value.

However, the selection of the threshold might be cumbersome, and it requires a knowledge of the magnitude of the fault. In what comes next we propose a method to achieve distributed FDI using only the full knowledge of the subgraph graph $\hat{\mathcal{G}}_i$, without resorting to complicated ways of choosing the threshold value and allowing $v(t) \not\equiv 0$. Given $\hat{\mathcal{G}}_i$, let $\mathcal{S}_i(\hat{\mathcal{V}}_i) \subseteq \hat{\mathcal{V}}_i$ be the set of the nodes for which node i measures states. We make the following assumption that will be valid until the end of this section.

Assumption 6.4.2. *For each node $i \in \mathcal{V}$ and the corresponding subgraph $\hat{\mathcal{G}}_i(\hat{\mathcal{V}}_i, \hat{\mathcal{E}}_i) \subseteq \mathcal{G}(\mathcal{V},\mathcal{E})$ containing the proximity graph $\mathcal{P}_i$, the state measurements of nodes in $\mathcal{S}_i(\hat{\mathcal{V}}_i) = \{i\} \cup \mathcal{N}_i \cup \mathcal{B}(\hat{\mathcal{V}}_i)$ are available to node i.*

An example for the measurement graph of node i is given in Figure 6.3(a). As



(a) (b)

Figure 6.3: (a) An example of a measurement graph of node i in the network of Figure 6.2 under Assumption 6.4.2. (b) The subnetwork used for designing a bank of UIO's at node 1 of the network depicted in Figure 6.2.

before, to achieve the fault detection and isolation task each node i considers $|\mathcal{N}_i|$ models of the form:

$$\dot{\phi}^i(t) = A^i_{\hat{\mathcal{G}}}\phi^i(t) + \psi^i(t) + B^i_{\hat{\mathcal{G}}}v^i_{\hat{\mathcal{G}}}(t) + E^i_k f_k(t) \tag{6.18}$$

where E^i_k is a vector of zeros except for the entry corresponding to node $k \in \mathcal{N}_i$, which is equal to one. We rewrite (6.18) as

$$\dot{\phi}^i(t) = A^i_{\hat{\mathcal{G}}}\phi^i(t) + B^i_{\hat{\mathcal{G}}}v^i_{\hat{\mathcal{G}}}(t) + \begin{bmatrix} E^i & E^i_k \end{bmatrix} \begin{bmatrix} \psi^i(t) \\ f_k(t) \end{bmatrix}, \tag{6.19}$$

with $E^i = [E^i_{m_1} \ \dots \ E^i_{m_{|\mathcal{B}(\hat{\mathcal{V}}_i)|}}]$, where $E^i_{m_l}$, $m_l \in \mathcal{B}(\hat{\mathcal{V}}_i)$, is a vector of zeros except for the entry corresponding to node $m_l \in \mathcal{B}(\hat{\mathcal{V}}_i)$ that is equal to one. For each of these models, a UIO that is decoupled from the unknown input $\begin{bmatrix} E^i & E^i_k \end{bmatrix} \begin{bmatrix} \psi^i(t) \\ f_k(t) \end{bmatrix}$ is designed.

Lemma 6.4.3. *Consider the distributed control system with a fault in node $j \in \mathcal{N}_i$ given by* (6.17) *and measurements satisfying Assumption 6.4.2. In the sense of Definition 6.3.1, there exists an UIO for node i that is decoupled from the faulty node j and the subgraph $\mathcal{V}\backslash\hat{\mathcal{V}}_i$.*

Proof. Recall the UIO existence condition in Proposition 6.3.1. From Assumption 6.4.2, node i measures its own states, as well as the states of nodes $j \in \mathcal{B}(\hat{\mathcal{V}}_i)$ and $j \in \mathcal{N}_i$, which are the ones affected by the unknown input $\psi^i(t)$ and the fault $f_j(t)$, respectively. Therefore it follows that $\text{rank}(C_iE^i) = \text{rank}(E^{i\top}E^i)$ and $\text{rank}(C_iE^i_k) = \text{rank}(E^{i\top}_kE^i_k)$, thus the first rank condition holds.

As for the second rank condition in (6.11), consider the subgraph $\tilde{\mathcal{G}}_i$ induced by the vertex set $\tilde{\mathcal{V}}_i = \mathcal{B}(\hat{\mathcal{V}}_i)$ with $\tilde{V}_i = |\tilde{\mathcal{V}}_i|$. Denote $\bar{\mathcal{G}}_i$ as the subgraph induced by the vertex set $\bar{\mathcal{V}}_i = \hat{\mathcal{V}}_i \backslash \tilde{\mathcal{V}}_i$, with $\bar{V}_i = |\bar{\mathcal{V}}_i|$ and note that $\hat{V}_i \triangleq |\hat{\mathcal{V}}_i| = \tilde{V}_i + \bar{V}_i$. Without loss of generality, the nodes may be rearranged so that the Laplacian of $\hat{\mathcal{G}}_i$, E^i_k, E^i, and C_i can be written as

$$\hat{\mathcal{L}} = \begin{bmatrix} \bar{\mathcal{L}}_i & \bar{\ell}_i \\ \bar{\ell}_i^\top & \tilde{\mathcal{L}}_i \end{bmatrix}, \; E^i_k = \begin{bmatrix} 0_{\hat{V}_i \times 1} \\ l_k \\ 0_{\tilde{V}_i \times 1} \end{bmatrix}, \; E^i = \begin{bmatrix} 0_{\hat{V}_i \times \tilde{V}_i} \\ 0_{\bar{V}_i \times \tilde{V}_i} \\ I_{\tilde{V}_i} \end{bmatrix},$$

$$C_i = \begin{bmatrix} \bar{C}_i & 0_{V_i \times \tilde{V}_i} & 0_{V_i \times \bar{V}_i} & 0_{V_i \times \tilde{V}_i} \\ 0_{\tilde{V}_i \times \bar{V}_i} & I_{\tilde{V}_i} & 0_{\bar{V}_i \times \bar{V}_i} & 0_{\bar{V}_i \times \tilde{V}_i} \\ 0_{V_i \times \bar{V}_i} & 0_{V_i \times \tilde{V}_i} & \bar{C}_i & 0_{V_i \times \tilde{V}_i} \\ 0_{\tilde{V}_i \times \bar{V}_i} & 0_{\tilde{V}_i \times \tilde{V}_i} & 0_{\tilde{V}_i \times \bar{V}_i} & I_{\tilde{V}_i} \end{bmatrix},$$

where $\bar{\ell}_i \in \mathbb{R}^{\bar{V}_i \times \tilde{V}_i}$, $l_k \in \mathbb{R}^{\bar{V}_i \times 1}$, and $\bar{C}_i \in \mathbb{R}^{|\mathcal{V}^1_i| \times \bar{V}_i}$ being a full row rank matrix where each of the rows have all zero entries except for one entry at the j-th position that corresponds to those nodes that are in $\mathcal{V}^1_i = \mathcal{N}_i \cup \{i\}$. Following a similar reasoning as in Theorem 6.3.4, one can verify that the second rank condition in (6.11) also holds. □

Such an UIO scheme can clearly be implemented for any subgraph $\hat{\mathcal{G}}_i$ containing the proximity graph $\mathcal{P}_i$. Applying Algorithm 6.3 or Algorithm 6.4 for the residuals obtained from these UIOs, with $\mathcal{G}$ replaced with $\hat{\mathcal{G}}_i$, addresses the first part of Problem 6.4.2. Hence, node i can detect and isolate a fault in node $j \in \mathcal{N}_i$ using only local models and measurements, as stated in the following result.

Theorem 6.4.4. *Consider a monitoring node i in a connected network satisfying Assumption 6.4.1 and a bank of UIO's calculated for the local subsystem* (6.19). *Using Algorithm 6.3 and the bank of observers, node i can detect and isolate a faulty node in its neighborhood.*

Proof. The proof follows from Lemma 6.4.3 and Theorem 6.4.1. □

6.5 Complexity Reduction of Distributed FDI

So far we have proposed the solutions to both Problems 6.4.1 and 6.4.2. In Section 6.4 we first showed that it is possible to detect the presence of a faulty node in the network distributedly, i.e., address Problem 6.4.1, at each node i, if i knows the exact model of its one-hop neighborhood and measuring the states of its neighbors. Then we introduced a method to address the first part of Problem 6.4.2 that not only eliminates the need to have an exact network model beyond a subgraph containing the proximity graph of a given node for that node to detect and isolate faults in its one-hop neighborhood, but it also reduces the size of the observers. However, such a result is derived under the assumption that the node has access to all the measurements of the states of its two-hop neighbors. In this section we show that the knowledge of the proximity graph is in fact the least amount of knowledge required to achieve distributed FDI when equal costs are associated with each necessary state measurement and network component that needs to be known, thus addressing the second part of Problem 6.4.2. Later, the complexity of the overall distributed FDI scheme is minimized by reducing the number of monitoring nodes while still ensuring that every node in the network is monitored.

6.5.1 Local Models and Additional Measurements

Suppose node i has the local model (6.19) for a given subgraph $\hat{\mathcal{G}}_i(\hat{\mathcal{V}}_i, \hat{\mathcal{E}}_i)$. Consider the case where equal costs are associated with each node ℓ in $\mathcal{B}(\hat{\mathcal{V}}_i)$, and with each of the edges that are known exactly, i.e., each $\{j, k\} \in \hat{\mathcal{E}}_i$. In other words, a cost is associated with any piece of information available to a node i; be it extra measurements or information about the existence of an edge between two nodes. This cost is minimized by solving the following optimization problem:

$$\underset{\mathcal{P}_i \subseteq \hat{\mathcal{G}}_i \subseteq \mathcal{G}}{\text{minimize}} \quad |\mathcal{S}_i(\hat{\mathcal{V}}_i)| + |\hat{\mathcal{E}}_i|. \tag{6.20}$$

We conclude this section by introducing the following result that shows that knowing $\mathcal{P}_i$ exactly is optimal, in the sense that it minimizes (6.20).

Theorem 6.5.1. *Consider a monitoring node i in an arbitrary connected network and a bank of UIO's calculated for the local subsystem $\hat{\mathcal{G}}_i$. Setting $\hat{\mathcal{G}}_i = \mathcal{P}_i$ simultaneously minimizes the number of state measurements $|\mathcal{S}_i|$ and the number of known network connections $|\hat{\mathcal{E}}_i|$ needed to design the bank of UIO's.*

Proof. Recall from Assumption 6.4.2 that $\mathcal{S}_i(\hat{\mathcal{V}}_i) = \{i\} \cup \mathcal{N}_i \cup \mathcal{B}(\hat{\mathcal{V}}_i)$. From Lemma 6.4.3 we know that any $\hat{\mathcal{G}}_i$ should be such that $\mathcal{P}_i \subseteq \hat{\mathcal{G}}_i$. To obtain a contradiction, assume that there is a $\mathcal{G}_i^\star(\mathcal{V}_i^\star, \mathcal{E}_i^\star)$ such that $\mathcal{P}_i$ is a strict subset of $\mathcal{G}_i^\star(\mathcal{V}_i^\star, \mathcal{E}_i^\star)$ that results in a smaller value for the objective function in (6.20). We can obtain it by adding vertices that are in $\mathcal{V}_i^\star \setminus \mathcal{V}_{P_i}$ one by one to $\mathcal{P}_i$. If we introduce a single vertex ℓ_1 to $\mathcal{P}_i$, then it is necessary that all the $\bar{\eta}$ edges $\{\ell_1, j\}$ such that $j \in \mathcal{V}_{\mathcal{P}_i}$ are exactly known, in addition to all the η edges incident to the vertices in $\mathcal{N}_i^2$. Call this new graph obtained from the addition of ℓ_1 and the aforementioned edges $\mathcal{G}_i^{+\ell_1}(\mathcal{V}_i^{+\ell_1}, \mathcal{E}_i^{+\ell_1})$. Then we have

$$\begin{aligned} |\mathcal{B}(\mathcal{V}_i^{+\ell_1})| + |\mathcal{E}_i^{+\ell_1}| &= |\mathcal{B}(\mathcal{V}_{P_i})| - \eta + 1 + |\mathcal{E}_{P_i}| + \eta + \bar{\eta} \\ &= |\mathcal{B}(\mathcal{V}_{P_i})| + 1 + |\mathcal{E}_{P_i}| + \bar{\eta}. \end{aligned}$$

Even for the case where there are no edges in the network connecting the nodes in $\mathcal{N}_i^2$, i.e., $\bar{\eta} = 0$, the cost function is increased by at least one. Repeating this argument for addition of any other vertex $\ell_j \in \mathcal{V}_i^\star \setminus \mathcal{V}_{P_i}$, one can deduce that the cost function does not decrease. Hence, there exists no $\mathcal{G}_i^\star$, such that $\mathcal{P}_i \nsubseteq \mathcal{G}_i^\star$, that minimizes the cost function given in (6.20). □

Theorem 6.5.1 provides the optimal subgraph $\hat{\mathcal{G}}_i$ that minimizes the amount of model knowledge and number of measurements where they are equally valued. However, if the cost of having measurements from a node is equal to $c_m \geq 0$ and the cost of knowing the existence of an edge is equal to $c_e \geq 0$, and $c_m \neq c_e$, (6.20) becomes

$$\underset{\mathcal{P}_i \subseteq \hat{\mathcal{G}}_i \subseteq \mathcal{G}}{\text{minimize}} \quad c_m|\mathcal{S}_i(\hat{\mathcal{V}}_i)| + c_e|\hat{\mathcal{E}}_i|. \tag{6.21}$$

One can construct simple examples with $c_m \neq c_e$ where taking $\hat{\mathcal{G}}_i = \mathcal{P}_i$ does not necessarily minimize the cost function proposed in (6.21).

6.5.2 Reducing the Number of Monitoring Nodes

It is not necessary for all the nodes in a network to monitor their neighbors and it is possible to decrease the number of monitoring nodes in the network while guaranteeing that each node in the network is being monitored by at least another node and calculating UIO's for only these nodes.

Assuming that each node monitors only its neighbors, we say that a FDI system in node i covers the set of nodes $\mathcal{N}_i$. Therefore, the objective is to select a minimum number of observer nodes that cover all the nodes in the network, i.e.,

$$\begin{aligned} \underset{S_o \subseteq \mathcal{V}}{\text{minimize}} \quad & |S_o| \\ \text{subject to} \quad & \bigcup_{i \in S_o} \mathcal{N}_i = \mathcal{V}, \end{aligned} \tag{6.22}$$

where S_o is the set of observer nodes.

As it can be seen, this is actually a set cover problem where we wish to determine a *minimum total dominating set*, *i.e.*, a set with minimum cardinality such that all nodes in the graph have at least one neighbor in that set. This is a well studied problem, having been classified as an NP-hard problem and two algorithms to approximately solve this problem can be found in Grandoni (2006).

Although the number of observers obtained by using $\mathcal{N}_i$ as the set of nodes covered by node i is not minimum, this method has one interesting property: all nodes in S_o are monitored by at least one neighbor. This means that even if an observer node is attacked, there is another observer node in the network that can detect it. Obviously, this decreases the vulnerability to faults in the monitoring nodes.

Other interesting properties may also be imposed by modifying the constraints in (6.22), such as having S_o to be connected, which is related to the *minimum connected dominating set* problem.

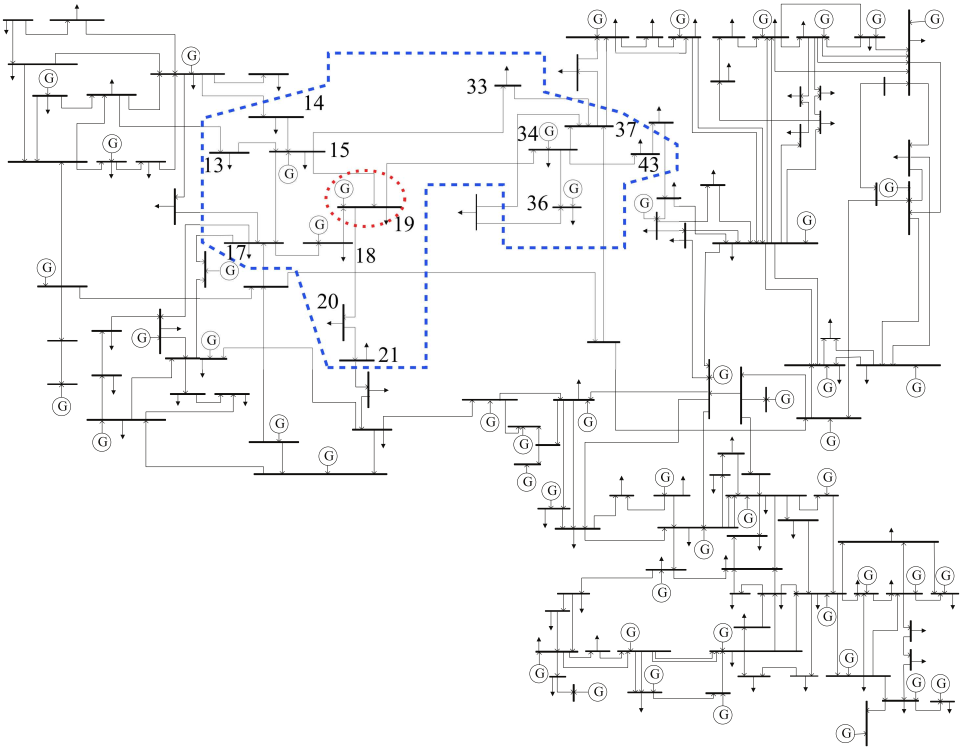


Figure 6.4: Diagram of the IEEE 118 bus power network. The monitoring node 19 is encircled with a red dotted line, while its 2-hop neighborhood is delimited by the blue dashed line.

Another way of minimizing the computational burden of the proposed method is to find a set of nodes that monitors all the nodes in the network with the minimum number of measurements, i.e., solving (6.22) with the cost function $|S_o|$ replaced with $\sum\limits_{i \in S_o} \deg(i)$. This problem can be solved first by finding all the dominating sets in the network and choosing the set that minimizes the cost function.

6.6 Numerical Examples

In this section we illustrate the solution proposed in the present chapter on a power network example. The simulations were carried out using the IEEE 118 bus network example available with the MATPOWER toolbox (Zimmerman *et al.*, 2009). A diagram of the power network is depicted in Figure 6.4.

We considered the classical linearized synchronous machine model (Kundur, 1994) for each node of the power network, leading to the global network dynamics

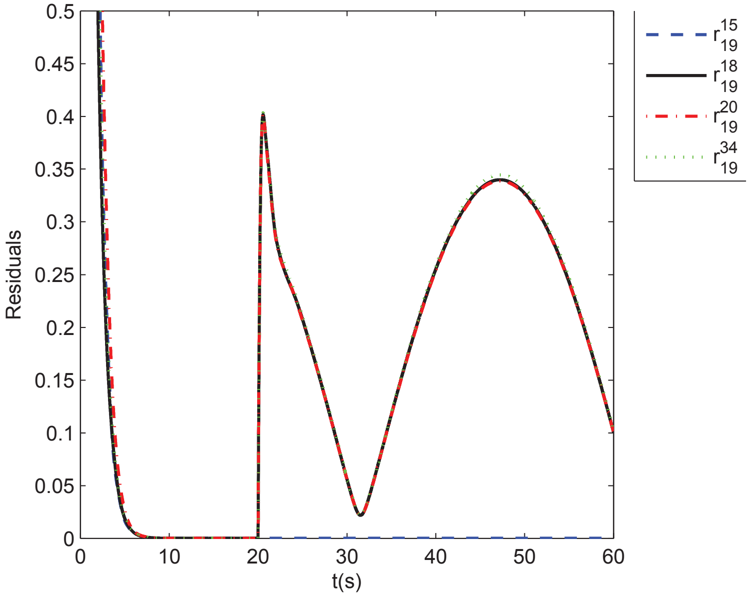


Figure 6.5: Residuals generated by the UIO bank at node 19 to detect node faults in $\mathcal{N}_{19}$. A sinusoidal fault is injected by node 15 after $t = 20\,s$. The fault in node 15 is successfully detected and isolated.

as in (6.1) with

$$A = \begin{bmatrix} 0_N & I_N \\ -\bar{M}\mathcal{L} & -\bar{M}\bar{D} \end{bmatrix}, \quad B = \begin{bmatrix} 0_N \ \bar{M} \end{bmatrix}^\top,$$

$$\bar{M} = \operatorname{diag}\left(\frac{1}{m_1}, \cdots, \frac{1}{m_N}\right), \quad \bar{D} = \operatorname{diag}\left(d_1, \cdots, d_N\right),$$

where $m_i > 0$ and $d_i > 0$ are the inertia and damping coefficients of node i, respectively, and $N = 118$ is the number of buses. Since these coefficients were not available in the example data files, they were randomly generated so that the load buses had considerably lower values than the generator buses, namely $m_g \approx 10^3 m_l$ and $d_g \approx 10^3 d_l$.

6.6.1 Faulty Node Detection using Local Models

In this example, node 19 is monitoring its neighbors for faulty behaviors using the method proposed in Section 6.4. Thus the network model knowledge needed is its 2−hop neighborhood, which consists of 26 states, as opposed to the 236 states of the

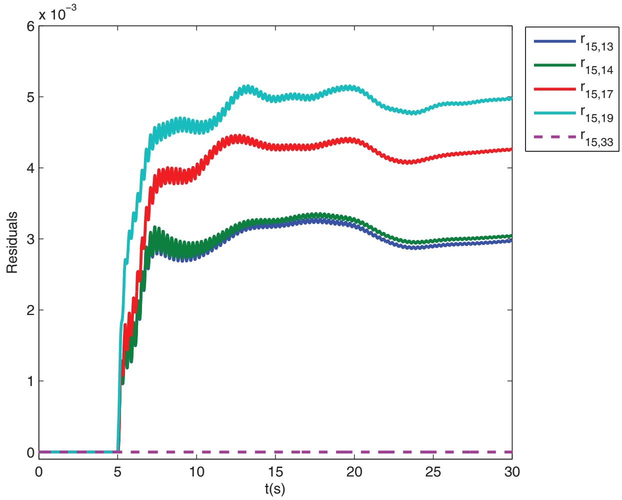


Figure 6.6: Residuals generated by the UIO bank at node 15 to detect edge faults. The edge between nodes 15 and 33 is removed at $t = 5\,s$. The edge fault is successfully detected and isolated.

global network. Using this smaller model, a bank of UIO's was generated according to Section 6.3.1 and Section 6.4.

In the simulations, node 15 exhibits a faulty behavior after $t = 20\,s$, which is successfully detected by node 19 as seen in Figure 6.5. Furthermore, all the residuals corresponding to other neighboring nodes become large while the one for node 15 remains at zero. Following Algorithm 6.3, node 15 is then detected and identified as the faulty node.

6.6.2 Faulty Edge Detection

Here we consider the case where node 15 monitors all its edges as proposed in Section 6.3.2. Note that in power networks the edges represent physical couplings and thus edge faults correspond to parameter faults described in Definition 6.2.3.(ii). We consider the scenario where the system is at equilibrium when the transmission line between nodes 15 and 33 is removed at $t = 5\,s$, which is modeled as $f^w_{15,33}(t) = -w_{15,33}$. This perturbation drives the system to another equilibrium point, enabling us to monitor the state trajectories and locate the faulty edge.

The residuals generated by the observers at node 15 are presented in Figure 6.6. As one can see, all the residuals diverge from zero except the one corresponding to the edge between nodes 15 and 33, hence the fault is successfully detected and isolated.

6.7 Summary

A distributed FDI scheme was proposed for detecting and isolating faults in nodes and edges of a networked multi-agent system. Additionally, the distributed FDI scheme designed using a given initial network model was shown to be resilient to the addition or removal of edges. Namely, fault detection can be achieved using this scheme by choosing suitable thresholds, provided that the proximity graph of the monitoring nodes remains constant. Later, we establish the minimum measurements required to be able to not only detect but also isolate the faulty nodes by each agent where the only model information they have is a local network model. Then, a solution to reduce the computational complexity of the distributed FDI scheme was proposed, where the solution lowers the number of monitoring nodes. Numerical result demonstrating the effectiveness of the proposed solutions were presented, taking the IEEE 118 bus power network as an example. As motivated by the example, the proposed methods can be fused to design a scalable and resilient distributed FDI architecture that achieves local fault detection and isolation despite unknown perturbations outside the local subsystem.

Chapter 7

Distributed Reconfiguration in Networked Control Systems

The proliferation of low cost embedded systems with radio capabilities has enabled the deployment of networked systems with increased performance and flexibility. However, these systems become increasingly complex and must be efficiently designed and operated. Several steps have been taken in this direction, in the development of resilient and fault tolerant architectures and technologies (Ding *et al.*, 2008; Blanke *et al.*, 2006) and plug-and-play control (Bendtsen *et al.*, 2013; Riverso *et al.*, 2013; Bodenburg and Lunze, 2013). In this chapter, we focus on distributed sensor and actuator reconfiguration in over-sensed and over-actuated networked control systems. In the event of malfunctioning actuators, sensors or other system components, control systems may exhibit poor performances or even become unstable (Blanke *et al.*, 2006; Poovendran *et al.*, 2012). Thus, the design of fault-tolerant control systems is of major importance. Examples of safety-critical systems that must be resilient to faults are power networks, aircrafts, nuclear power plants and chemical plants.

7.1 Contributions and Related Work

Since the 1970s, much research has been conducted in fault-tolerant control systems, fault detection and diagnosis (FDD) and reconfigurable control (Maciejowski, 1997; Blanke *et al.*, 2006; Lunze and Richter, 2008; Zhang and Jiang, 2008; Ding, 2008; Härkegård and Glad, 2005). FDD deals with the identification of faults, while reconfigurable control proposes methods to reconfigure a system after a fault has been detected. The objectives of reconfiguration are generally to recover stabilization of the system, maintaining the same state trajectory (also known as model-matching), achieving the same equilibrium point or minimizing the loss in performance inflicted by the fault. Model-matching reconfiguration in particular, has been the focus of much of the research in this area (Lunze and Richter, 2008). Many types of faults in sensors, actuators and other system components have been considered in both linear and nonlinear systems. However, the vast majority of the solutions rely on a centralized approach (Wu *et al.*, 2000; Staroswiecki *et al.*, 2007; Staroswiecki and Cazaurang, 2008; Staroswiecki and Berdjag, 2010; Richter *et al.*, 2011). Due to the increased complexity and size of current control systems, such techniques may be impractical (Åkerberg *et al.*, 2011; Poovendran *et al.*, 2012). Through the increased computation and communication capabilities of embedded devices in these systems, FDD can technically move from a centralized implementation to a distributed one. However, distributed FDD and reconfiguration to enable distributed fault tolerant systems has been much less explored. The architecture of such systems is discussed by Campelo *et al.* (1999); Jiang *et al.* (2007); Jin and Yang (2009), while Yang *et al.* (2010) employ a distributed FDD to perform a centralized reconfiguration. To the best of our knowledge, distributed reconfiguration has not yet been addressed in the literature. Application examples where distributed reconfiguration is beneficial are distributed control of wind-farms (Morrisse *et al.*, 2012), farming and livestock systems (Bendtsen *et al.*, 2013) and data-server cooling systems (Ellsworth *et al.*, 2008).

In this chapter, we address the problem of distributed reconfiguration for networked control systems with sensor and actuator faults and redundancies. Using the proposed scheme, healthy sensors and actuators are able to locally compensate for faults while disabling the faulty sensors and actuators. The proposed distributed method minimizes the steady-state estimation error covariance and a linear-quadratic control cost under faults while achieving model-matching: the desired closed-loop estimation error and dynamics remain the same with and without the fault. The distributed algorithm is shown to converge to the optimal solution asymptotically. Additionally, the stability of the closed-loop system is analyzed when the distributed reconfiguration algorithm terminates in finite-time.

The rest of this chapter is organized as follows. Section 7.2 presents the system architecture and formulates the problem. The centralized solution to the reconfiguration problem is presented in Section 7.3. In Section 7.4, it is shown that the reconfiguration can be distributed among the sensor or actuator nodes and an efficient algorithm is devised. Stability properties of the closed-loop system under the proposed distributed reconfiguration scheme are given in Section 7.5. Finally, numerical examples illustrate the distributed reconfiguration methods in Section 7.6 and Section 7.7 concludes this chapter.

7.1.1 Notation

For a vector x, $\|x\| = \|x\|_2$ denotes the Euclidean norm of x. Given a matrix A, $\|A\|_2 = \max_u \frac{\|Au\|}{\|u\|}$ denotes the induced 2-norm of A, while $\kappa(A) = \|A\|_2 \|A^\dagger\|_2$ denotes the condition number of A. Additionally, $\mathcal{A} \setminus \mathcal{B}$ denotes the set obtained by removing set $\mathcal{B}$ from set $\mathcal{A}$, for $\mathcal{B} \subseteq \mathcal{A}$. A network is represented by an undirected graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$ with vertex set $\mathcal{V}$ and edge set $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$. The edge $e_k = (i, j) \in \mathcal{E}$ indicates that nodes i and j can exchange information. Denote $\mathcal{N}_i = \{j \in \mathcal{V} : \; j \neq$

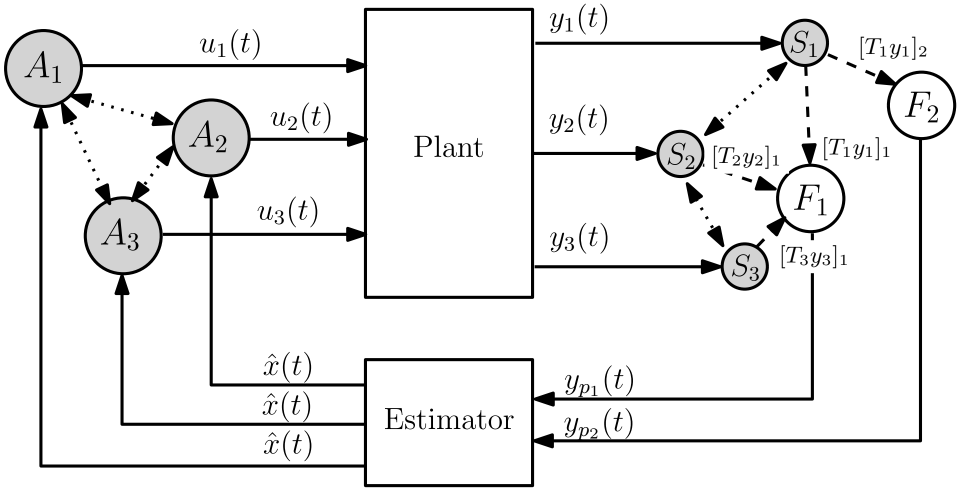


Figure 7.1: Networked control system with a network of sensors S_1, S_2 and S_3, aggregator nodes F_1 and F_2 and actuators A_1, A_2 and A_3. Sensors and actuators are responsible for reconfiguring themselves when system failures occur, through local information exchange in the network.

$i, (i, j) \in \mathcal{E}\}$ as the neighbor set of node i, where we assume that the network has no self-loops. Define $\mathcal{C}$ as the span of real symmetric matrices, $\mathcal{S}^n$, with sparsity pattern induced by the network, i.e., $\mathcal{C} = \{S \in \mathcal{S}^n : \; S_{ij} = 0 \,\text{if}\, i \neq j \,\text{and}\, (i, j) \notin \mathcal{E}\}$.

7.2 Problem Formulation

The architecture of the considered networked control system is depicted in Figure 7.1. This architecture has two networks, one of sensors and one of actuators. Each network has redundancy in components, which means that nominal operation can be maintained in spite of faulty components. The precise meaning of redundancy in our setup will be given later in this section. Each network is represented by an undirected graph. Each sensor or actuator is able to exchange information with its neighbors within the network. In typical applications such as building automation and industrial process control, a large number of sensors is expected to be deployed. To reduce the sensor-to-estimator communication, the information from the sensor nodes is fused at aggregator nodes, which connect to the estimator. The estimator is responsible for computing the state-estimate to be broadcasted to the actuators in the network which compute the control input values. The individual components of the system are described below.

7.2.1 System Model

Suppose the plant is modeled by a stochastic linear time-invariant differential equation,

$$\mathrm{d}x(t) = Ax(t)\,\mathrm{d}t + B\Gamma_u(t)u(t)\,\mathrm{d}t + \mathrm{d}w(t) \tag{7.1}$$

$$y(t)\,\mathrm{d}t = \Gamma_y(t)\left(Cx(t)\,\mathrm{d}t + \mathrm{d}v(t)\right), \tag{7.2}$$

with a state $x(t) \in \mathbb{R}^{n_x}$, $y(t) \in \mathbb{R}^{n_y}$ and $u(t) \in \mathbb{R}^{n_u}$ are the measurement vector and input vector, respectively, and $w(t) \in \mathbb{R}^{n_x}$ and $v(t) \in \mathbb{R}^{n_y}$ are independent Wiener processes with uncorrelated increments (Åström, 1970). The incremental covariances are $W\,\mathrm{d}t$ and $V\,\mathrm{d}t$, respectively.

Sensor and actuator faults are modelled by the diagonal matrices $\Gamma_y(t) \in \mathbb{R}^{n_y \times n_y}$ and $\Gamma_u(t) \in \mathbb{R}^{n_u \times n_u}$, respectively, with $[\Gamma_y(t)]_{ii} = \gamma_{y_i}(t) \in \{0,1\}$ and $[\Gamma_u(t)]_{ii} = \gamma_{u_i}(t) \in \{0,1\}$. Here, $\gamma_{y_i}(t)$ ($\gamma_{u_i}(t)$) represents the effectiveness of sensor (actuator) i at time t, where $\gamma_{y_i}(t) = 1$ ($\gamma_{u_i}(t) = 1$) means that the sensor (actuator) is functioning (healthy), while $\gamma_{y_i}(t) = 0$ ($\gamma_{u_i}(t) = 0$) indicates that the sensor (actuator) is faulty. The system is initially under nominal conditions, hence $\Gamma_y(t) = I$ and $\Gamma_u(t) = I$ for $t < 0$. All faults are assumed to occur simultaneously at time $t = 0$ and remain unchanged thereafter, which allows the time argument to be omitted. However, the methods devised in this work directly apply to the non-simultaneous fault case.

The sensor nodes apply a local linear transformation to the sensor measurements and transmit these values through the network to aggregation nodes which fuse the sensor data from several sensors. The fused signal is aggregated as

$$y_p(t)\,\mathrm{d}t = Ty(t)\,\mathrm{d}t = T\Gamma_y Cx(t)\,\mathrm{d}t + T\Gamma_y\,\mathrm{d}v(t), \tag{7.3}$$

where $T \in \mathbb{R}^{n_p \times n_y}$ is the aggregation matrix and $y_p(t) \in \mathbb{R}^{n_p}$ is transmitted to the estimator. It is assumed that the number of fused variables n_p is strictly smaller than the number of measurements n_y.

The sensor and actuator networks are represented by the connected and undirected graphs $\mathcal{G}_y(\mathcal{V}_y, \mathcal{E}_y)$ with $|\mathcal{V}_y| = n_y$ vertices and $\mathcal{G}_u(\mathcal{V}_u, \mathcal{E}_u)$ with $|\mathcal{V}_u| = n_u$ vertices, respectively. For simplicity of presentation, we assume that each aggregator node is connected to all sensor nodes. The set of sensor and actuator nodes is defined as $\mathcal{V} \triangleq \mathcal{V}_y \cup \mathcal{V}_u$, whereas we denote $\mathcal{V}^f \subseteq \mathcal{V}$ as the set of faulty nodes. Let the set of healthy nodes be $\mathcal{V}^h \triangleq \mathcal{V} \setminus \mathcal{V}^f$ with $\mathcal{E}^h = \{(i,j) \in \mathcal{E} |\, i,j \in \mathcal{V}^h\}$. The subgraphs $\mathcal{G}_y^h(\mathcal{V}_y^h, \mathcal{E}_y^h)$ and $\mathcal{G}_u^h(\mathcal{V}_u^h, \mathcal{E}_u^h)$ correspond to the graphs of the healthy sensor and actuator nodes, respectively, where $\mathcal{V}_y^h \triangleq \mathcal{V}^h \cap \mathcal{V}_y$, $\mathcal{V}_u^h \triangleq \mathcal{V}^h \cap \mathcal{V}_u$, $\mathcal{E}_y^h \triangleq \mathcal{E}^h \cap \mathcal{E}_y$, and $\mathcal{E}_u^h \triangleq \mathcal{E}^h \cap \mathcal{E}_u$.

We assume that the controller is given by the continuous-time linear-quadratic Gaussian (LQG) controller (Åström, 1970). Let the pair (TC, A) be observable and (A, B) be controllable. Next we describe the controller and estimator design under nominal conditions with $\Gamma_u = I$ and $\Gamma_y = I$. For LQG control, the feedback gain is obtained as the minimizer of the control cost criterion

$$J_c \triangleq \lim_{\mathcal{T}\to\infty} \frac{1}{\mathcal{T}} \int_0^{\mathcal{T}} \mathbb{E}\{x(t)^\top Qx(t) + u(t)^\top Ru(t)\}\ \mathrm{d}t$$

where $Q \succeq 0$ and $R \succ 0$ are weight matrices. We assume R is diagonal. The optimal LQ controller is given by

$$u(t) = -K\hat{x}(t) = -R^{-1}B^\top P\hat{x}(t) \tag{7.4}$$

where $\hat{x}(t)$ is the state estimate and P the solution to the Riccati equation

$$A^\top P + PA - PBR^{-1}B^\top P + Q = 0.$$

The state-estimate is computed by the Kalman-Bucy filter (Åström, 1970) as follows

$$\dot{\hat{x}}(t) = (A - LTC)\hat{x}(t) + Bu(t) + Ly_p(t), \tag{7.5}$$

with

$$L = \Sigma C^\top T^\top (TVT^\top)^{-1},$$

where $\Sigma = \lim_{t\to\infty} \mathbb{E}\{e(t)e(t)^\top\}$ is the steady-state covariance matrix of the estimation error $e(t) = \hat{x}(t) - x(t)$ given by the Riccati equation

$$A\Sigma + \Sigma A^\top - \Sigma C^\top T^\top (TVT^\top)^{-1} TC\Sigma + W = 0.$$

The Kalman-Bucy filter minimizes the expected mean-squared error, which we denote as the estimation cost function:

$$J_e \triangleq \lim_{\mathcal{T}\to\infty} \frac{1}{\mathcal{T}} \int_0^{\mathcal{T}} \mathbb{E}\{e(t)^\top e(t)\}\, \mathrm{d}t. \tag{7.6}$$

From now on, we drop the time argument (t) when it is clear from the context.

7.2.2 Reconfiguration Problem

Consider a scenario where faults have disabled several sensor and actuator nodes, yielding $\Gamma_u \neq I$ and $\Gamma_y \neq I$. A possible corrective action is to modify the aggregation matrix T and feedback matrix K so that only the remaining healthy sensors and actuators are used to guarantee a certain level of performance of the system. Let $\bar{u} \in \mathbb{R}^{n_u}$ and $\bar{y}_p \in \mathbb{R}^{n_p}$ denote the reconfigured control and sensor fusion signals after the fault. They are given by

$$\begin{aligned} \bar{y}_p\, \mathrm{d}t &= \bar{T}y\, \mathrm{d}t = \bar{T}\Gamma_y Cx\, \mathrm{d}t + \bar{T}\Gamma_y\, \mathrm{d}v, \\ \bar{u} &= -\bar{K}\hat{x}. \end{aligned}$$

Denote $\bar{A}_c(\bar{K}) = A - B\Gamma_u\bar{K}$ and $\bar{A}_e(\bar{T}) = A - L\bar{T}\Gamma_y C$ as the system matrices for the closed-loop dynamics of the system and estimator, respectively. The objective of the reconfiguration is to achieve model-matching (Staroswiecki and Cazaurang, 2008) for both the estimation dynamics and the closed-loop system dynamics by computing $\bar{T}$ and $\bar{K}$ after the fault occurs, respectively. Model-matching is a common reconfiguration goal, as it guarantees maintained system behavior in the presence of faults. The definition of model-matching reconfiguration is as follows. Let us denote the closed-loop estimator dynamics before the fault as $A_e = A - LTC$ and the nominal closed-loop system matrix as $A_c = A - BK$. Then, *model-matching on the estimation error dynamics* is achieved if $\bar{A}_e(\bar{T}) = A_e$ for some new aggregation matrix $\bar{T}$. Similarly, *model-matching on the closed-loop system dynamics* is achieved if $\bar{A}_c(\bar{K}) = A_c$ for some new feedback gain matrix $\bar{K}$.

Assumption 7.2.1. *The actuator and sensor networks have sufficient redundancy such that model-matching is feasible in case of faults, i.e.,*

$$\begin{aligned}\operatorname{Im}(BK) &\subseteq \operatorname{Im}(B\Gamma_u), \\ \operatorname{Im}\left(C^\top T^\top\right) &\subseteq \operatorname{Im}\left(C^\top \Gamma_y\right).\end{aligned}$$

As the model-matching constraints are under-determined, i.e., they admit multiple solutions, we propose to find the model-matching solutions that minimize certain quadratic costs. In particular, the cost function for sensor reconfiguration is the quadratic estimation cost (7.6) under the fault

$$J_e(\bar{T}) = \lim_{\tau\to\infty} \frac{1}{\tau} \int_0^\tau \mathbb{E}\{\bar{e}^\top \bar{e}\}\ \mathrm{d}t \tag{7.7}$$

where $\bar{e}$ is the estimation error after the fault occurred. Furthermore, we define the objective function of the actuator reconfiguration as the quadratic control cost for the reconfigured control input

$$\begin{aligned} J_c(\bar{K}) = \lim_{\tau\to\infty} \frac{1}{\tau} \quad & \int_0^\tau \mathbb{E}\left\{x^\top \left(Q + \bar{K}^\top \Gamma_u R \Gamma_u \bar{K}\right) x\right\}\ \mathrm{d}t \\ & \text{subject to} \quad \dot{x} = \left(A - B\Gamma_u \bar{K}\right) x, \end{aligned} \tag{7.8}$$

where the expectation is taken with respect to the initial condition $x(0)$, which is a zero-mean Gaussian random variable with the positive definite covariance matrix $R_0 \triangleq \mathbb{E}\{x(0)x(0)^\top\}$.

The sensor and actuator networked reconfiguration problem is to find the reconfigured aggregation matrix $\bar{T}$ and feedback gain matrix $\bar{K}$ which minimize the estimation (7.7) and control cost (7.8), respectively, subject to the model-matching condition.

The sensor reconfiguration can be re-formulated as

$$\begin{aligned} \underset{\bar{T}}{\text{minimize}} \quad & J_e(\bar{T}) \\ \text{subject to} \quad & A - L\bar{T}\Gamma_y C = A - LTC, \end{aligned} \tag{7.9}$$

while the actuator reconfiguration problem is

$$\begin{array}{ll} \underset{\bar{K}}{\text{minimize}} & J_c(\bar{K}) \\ \text{subject to} & A - B\Gamma_u\bar{K} = A - BK. \end{array} \tag{7.10}$$

The solution to the these optimization problems may be achieved in a centralized or distributed fashion. Next we describe a centralized approach to solve them, in which we assume that the reconfiguration takes place instantaneously. Later, we propose an efficient distributed solution based solely on local information exchange among sensor nodes and actuators nodes. In Section 7.5 we analyze the stability properties of the proposed distributed algorithm when the reconfiguration is not instantaneous.

7.3 Centralized Sensor and Actuator Reconfiguration

We now tackle the centralized sensor and actuator reconfiguration problems. Their solutions are derived and the cnetralized reconfiguration mechanisms are illustrated.

7.3.1 Centralized Sensor Reconfiguration

The optimal solution to (7.9) can be characterized as follows.

Proposition 7.3.1. *The solution to the optimization problem* (7.9) *is*

$$\bar{T}^\star = TC(C^\top V^{-1}\Gamma_y C)^\dagger C^\top \Gamma_y V^{-1}. \tag{7.11}$$

In order to prove Proposition 7.3.1, we use the following lemma.

Lemma 7.3.2. *Optimization problem* (7.9) *is equivalent to*

$$\begin{array}{ll} \underset{\bar{T}}{\textit{minimize}} & \operatorname{tr}\left((W + L\bar{T}\Gamma_y V\Gamma_y \bar{T}^\top L^\top)Z_e\right) \\ \textit{subject to} & LTC = L\bar{T}\Gamma_y C \\ & 0 = A_e^\top Z_e + Z_e A_e + I. \end{array} \tag{7.12}$$

Proof. The first constraint in (7.12) is the model-matching constraint, which is derived as follows. In Section 7.2.2, model-matching is guaranteed if the closed-loop matrix before fault is the same as after the fault, i.e.,

$$A - L\bar{T}\Gamma_y C = A - LTC = A_e.$$

Moreover, the objective function and last constraint follow are given as follows. The objective function J_e in (7.7) is equivalent to $J_e = \operatorname{tr}(\bar{\Sigma})$, where $\bar{\Sigma}$ is steady-state covariance of the estimation error after a fault and defined as $\bar{\Sigma} = \lim_{t\to\infty} \mathbb{E}\{\bar{e}(t)\bar{e}(t)^\top\}$.

Additionally, under any given estimator gain L, $\bar{\Sigma}$ is given by the following Lyapunov equation (see Åström (1970) for details),

$$(A - LTC)\bar{\Sigma} + \bar{\Sigma}(A - LTC)^\top + W + L\bar{T}\Gamma_y V \Gamma_y{}^\top \bar{T}^\top L^\top = 0.$$

The solution of the above Lyapunov equation, can also be expressed as

$$\bar{\Sigma} = \int_0^\infty e^{A_e t}(W + L\bar{T}\Gamma_y V \Gamma_y \bar{T}^\top L^\top) e^{A_e^\top t}\,\mathrm{d}t.$$

By noticing that the term $W + L\bar{T}\Gamma_y V \Gamma_y \bar{T}^\top L^\top$ is independent of time, one can arrive to the following equivalence of the cost

$$J_e = \mathrm{tr}(\bar{\Sigma}) = \mathrm{tr}\left((W + L\bar{T}\Gamma_y V \Gamma_y \bar{T}^\top L^\top)\int_0^\infty e^{A_e^\top t} e^{A_e t}\,\mathrm{d}t\right).$$

By denoting $Z_e = \int_0^\infty e^{A_e^\top t} e^{A_e t}\,\mathrm{d}t$ and noticing that Z_e is the solution to the Lyapunov equation $A_e^\top Z_e + Z_e A_e + I = 0$, the proof is concluded. □

We now derive the optimal solution to (7.12), which is also the solution to the sensor reconfiguration problem (7.9).

Proof of Proposition 7.3.1. Consider the optimization problem (7.12), which is convex. Note that the second equality constraint is a Lyapunov equation with the Hurwitz system matrix A_e, determined by the model-matching condition. Hence, the variable Z_e is uniquely defined by the constraint and can be computed before hand. The Lagrangian function for (7.12) is

$$\mathcal{L}(\bar{T}, \Lambda) = \mathrm{tr}\left((W + L\bar{T}\Gamma_y V \Gamma_y \bar{T}^\top L^\top) Z_e\right) + \mathrm{tr}\left(\Lambda^\top (LTC - L\bar{T}\Gamma_y C)\right),$$

where $\Lambda \in \mathbb{R}^{n_x \times n_x}$ represents the Lagrange multipliers. Using the trace derivative expressions, the Karush-Kuhn-Tucker (KKT) optimality conditions can be written as

$$\begin{aligned} 0 &= \frac{\partial}{\partial \bar{T}}\mathcal{L}(\bar{T}, \Lambda) = 2L^\top Z_e L\bar{T}\Gamma_y V \Gamma_y - L^\top \Lambda C^\top \Gamma_y \\ 0 &= LTC - L\bar{T}\Gamma_y C \end{aligned}$$

and can be rewritten as

$$\begin{aligned} 0 &= \bar{T}\Gamma_y - \frac{1}{2}(L^\top Z_e L)^\dagger L^\top \Lambda C^\top V^{-1}\Gamma_y \\ 0 &= LTC(C^\top V^{-1}\Gamma_y C)^\dagger - \frac{1}{2}L(L^\top Z_e L)^\dagger L^\top \Lambda. \end{aligned}$$

Solving the above equations yields the optimal solution (7.11). □

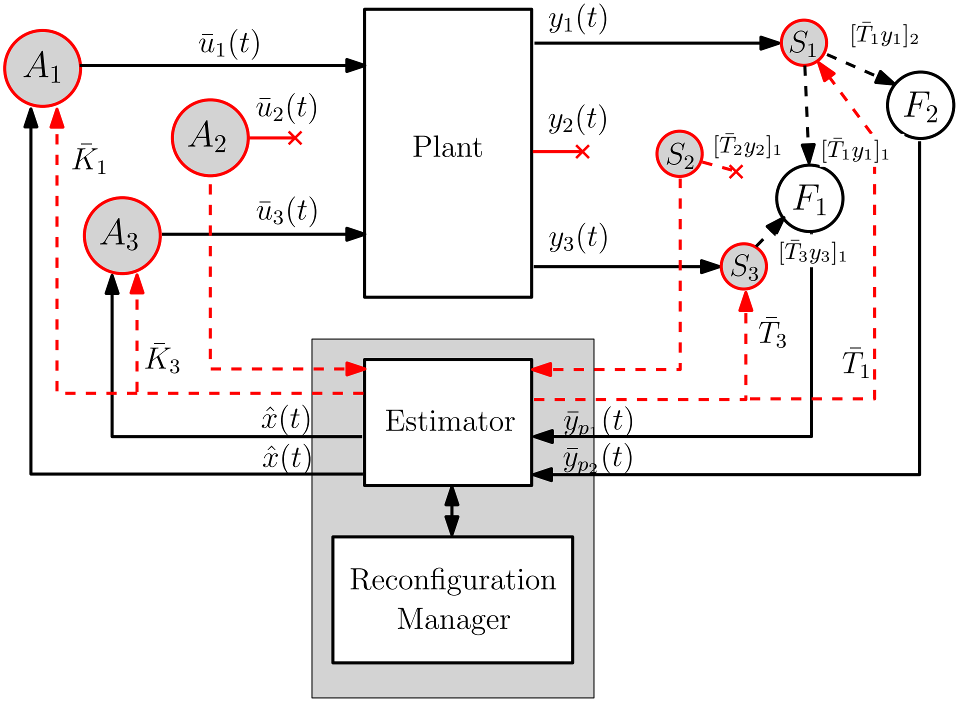


Figure 7.2: Networked control system with centralized sensor and actuator reconfiguration. Faults are reported by the sensors and actuators to the centralized estimator. Red dashed arrows represent the transmission of information related to faults. In the sensor reconfiguration case, the fault information may also be first transmitted from the centralized unit to the aggregators, and then from the aggregators to the sensors.

Figure 7.2 illustrates the centralized reconfiguration that is performed by a system component denoted as reconfiguration manager. A fault occurs at sensor S_2, which detects that it is faulty, reporting it to the reconfiguration manager which now knows Γ_y. The reconfiguration manager solves (7.11) to derive the new aggregation matrix $\bar{T} = [\bar{T}_1 \dots \bar{T}_{n_y}]$, where $\bar{T}_i$ is a column vector corresponding to the i-th column of $\bar{T}$. Then, $\bar{T}_1$ is sent to sensor S_1 and $\bar{T}_3$ to sensor S_3, which compute $\bar{T}_1 y_1$ and $\bar{T}_3 y_3$, where $\bar{T}_i y_i = [[\bar{T}_i y_i]_1 \dots [\bar{T}_i y_i]_{n_p}]^\top$. Each non-zero component $[\bar{T}_i y_i]_j$ is sent to the j-th aggregator, allowing each aggregator node to compute y_{p_j} and transmit this value to the estimator.

7.3.2 Centralized Actuator Reconfiguration

The optimal centralized actuator reconfiguration is now presented.

Proposition 7.3.3. *The solution to the optimization problem* (7.10) *is*

$$\bar{K}^{\star} = \Gamma_u R^{-1} B^{\top} (B\Gamma_u R^{-1} B^{\top})^{\dagger} BK. \tag{7.13}$$

To prove the above result, we use the following lemma.

Lemma 7.3.4. *The optimization problem* (7.10) *is equivalent to*

$$\begin{array}{ll} \underset{\bar{K}}{\textit{minimize}} & \operatorname{tr}\big((Q + \bar{K}^{\top}\Gamma_u R \Gamma_u \bar{K}) Z_c\big) \\ \textit{subject to} & BK = B\Gamma_u \bar{K} \\ & 0 = A_c Z_c + Z_c A_c^{\top} + R_0. \end{array} \tag{7.14}$$

Proof. For a given controller $\bar{K}$ satisfying the model-matching constraint $B\Gamma_u\bar{K} = BK$, the objective function $J_c(\bar{K})$ in (7.8) is given by

$$\begin{aligned} J_c(\bar{K}) \quad = \quad & \int_0^{\infty} \mathbb{E}\left\{x(t)^{\top}(Q + \bar{K}^{\top}\Gamma_u R \Gamma_u \bar{K}) x(t)\right\} \mathrm{d}t \\ & \text{subject to} \quad \dot{x}(t) = A_c x(t). \end{aligned}$$

Solving the differential equation, the cost $J_c(\bar{K})$ can be rewritten as

$$\begin{aligned} J_c(\bar{K}) &= \int_0^{\infty} \mathbb{E}\left\{ x(0)^{\top} e^{A_c^{\top} t} \left(Q + \bar{K}^{\top}\Gamma_u R \Gamma_u \bar{K}\right) e^{A_c t} x(0) \right\} \mathrm{d}t \\ &= \operatorname{tr}\left\{ \int_0^{\infty} \mathbb{E}\left\{ e^{A_c t} x(0) x(0)^{\top} e^{A_c^{\top} t} \left(Q + \bar{K}^{\top}\Gamma_u R \Gamma_u \bar{K}\right) \right\} \mathrm{d}t \right\} \\ &= \operatorname{tr}\left\{ \int_0^{\infty} e^{A_c t} R_0 e^{A_c^{\top} t} \, \mathrm{d}t \left(Q + \bar{K}^{\top}\Gamma_u R \Gamma_u \bar{K}\right) \right\} \\ &= \operatorname{tr}\left\{ Z_c \left(Q + \bar{K}^{\top} R \bar{K}\right) \right\}, \end{aligned}$$

where $Z_c \succ 0$ is the unique solution to the Lyapunov equation

$$(A - BK)Z_c + Z_c(A - BK)^{\top} + R_0 = 0,$$

thus concluding the proof. □

Proof of Proposition 7.3.3. Consider the optimization problem (7.14). Similar to the proof of Proposition 7.3.1, the variable Z_c is the unique solution to the Lyapunov equation given by the second equality constraint. The Lagrangian function for (7.14) is $\mathcal{L}(\bar{K}, \Lambda) = \operatorname{tr}\big((Q + \bar{K}^{\top}\Gamma_u R \Gamma_u \bar{K}) Z_c\big) + \operatorname{tr}\big(\Lambda^{\top}(BK - B\Gamma_u \bar{K})\big)$, where $\Lambda \in \mathbb{R}^{n_x \times n_x}$ represents the Lagrange multipliers. Moreover, the KKT optimality conditions are

$$\begin{aligned} 0 &= \frac{\partial}{\partial \bar{K}} \mathcal{L}(\bar{K}, \Lambda) = 2\Gamma_u R \Gamma_u \bar{K} Z_c - \Gamma_u B^{\top} \Lambda \\ 0 &= BK - B\Gamma_u \bar{K} \end{aligned}$$

and can be rewritten as

$$\begin{aligned} 0 &= \Gamma_u \bar{K} - \frac{1}{2}\Gamma_u R^{-1} B^\top \Lambda Z_c^{-1} \\ 0 &= (B\Gamma_u R^{-1} B^\top)^\dagger BK - \frac{1}{2}\Lambda Z_c^{-1}. \end{aligned}$$

Solving the above equations yields (7.13). □

Figure 7.2 depicts also a fault in the actuator network. A fault occurs at actuator A_2, which reports to the reconfiguration manager. The reconfiguration manager then solves (7.14) to derive the new controller $\bar{K} = [\bar{K}_1^\top \dots \bar{K}_{n_u}^\top]^\top$, where $\bar{K}_i$ is a row vector corresponding to the i-th row of $\bar{K}$. Then, $\bar{K}_1$ is transmitted to to actuator A_1 and $\bar{K}_3$ to actuator A_3, which allows them to compute and apply $\bar{u}_1$ and $\bar{u}_3$, respectively.

We highlight that the centralized actuator reconfiguration solution may be also obtained through other problem formulations. Härkegård and Glad (2005) proposed to solve actuator redundancy through control allocation, which was formulated as an optimization problem using the concept of virtual actuators. By appropriately choosing the objective function, the solution (7.13) can be obtained. Moreover, the same result may be obtained using the pseudo-inverse method (Gao and Antsaklis, 1991; Staroswiecki, 2005) if the matrix R has identical elements. Otherwise, a modification of the method, to take R into account, is required.

7.4 Distributed Sensor and Actuator Reconfiguration

In this section, we propose a distributed algorithm to solve the reconfiguration problem. We begin by rewriting the centralized sensor and actuator reconfiguration problems in Lemmas 7.3.2 and 7.3.4 as quadratic optimization problems with a separable cost function and a global equality constraint.

Lemma 7.4.1. *Consider a set of l vectors $\eta_i \in \mathbb{R}^r$, for $i = 1, \dots, l$, and let $S \in \mathbb{R}^{l\times l}$ be a diagonal matrix with non-negative entries. The sensor and actuator reconfiguration problems* (7.12) *and* (7.14) *can be rewritten in the following form:*

$$\begin{aligned} \underset{\eta_1,\dots,\eta_l \in \mathbb{R}^r}{\mathit{minimize}} \quad & \sum_{i=1}^{l} S_{ii} \|\eta_i\|^2 \\ \mathit{subject\ to} \quad & \sum_{i=1}^{l} H_i \eta_i = \omega \end{aligned} \tag{7.15}$$

where $H_i \in \mathbb{R}^{n_x^2 \times r}$ and $\omega \in \mathbb{R}^{n_x^2}$. Let $H = \begin{bmatrix} H_1 & \dots & H_l \end{bmatrix}$ and $\eta = \begin{bmatrix} \eta_1^\top & \dots & \eta_l^\top \end{bmatrix}^\top$.

For the sensor case, we have $\bar{T} = \begin{bmatrix} \eta_1 & \dots & \eta_{n_y} \end{bmatrix}$, $H = (C^\top \Gamma_y^\top) \otimes L$, $\omega = vec(LTC)$, and $S_{ii} = [\Gamma_y]_{ii} V_{ii}$.

The actuator case is retrieved with $\bar{K} = \begin{bmatrix}\eta_1 & \dots & \eta_{n_u}\end{bmatrix}^\top$, $H = (I \otimes B\Gamma_u)\, P_r^{-1}$ *with* $P_r \in \mathbb{R}^{n_u n_x \times n_u n_x}$ *being a permutation matrix such that* $vec\big(\bar{K}\big) = P_r^{-1}\eta$, $\omega = vec\,(BK)$ *and* $S_{ii} = [\Gamma_u]_{ii} R_{ii}$.

In order to prove Lemma 7.4.1, we rewrite the sensor and actuator reconfiguration problems (7.12) and (7.14) as quadratic optimization problems with equality constraints.

Distributed sensor reconfiguration

For the sensor reconfiguration problem, we have the following result.

Lemma 7.4.2. *Define* $\bar{T} = \big[\eta_1 \cdots \eta_{n_y}\big]$, $\eta_i \in \mathbb{R}^{n_p}$ *and let* ${H^e}_i \in \mathbb{R}^{{n_x}^2 \times n_p}$, *for* $i = 1, \dots, n_y$. *The optimization problem* (7.12) *can be rewritten as*

$$\begin{aligned} \underset{\eta_1,\dots,\eta_{n_y}}{\mathit{minimize}} \quad & \sum_{i=1}^{n_y} [\Gamma_y]_{ii} V_{ii} \|\eta_i\|^2 \\ \mathit{subject\ to} \quad & \sum_{i=1}^{n_y} {H^e}_i \eta_i = \omega^e, \end{aligned}$$

where $H^e \triangleq \big[{H^e}_1 \dots {H^e}_{n_y}\big] = \Big((C^\top \Gamma_y^\top) \otimes L\Big)$ *and* $\omega^e = vec\,(LTC)$.

Proof. Recall that the cost J_e in (7.6) is given by

$$J_e = \operatorname{tr}(\bar{\Sigma}) = \operatorname{tr}((W + L\tilde{T}\Gamma_y V \Gamma_y T^\top L^\top) Z_e),$$

as derived in (7.12). As shown in Proposition 7.3.1, the optimal solution is independent of the constant terms W and $L^\top Z_e L$, which can be replaced with 0 and I, respectively. Since V and Γ are diagonal, one can write the new objective function as

$$\operatorname{tr}(\bar{T}\Gamma_y V \Gamma_y \bar{T}^\top) = \operatorname{tr}(\sum_{i=1}^{n_y} [\Gamma_y]_{ii} V_{ii} \eta_i \eta_i^\top) = \sum_{i=1}^{n_y} [\Gamma_y]_{ii} V_{ii} \|\eta_i\|^2.$$

The model-matching constraint follows directly by applying the vectorization operation. □

Distributed actuator reconfiguration

We now rewrite the centralized actuator reconfiguration problem from Lemma 7.3.4 as a quadratic problem with an equality constraint.

Lemma 7.4.3. *Define* $\bar{K} = \begin{bmatrix}\eta_1 & \dots & \eta_{n_u}\end{bmatrix}^\top$, *the column vector* $\eta_i \in \mathbb{R}^{n_x}$, P_r *as the permutation matrix for which* $P_r vec\big(\bar{K}\big) = \big[\eta_1^\top \dots \eta_{n_u}^\top\big]^\top$, *and let* ${H^c}_i \in \mathbb{R}^{n_x^2 \times n_x}$

for $i = 1, \ldots, n_u$. The optimization problem (7.14) *can be rewritten as*

$$\begin{aligned} \underset{\eta_1,\ldots,\eta_{n_u}}{\textit{minimize}} \quad & \sum_{i=1}^{n_u} [\Gamma_u]_{ii} R_{ii} \|\eta_i\|^2 \\ \textit{subject to} \quad & \sum_{i=1}^{n_u} H^c{}_i \eta_i = \omega^c \end{aligned}$$

where $H^c \triangleq [H^c{}_1 \ldots H^c{}_{n_u}] = (I \otimes B\Gamma_u)\, P_r^{-1}$ and $\omega^c = vec(BK)$.

Proof. The proof follows the derivations from Lemma 7.4.2 and is therefore omitted. □

Proof of Lemma 7.4.1. The proof follows directly from Lemmas 7.4.2 and 7.4.3. □

The variables $\eta_i \in \mathbb{R}^r$ and $q_i \in \mathbb{R}^{{n_x}^2}$ have the following interpretation. For the case of sensor reconfiguration, each η_i represents the aggregation matrix $\bar{T}$ components for the i-th sensor (i-th column of $\bar{T}$), i.e., how sensor i transforms its information to be transmitted to each of the fusion nodes that it is connected to. In the same manner, each $\eta_i^\top$ corresponds to the i-th actuator state-feedback matrix $\bar{K}$ components, i.e., the i-th row of $\bar{K}$. The value of ω corresponds to the vectorization of the closed-loop estimator dynamics and closed-loop system dynamics before a fault occurs, for the case of sensor and actuator reconfiguration, respectively. This represents the quantity that ideally must be maintained by the combination of all sensor (actuator) nodes during the reconfiguration, which refers to the model-matching constraint.

The optimization problem (7.15) may be solved distributedly using dual decomposition and iterative algorithms (Everett III, 1963; Shor *et al.*, 1985; Johansson, 2008). A requirement is that the network remains connected when faults occur. Using dual decomposition algorithms, the optimal solution to problem (7.15) is guaranteed to be achieved asymptotically in the number of iterations (Boyd *et al.*, 2011). The main drawback is that the global equality constraint of the problem is only ensured asymptotically. Therefore, model-matching is not guaranteed at every iteration. Due to this fact, we later analyze the stability of the system under the distributed reconfiguration in Section 7.5.

To solve the dual optimization problem of (7.15) we resort to the distributed alternating direction method of multipliers (ADMM) algorithm (Boyd *et al.*, 2011). In the following, the decision variable η at each iteration $k \geq 0$ is denoted as $\eta[k]$.

Theorem 7.4.4. *Define $q_1, \ldots, q_l \in \mathbb{R}^{n_x^2}$ such that $\sum_{i=1}^{l} q_i = \omega$ and local variables $\zeta_1, \ldots, \zeta_l \in \mathbb{R}^{n_x^2}$. Let*

$$\eta_i[k] = \frac{1}{2} S_{ii}^{-1} H_i^\top \zeta_i[k]$$

where $\zeta_i[k]$ is computed by the following algorithm:

$$
\begin{aligned}
\zeta_i[k+1] &= \Big(\frac{1}{2}H_i S_{ii}^{-1} H_i^\top + \rho|\mathcal{N}_i|I\Big)^{-1}\Big(q_i - \rho\sum_{j\in\mathcal{N}_i}\mu_{i,(i,j)}[k] - \pi_{(i,j)}[k]\Big)\\
\xi_{i,(i,j)}[k+1] &= \alpha\zeta_i[k+1] + (1-\alpha)\pi_{(i,j)}[k],\\
\xi_{j,(i,j)}[k+1] &= \alpha\zeta_j[k+1] + (1-\alpha)\pi_{(i,j)}[k],\\
\pi_{(i,j)}[k+1] &= \frac{1}{2}\Big(\xi_{i,(i,j)}[k+1] + \mu_{i,(i,j)}[k] + \xi_{j,(i,j)}[k+1] + \mu_{j,(i,j)}[k]\Big),\\
\mu_{i,(i,j)}[k+1] &= \mu_{i,(i,j)}[k] + \xi_{i,(i,j)}[k+1] - \pi_{(i,j)}[k+1],
\end{aligned}
\tag{7.16}
$$

where $\rho > 0$ is the step size, $\alpha \in (0, 2)$ is a relaxation parameter, $\rho\mu_{i,(i,j)}$ is the Lagrange multiplier of node i associated with the constraint $\zeta_i = \pi_{(i,j)}$, and $\xi_{i,(i,j)}$ is an auxiliary variable private to node i associated with the edge (i, j). Then, $\eta[k]$ converges to the solution of (7.15).

Note that the algorithm in Theorem 7.4.4 is distributed since it only requires communication between neighbors to exchange local values.

To prove Theorem 7.4.4, we first derive the dual form of (7.15).

Lemma 7.4.5. *Let $f_i(\eta_i) = \eta_i^\top S_{ii}\eta_i$. The optimization problem* (7.15) *can be rewritten in the following dual form:*

$$
\begin{aligned}
\underset{\{\zeta_i\},\,\{\pi_{(i,j)}\}}{\textit{minimize}} \quad & \sum_{i=1}^{l}\Big(\frac{1}{4}\zeta_i^\top H_i S_{ii}^{-1} H_i^\top \zeta_i - q_i^\top\zeta_i\Big)\\
\textit{subject to} \quad & \zeta_i = \pi_{(i,j)}, \quad \forall i \in \mathcal{V},\ j \in \mathcal{N}_i.
\end{aligned}
\tag{7.17}
$$

Proof. Consider the optimization problem (7.15). Using the Lagrange multiplier ζ associated with the equality constraint, the optimal solution may be computed by solving

$$
\underset{\zeta}{\text{maxmize}}\ \underset{\eta_1,\,\dots,\,\eta_l}{\text{minimize}}\ \sum_{i=1}^{l}\Big(f_i(\eta_i) - \zeta^\top H_i\eta_i\Big) + \zeta^\top\omega.
$$

Introducing the local variables $\zeta_1, \dots, \zeta_l$ and $q_1, \dots, q_l$ satisfying $\sum_{i=1}^{l} q_i = \omega$ and imposing the constraint $\zeta_i = \zeta_j$ for all $i \neq j$ yields

$$
\begin{aligned}
\underset{\zeta_1,\,\dots,\,\zeta_l}{\text{maximize}} \quad & \sum_{i=1}^{l}\underset{\eta_i}{\text{minimize}}\ \Big(f_i(\eta_i) - \zeta_i^\top H_i\eta_i\Big) + \zeta_i^\top q_i\\
\text{subject to} \quad & \zeta_i = \zeta_j, \quad \forall i, j,\ i \neq j.
\end{aligned}
$$

Each inner optimization problem with respect to η_i can be rewritten in terms of the convex conjugate function, i.e., $-f_i^\star(\zeta) = \min_{\eta_i} f_i(\eta_i) - \zeta^\top\eta$. Introducing the convex conjugate function in the objective function results in the following

$$\begin{aligned} \underset{\{\zeta_i\},\{\pi_{(i,j)}\}}{\text{minimize}} \quad & \sum_{i=1}^{l} \Big(f_i^\star(H_i^\top \zeta_i) - q_i^\top \zeta_i \Big) \\ \text{subject to} \quad & \zeta_i = \pi_{(i,j)}, \quad \forall i \in \mathcal{V},\ j \in \mathcal{N}_i, \end{aligned}$$

one can compute the value of the convex conjugate function as follows $f_i^\star(H_i^\top \zeta_i) = \frac{1}{4}\zeta_i^\top H_i S_i^{-1} H_i^\top \zeta_i$. Substituting this value in the objective function of the dual problem, gives (7.17) in Lemma 7.4.5. □

Proof of Theorem 7.4.4. The value of $\eta[k]$ is obtained as $\eta[k] = \text{argmin}_{x_i} f_i(x_i) - \zeta^\top H_i x_i = \frac{1}{2} S_{ii}^{-1} H_i^\top \zeta_i[k]$. The ADMM algorithm (7.16) follows from Boyd *et al.* (2011) and is thus omitted. □

The variables $q_i \in \mathbb{R}^{{n_x}^2}$ and $\zeta_i \in \mathbb{R}^{{n_x}^2}$ have the following interpretation. Vector q_i describes how the vectorization of the closed-loop dynamics, i.e. ω, is assigned among all nodes in the network. Note that the assignment is only constrained by the condition $\sum_{i=1}^{l} q_i = \omega$, thus admitting several solutions. For instance, one could have the closed-loop dynamics available only to one node, by having $q_1 = \omega$ and $q_j = 0$ for all $j \neq 1$. Variable ζ_i, only available at node i, is a local copy of the Lagrange multiplier associated with the model-matching constraint $H\eta = \omega$.

The following result indicates how the parameters q_i can be updated locally by the healthy nodes after a fault has occurred.

Lemma 7.4.6. *Let $j \in \mathcal{V}^f$ be an arbitrary faulty node, denote $\mathcal{J} \subseteq \mathcal{N}_j \cap \mathcal{V}_h$ as a subset of its healthy neighbors and assume $\mathcal{J}$ is not empty. Given the set $\{\bar{q}_i\}_{i\in\mathcal{V}}$ such that $\sum_{i\in\mathcal{V}} \bar{q}_i = \omega$, the set $\{q_i\}_{i\in\mathcal{V}}$ satisfying $\sum_{i\in\mathcal{V}_h} q_i = \omega$ can be computed as*

$$q_i = \begin{cases} \bar{q}_i, & i \notin \mathcal{J} \\ \bar{q}_i + \nu_i \bar{q}_j, & i \in \mathcal{J} \end{cases},$$

where $\nu_i \geq 0$ for all $i \in \mathcal{J}$ and $\sum_{i\in\mathcal{J}} \nu_i = 1$.

Proof. The computations are performed locally, since by construction only the neighbors of the faulty node j are involved in the computations. The coefficient ν_i indicates how much i compensates for the contribution of the faulty node j before the fault. Moreover, having $\bar{q}_i + \nu_i \bar{q}_j,\ i \in \mathcal{J}$ and $\sum_{i\in\mathcal{J}} \nu_i = 1$ ensures that $\sum_{i\in\mathcal{V}} \bar{q}_i = \omega$. Hence, each healthy node i in the neighborhood of the faulty node must solely exchange and agree on the set of parameters ν_i. □

In the above scheme, since the sensor and actuator networks are disjoint, the update of q_i for a sensor (actuator) fault is performed within the sensor (actuator) network.

The distributed reconfiguration algorithm can be summarized in Algorithm 7.5. An illustration of the distributed sensor and actuator reconfiguration is shown in

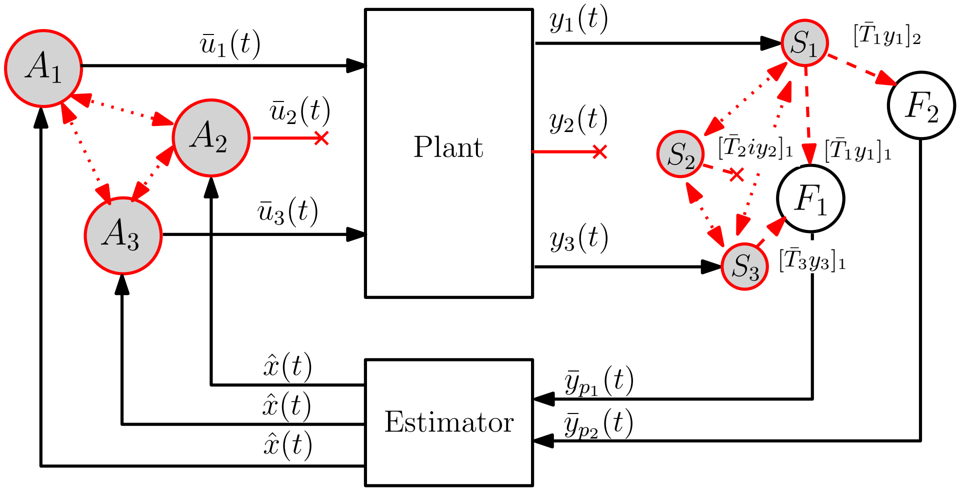


Figure 7.3: Networked control system with distributed sensor and actuator reconfiguration. Faults are detected by the sensors and actuators which are responsible for the reconfiguration. Reconfiguration is achieved through the communication among sensors and among actuators in a distributed manner through the sensor and actuator network, respectively.

Figure 7.3 where a fault occurs at sensor S_2 and actuator A_2 at $t = 0$. The sensors locally infer that sensor S_2 is no longer functioning, so sensors S_1 and S_3 reconfigure themselves. This is performed locally by each sensor computing the value of $\bar{T}_1$ and $\bar{T}_3$, and calculating $\bar{T}_1 y_1$ and $\bar{T}_3 y_3$. Each component $[\bar{T}_i y_i]_j$ is sent to the j-th aggregator, allowing each aggregator node to compute z_j and transmit this value to the controller node. Similarly, the actuators locally infer that actuator A_2 is faulty, so actuators A_1 and A_3 reconfigure themselves. This is a local operation where each actuator computes the value of $\bar{K}_1$ and $\bar{K}_3$.

7.5 Closed-Loop Stability under Distributed Reconfiguration

The proposed distributed algorithm converges to the optimum asymptotically as it solves the dual problem. Primal feasibility (model-matching), i.e., $H\eta[k] = \omega$, is only achieved in the limit. Therefore, one relevant concern is the system's stability when the dual algorithm is terminated in finite time. The following result shows that stability can be guaranteed in finite time, where the time will depend on the particular set of faults that have occurred.

Algorithm 7.5 Distributed sensor/actuator reconfiguration

1. Detect and isolate sensor/actuator faults and disconnect the faulty nodes at $t = 0$;
2. Locally compute q_i as per Lemma 7.4.6;
3. Compute the optimal solution $\zeta_i^\star$ to the dual problem (7.17) using the ADMM algorithm in Theorem 7.4.4;
4. Compute the primal optimal solution $\eta_i^\star = \frac{1}{2} S_{ii}^{-1} H_i^\top \zeta_i^\star$;
5. Each sensor/actuator node i applies $\eta_i^\star$.

Consider the general system $\dot{v} = \big((D + \Delta\big) v$ with D stable and uncertainty Δ, where $\text{vec}\,(\Delta) = H\eta[k] - \omega$. For the sensor reconfiguration analysis, we have $v = \hat{x}$, $D = A_e$, $H = (C^\top \Gamma_y^\top) \otimes L$ and $\omega = \text{vec}\,(LTC)$. Similarly, in the actuator reconfiguration case $v = x$, $D = A_c$, $H = (I \otimes B\Gamma_u)\, P_r^{-1}$ and $\omega = \text{vec}\,(BK)$.

First we recall a necessary and sufficient condition for robust stability with bounded uncertainties.

Lemma 7.5.1 (Lee *et al.* (1996)). *Consider the system* $\dot{v} = \big((D + \Delta\big) v$ *with* D *stable and uncertainty* Δ*. The system is stable for any norm-bounded uncertainty* $\|\Delta\|_F \leq \phi$ *with* $\phi > 0$ *if and only if there exists a positive definite matrix* X *such that*

$$D^\top X + XD + XX + \phi^2 I \prec 0.$$

Theorem 7.5.2. *Consider the sequence of vectors* $\{\eta[k]\}$ *converging to* $\eta^\star \in \mathcal{H} = \{\eta : H\eta = w\}$ *and define* $\Delta[k]$ *such that vec* $\big(\Delta[k]\big) = H\eta[k] - w$*. Suppose there exist matrices* $X \succ 0$ *and* $M \succ 0$ *satisfying the matrix equation* $D^\top X + XD + X^2 + M = 0$ *and a positive decreasing function of* k*,* $\epsilon[k] > 0$*, such that* $\|\Delta[k]\|_F \leq \epsilon[k] \|\Delta_0\|_F$ *holds for all* k*. Let* $\bar{k}$ *be an integer for which the following inequality holds:*

$$\epsilon_{\bar{k}} < \frac{\sqrt{\lambda_{\min}(M)}}{\|H\eta_0 - w\|}.$$

Then, the system under faults with dynamics given by $\dot{v} = \big(D + \Delta[k]\big) v$ *is stable for* $k \geq \bar{k}$*.*

Proof. Suppose that $\|\Delta[k]\|_F \leq \epsilon[k] \|\Delta[0]\|_F$ and consider $\phi[k] = \|\Delta[k]\|_F$. From Lemma 7.5.1, the closed-loop system at time k is guaranteed to be stable if $D^\top X + XD + X^2 + \phi[k]^2 I = -M + \phi[k]^2 I \prec 0$, which is equivalent to $\phi[k] < \sqrt{\lambda_{\min}(M)}$. Note that the latter is ensured for $\bar{k}$ when $\epsilon[\bar{k}]\phi_0 < \sqrt{\lambda_{\min}(M)}$. Since $\epsilon[k]$ is decreasing with k, concludes the proof. □

The above result provides a method to terminate the dual algorithm while ensuring stability. It only requires knowledge of the convergence properties of the dual algorithm, namely the function $\psi[k]$, and the initial distance $\|\Delta[0]\|_F$. The latter can be computed at the beginning, since it only depends on the initial condition of the algorithm and the nominal controller. Furthermore, note that a zero initial condition of the dual algorithm yields $\phi[0] = \|\omega\|$, which can be made locally available to each agent. Convergence properties of dual algorithms are readily available in the literature (Ghadimi *et al.*, 2014; Nedic *et al.*, 2010). Next we apply the results of Theorem 7.5.2 to the ADMM algorithm for the distributed reconfiguration problem formulated in Theorem 7.4.4.

Lemma 7.5.3. *Consider the optimization problem* (7.15)*, its equivalent dual formulation* (7.17)*, and the ADMM algorithm described in Theorem 7.4.4. Let* $\zeta^\star = \lim_{k\to\infty} \zeta[k]$ *be the optimal solution to* (7.17)*. Then, we have*

$$\|\zeta[k] - \zeta^\star\| \leq \psi \|\zeta[k-1] - \zeta^\star\|,$$

for all k *with* $\psi \in [0\ 1)$*.*

Proof. The proof follows directly from Ghadimi *et al.* (2014, Theorem 1), where the decay rate ψ can be found. □

Theorem 7.5.4. *Consider the optimization problem* (7.15)*, its equivalent dual formulation* (7.17)*, and the ADMM algorithm described in Theorem 7.4.4. The closed-loop system obtained at time* k *from* $\eta[k]$ *is guaranteed to be stable for all* $k \geq \bar{k}$ *with*

$$\bar{k} = \left\lceil \frac{\log(\sqrt{\lambda_{\min}(M)}) - \log\left(\|H\eta_0 - \omega\| \kappa(HS^{-1}H^\top)\right)}{\log(\psi)} \right\rceil .$$

Proof. We have $H\eta[k] = -1/2HS^-1H^\top\zeta_k$ for all k. Furthermore, we can derive the following bound

$$\|H\eta[k] - H\eta^\star\| = \|1/2HS^{-1}H^\top(\zeta_k - \zeta^\star)\| \leq \|1/2HS^{-1}H^\top\|_2 \|(\zeta_k - \zeta^\star)\|.$$

Using Lemma 7.5.3, we have

$$\begin{aligned}\|H\eta[k] - H\eta^\star\|_2 &\leq \|1/2HS^{-1}H^\top\|_2 \psi^k \|(\zeta[0] - \zeta^\star)\| \\ &\leq \kappa(HS^{-1}H^\top)\psi^k \|H\eta[0] - H\eta^\star\|.\end{aligned}$$

Recalling that $\|\Delta[0]\|_F = \|H\eta[0] - w\| = \|H\eta[0] - H\eta^\star\|$ and applying Theorem 7.5.2, we observe that the closed-loop system is stable for all k such that

$$\psi^k < \frac{\sqrt{\lambda_{\min}(M)}}{\|H\eta[0] - H\eta^\star\| \kappa(HS^{-1}H^\top)}.$$

The proof concludes by taking the logarithm of both sides and rearranging the terms. □

Next, we compute the matrices X and M that maximize the magnitude of the uncertainty for which stability is ensured.

Proposition 7.5.5. *Denote $X^\star$ and $\sigma^\star$ as the optimal solution to the convex optimization problem*

$$\begin{aligned}
\underset{X,\,\sigma}{\textit{maximize}}\quad & \sigma \\
\textit{subject to}\quad & \sigma > 0 \\
& X \succ 0 \\
& 0 \succ D^\top X + XD + \sigma I \\
& 0 \prec \begin{bmatrix} -D^\top X - XD - \sigma I & X \\ X & I \end{bmatrix}.
\end{aligned} \tag{7.18}$$

Then, matrix $X^\star$ satisfies the robust stability constraint $D^\top X + XD + X^2 + \phi^2 I \prec 0$ with $\phi^2 = \sigma^\star$ being the largest disturbance magnitude for which stability is ensured by Proposition 7.5.1. Additionally, we have that the optimal matrix M is given by $M^\star = -D^\top X^\star - X^\star D - X^{\star 2} \succ 0$.

Proof. Note that the largest disturbance magnitude ϕ for which stability is ensured by Lemma 7.5.1 can be computed as

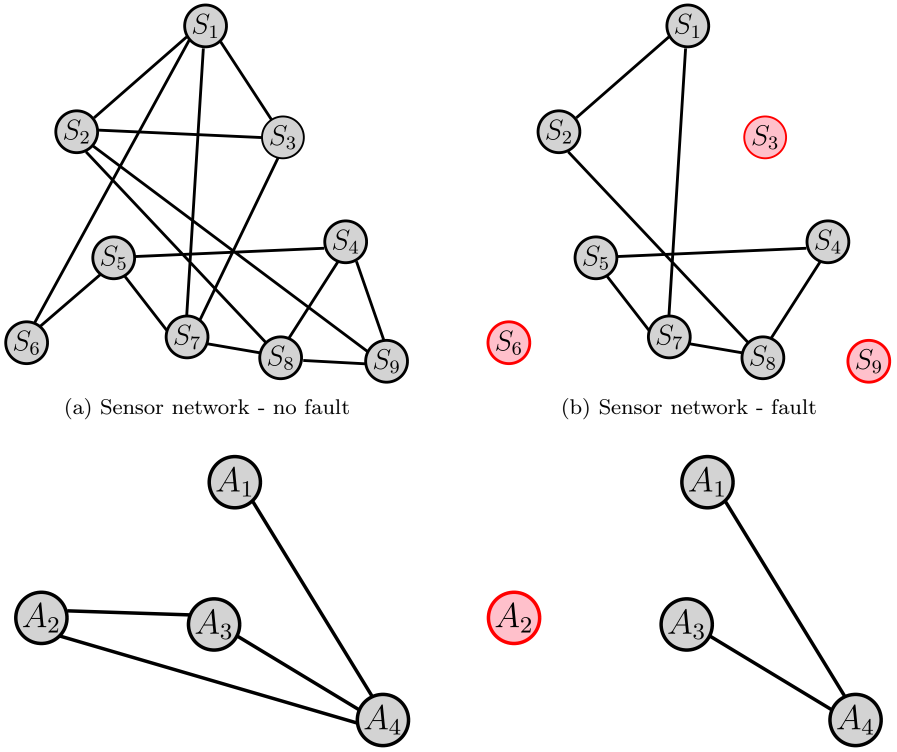
$$\begin{aligned}
\underset{X \succ 0,\, \phi^2 > 0}{\text{maximize}}\quad & \phi^2 \\
\text{subject to}\quad & 0 \succ D^\top X + XD + XX + \phi^2 I.
\end{aligned}$$

Applying the Schur complement to $-D^\top X - XD - \sigma I - XX \succ 0$ and denoting $\sigma = \phi^2$, the latter optimization problem can be rewritten as (7.18). □

The value $\bar{k}$ assures that stability can be achieved in a finite-time. Its calculation can be efficiently performed in a centralized manner, while a distributed computation would require knowledge of the particular set of faults by all nodes. We remark that, since Lemma 7.5.1 used in Theorem 7.5.4 provides a conservative stability guarantee, the obtained $\bar{k}$ is expected to be conservative. This will be later illustrated in the numerical example.

7.6 Numerical Example

We now provide a numerical example in order to validate the proposed distributed reconfiguration method. The aim is to control the temperature dynamics in two adjacent rooms, where 9 sensors are deployed to measure the temperature and 4 heaters actuate the system. The system dynamics, measured outputs and aggre-



(a) Sensor network - no fault

(b) Sensor network - fault

(c) Actuator network - no fault

(d) Actuator network - fault

Figure 7.4: Sensor and actuator network graph. The healthy nodes are colored black and the faulty nodes are colored red.

gated outputs are given by (7.1), (7.2) and (7.3), respectively, where

$$A = \begin{bmatrix} 9 & 2.5 \\ 4 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 56.5705 & 80.2208 & 4.1595 & -11.6809 \\ -3.2132 & -12.7760 & 57.3006 & 94.6012 \end{bmatrix},$$

$$C = \begin{bmatrix} 1 & 0.1 \\ -2 & -0.2 \\ 4 & 0.4 \\ 0.1 & 1 \\ -0.5 & -5 \\ 0.3 & 3 \\ 1 & 1 \\ 1 & 1 \\ 0.5 & 0.5 \end{bmatrix}, \quad T = \begin{bmatrix} 0.3689 & 0.2634 & 0 \\ 0.0424 & 0.1773 & 0 \\ 0.2422 & 0 & 0.5250 \\ 0 & 0.8812 & 0.7350 \\ 0.2480 & 0 & 0.8610 \\ 0 & 0.6299 & 0.6075 \\ 0 & 0.6057 & 0.1400 \\ 0 & 0.6443 & 0.6351 \\ 0.6414 & 0 & 0.1869 \end{bmatrix}^{\top}.$$

To enable reference tracking, the plant is augmented with two integral states, representing the integral error at each physical state. The control cost parameters are

$$R = \begin{bmatrix} 50 & 0 & 0 & 0 \\ 0 & 100 & 0 & 0 \\ 0 & 0 & 150 & 0 \\ 0 & 0 & 0 & 200 \end{bmatrix}$$

and $Q = 100I$, while the noise covariances are $V = 2I$ and $W = 5I$. Moreover, the state estimate and control input are given by (7.5) and (7.4), respectively. The initial estimation gain L and control input gain K are the solutions to the LQG controller design problem. The ADMM parameters in (7.16) are set to $\rho = 1$ and $\alpha = 1.5$. We highlight that there exist schemes to choose the ADMM parameters ρ and α to increase the convergence speed of the algorithm (Ghadimi *et al.*, 2014). These methods are optimal if executed in a centralized manner, but sub-optimal distributed methods are also provided in Ghadimi *et al.* (2014).

The sensor network graph is given in Figures 7.4a and 7.4b while the actuator network is depicted in Figures 7.4c and 7.4d, for the nominal and faulty cases.

We start by analyzing the performance of the distributed reconfiguration scheme presented in Section 7.4 for the sensor and actuator faults depicted in Figure 7.4. As performance indicators, we consider the normalized objective function errors $|J_e[k] - J_e^\star|$ and $|J_c[k] - J_c^\star|$, the errors in the model-matching constraint $\|H^e\eta[k] - w^e\|$ and $\|H^c\eta[k] - w^c\|$ and the maximum real part of the eigenvalues of $A_e[k] = A - L\bar{T}[k]\Gamma_y C$ and $A_c[k] = A - B\Gamma_u\bar{K}[k]$. The results are depicted in Figure 7.5. As it can be seen, the distributed method asymptotically achieves the optimal cost and guarantees the model-matching constraint. Moreover, the state estimation error dynamics is unstable for the first 2 steps, i.e., $\max_i \Big\{\Re\big(\lambda_i(A_e[k])\big)\Big\} > 0,\ k = 1, 2$, while the closed-loop dynamics are unstable for only the first step since $\max_i \Big\{\Re\big(\lambda_i(A_c[k])\big)\Big\} > 0,\ k = 1$. Applying Theorem 7.5.4 from Section 7.5, we achieve that $A_e[k]$ is stable for $k \geq \bar{k} = 53$ steps and $A_c[k]$ is stable for $k \geq \bar{k} = 8$ steps. Since Lemma 7.5.1 used in Theorem 7.5.4, provides a conservative stability guarantee, the obtained $\bar{k}$ is expected to be conservative. The distributed sensor reconfiguration takes 15 steps to converge to $|J_e[k] - J_e^\star| < 10^{-3}$ and $\|H^e\eta[k] - w^e\| < 10^{-1}$. Similarly, the distributed actuator reconfiguration takes approximately 16 steps to converge.

The time-responses of the distributed sensor and actuator reconfiguration under the faults in Figure 7.4 are depicted in Figure 7.6 where are shown the state trajectories, the estimation error and the control input values. In Figure 7.6 we depict the case where the sensor and actuator detection, isolation and reconfiguration is assumed to take place instantaneously (solid line) and the case where the sensor and actuator detection and isolation is instantaneous but the reconfiguration is performed in real-time (solid-star line). In the latter case, each step of the reconfiguration is set to take 1 s to run, which includes both computation and

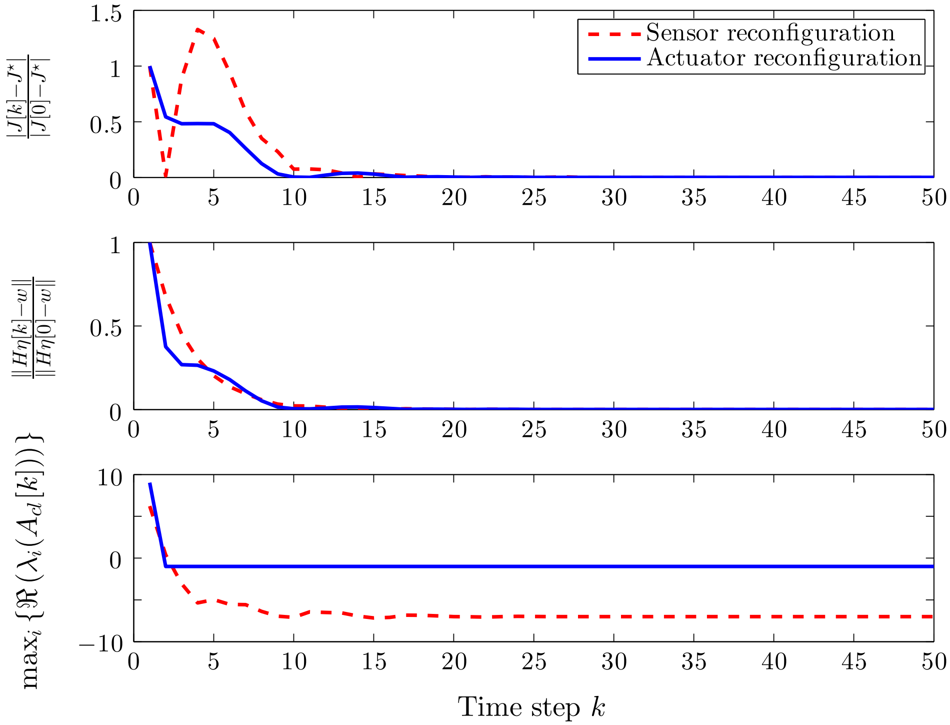


Figure 7.5: Performance of the distributed sensor and actuator reconfiguration method for the networks depicted in Figure 7.4.

communication time. Such a large time is selected so one can analyse the impact of a slow real-time reconfiguration in the system dynamics. However, in practice, the computation and communication times can be greatly reduced. This case aims at demonstrating the impact of applying the reconfigured output, before the reconfiguration algorithm has converged to a stable region, which takes at least, 3 s for the sensor reconfiguration and 1 s for the actuator reconfiguration. Additionally, we depict the case where reconfiguration does not take place (dashed line). The sensor faults occur at time $t = 10$ s and the actuator faults at $t = 300$ s. As it can be seen, the system performance greatly deteriorates when reconfiguration is not performed. When the actuator reconfiguration is not instantaneous, a slight loss of performance (maximum deviation of 0.1°C) occurs in the first 1 s, but is recovered afterwards.

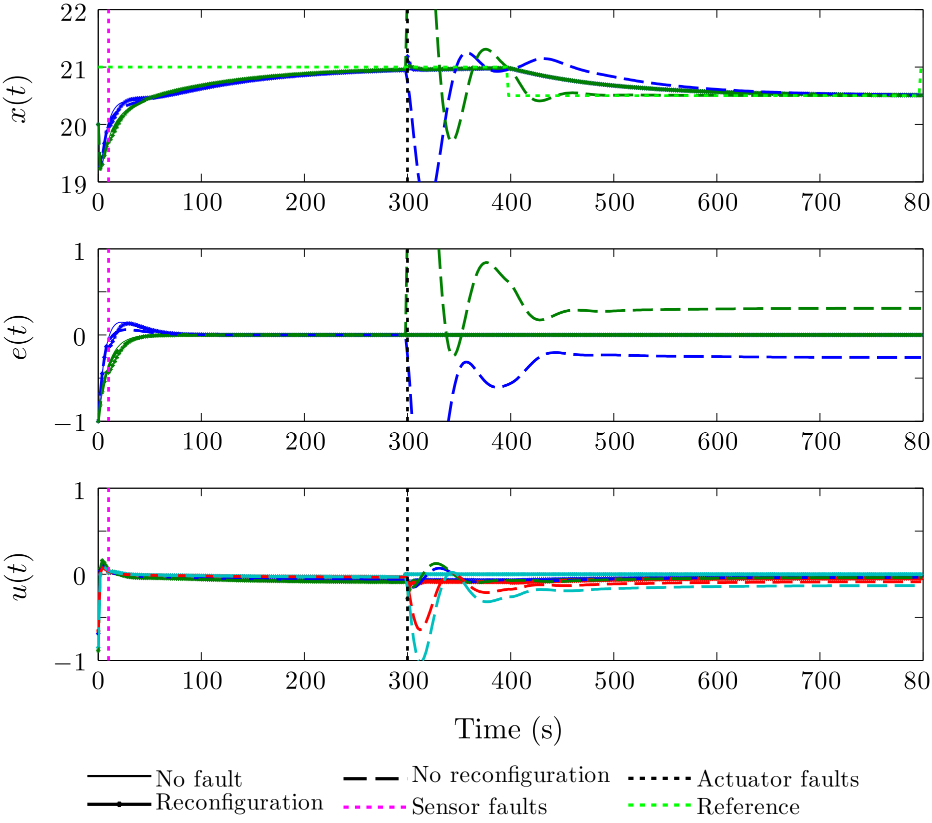


Figure 7.6: Time-response of the state and estimation error trajectories and control input for the distributed sensor and actuator reconfiguration in Figure 7.4. Reference value to be tracked is depicted by the black dotted line. Sensor faults occur at time $t = 10\,\mathrm{s}$ and actuator faults at $t = 300\,\mathrm{s}$. Instantaneous reconfiguration (solid), real-time reconfiguration (solid-star) and no reconfiguration (dashed), are compared.

7.7 Summary

In this chapter, we developed a distributed reconfiguration method for networked control systems under sensor and actuator faults. The proposed approach guarantees a model-matching reconfiguration while minimizing the steady-state estimation error covariance and a linear-quadratic control cost. The distributed reconfiguration method is guaranteed to achieve the same solution as the centralized reconfiguration, while only requiring local cooperation among healthy sensors and actuators. A numerical example demonstrates the effectiveness of our approach.

Chapter 8

Conclusions and Future Work

This thesis considered the cyber security and resilience of networked control systems, contributing towards a framework capable of analyzing and designing such systems. Building upon existing approaches within fault-tolerant systems and IT security, an conceptual architecture to study cyber security and resilience was discussed and illustrated in several attack scenarios. Metrics quantifying the impact and effort of attacks were described, and methods to detect stealthy attacks were proposed. Fault detection and isolation for large-scale systems with different types of faults were also discussed. Regarding the mitigation of failures, distributed reconfiguration schemes for actuator and sensor networks were proposed.

A brief summary of the thesis contributions and possible future research directions are discussed below.

8.1 Conclusions

In this thesis, we addressed several topics concerning the cyber security and resilience of networked control systems. The main contributions are as follows.

Bridging IT Security and Fault-Tolerant Control: In Chapter 2, we gave an overview of the fault-tolerant and IT security frameworks and discussed the conceptual differences between security and fault-tolerance in networked control systems. Some of these differences were illustrated through examples.

Modelling Framework for Malicious Adversaries: Models for malicious adversaries from a control-theoretic perspective were proposed for several attack scenarios in Chapter 3. Following the concepts behind fault-tolerant control, the core components relevant to security and resilience were identified and used to establish a reference architecture for networked control systems with malicious adversaries. An attack-scenario space based on this architecture was proposed, and it was used to map and qualitatively compare several attack scenarios studied in the literature.

Different scenarios were discussed in detail and illustrated through practical and numerical experiments.

Cyber Security Metrics for Stealthy Adversaries: The modelling framework was used to develop risk-related metrics in Chapter 4. In particular, the proposed metrics were formulated as constrained optimization problems, capturing trade-offs among adversary goals and constraints such as attack impact on the control system, attack detectability, and adversarial resources. Although the problems are non-convex, some can be related to system theoretic concepts such as zeros and weighted $\mathcal{H}_\infty$-norm of the closed-loop system and, thus, may be solved efficiently. Consequently, attacks stemming from unstable zeros were identified as stealthy attacks with a high impact potential.

Detection of Stealthy Adversaries: Chapter 5 addressed this class of open-loop stealthy attacks with high impact potential: the zero-dynamics attacks. The problem of revealing open-loop zero-dynamics attacks computed offline was addressed by modifying the system structure in terms of the outputs, inputs, and dynamics. For changes in each of these components, we provided necessary and sufficient conditions for attacks to be revealed. Furthermore, we provided an algorithm to incrementally add measurements and thus reveal attacks. A coordinated scaling of the inputs by the actuator and controller was also proposed. We quantified the resulting increase in output energy in terms of the initial condition and scaling factor. Both these changes on the inputs and outputs are able to reveal attacks while not affecting the system performance when no attack is present.

Distributed Fault Detection: Distributed fault detection and isolation (FDI) schemes for large-scale systems were proposed in Chapter 6. In particular, we considered this problem for networks of interconnected nodes with double integrator dynamics, corresponding to simplified models for teams of mobile robots or power networks. A distributed FDI scheme based on unknown input observers was proposed and its feasibility was analyzed with respect to local measurements. Some infeasibility results were also provided. The complexity of the proposed scheme was also analyzed, showing that it may not be scalable in the number of network nodes. Methods to reduce the complexity of the scheme were consequently discussed.

Distributed Reconfiguration in Networked Control Systems: In Chapter 7, a distributed reconfiguration scheme for networked control systems under sensor and actuator faults was proposed. The approach guarantees the recovery of the closed-loop dynamics, while minimizing the steady-state estimation error covariance and quadratic control cost. The distributed reconfiguration method is guaranteed to achieve the same solution as the centralized reconfiguration, while only requiring local cooperation among healthy sensors and actuators. Results establishing the stability of the closed-loop system, when the distributed algorithm is terminated in finite-time, are given.

8.2 Future Work

There are several research directions on cyber security and resilience to explore extending the work presented in this thesis. In this section, we discuss some of them.

Models for Resilient Control Systems: This thesis considered a reference architecture for networked control systems that included an ideal communication network and linear time-invariant models for the plant, controller, and anomaly detector. While numerous relevant questions can be tackled with such models, it may be necessary to generalize the architecture in order to capture other aspects. For instance, fault-tolerant control architectures commonly have supervisory schemes that choose the active components and control policies, depending on the output from the anomaly detectors. Such schemes have nonlinear effects that are not captured by linear systems, thus their response to attacks may not be fully captured with the models in the thesis. On a similar note, the performance of both the networked control system and the adversaries may be affected by network imperfections, such as packet losses and time-varying delays. In particular, the attack impact and stealthiness may drastically change in the presence of such uncertainties, as attacks that are stealthy with ideal networks may become detectable using schemes that explore statistical models of packet losses.

Security and Resilience Metrics: The metrics discussed in Chapter 4 considered trade-offs between impact and required resources for stealthy adversaries. Although the case of stealthy adversaries is interesting, analyzing the performance of resilient control systems with detectable adversaries is also relevant. In particular, resilience metrics for the case when threats are detected and mitigated have been proposed (Wei and Ji, 2010). The proposed maximum-impact metrics resemble the $\mathcal{H}_\infty$ robust control formulation, which suggests the design of control algorithms reducing such metrics and, thus, improving resiliency. Some security metrics were of combinatorial nature and may be hard to compute for large systems. Hence, developing efficient algorithms to compute or approximate such metrics is a relevant direction. In particular, application-specific models may provide structural properties that can be leveraged to develop efficient algorithms, as was recently shown for electric power systems (Sou *et al.*, 2013b).

Distributed Fault Diagnosis: The proposed distributed FDI scheme was applied to a swing-equation model of a power transmission network. However, more detailed models and different measurement configurations are sometimes needed.

Developing distributed FDI schemes for such systems is an interesting research direction. For instance, distributed FDI for linear differential-algebraic systems was recently tackled by Pasqualetti *et al.* (2013).

Distributed Fault-Tolerant Control: The distributed reconfiguration scheme discussed in the thesis did not require any particular system structure, apart from sufficient actuator and sensor redundancy to satisfy a model-matching constraint. However, certain structural properties may be leveraged to design fault-tolerant control schemes for scenarios with less redundancy.

Privacy in Estimation and Control: The attack scenarios discussed in the thesis were mostly comprised of data deception and denial-of-service attacks. In these scenarios, the adversary aimed at disrupting the system by tampering with the sensor and actuator data. In addition to such scenarios, disclosure attacks gathering private information from the plant and control algorithms are also relevant. Methodologies to address privacy while ensuring adequate levels of control and estimation performance are required to handle disclosure attacks.

Bibliography

1973. Fly-by-wire for combat aircraft. *Flight International.* Available at [Online]: `www.flightglobal.com/pdfarchive/view/1973/1973%20-%202228.html`. Last accessed: 10 Sep. 2014.

1991. ITU-T X.800: Security architecture for open systems interconnection for ccitt applications. *ITU Telecommunication Standardization Sector.* Available at [Online]: `handle.itu.int/11.1002/1000/3102`. Last accessed on: 10 Sep. 2014.

2013. ICS-CERT year in review. *U.S. DHS.* Available at [Online]: `ics-cert.us-cert.gov/sites/default/files/documents/Year_In_Review_FY2013_Final.pdf`. Last accessed: 10 Sep. 2014.

2014. Electric sector failure scenarios and impact analyses. *NESCOR, Electric Power Research Institute.* Available at [Online]: `www.smartgrid.epri.com/doc/NESCOR%20failure%20scenarios%2006-30-14a.pdf`. Last accessed: 10 Sep. 2014.

A. Abur and A.G. Exposito. 2004. *Power System State Estimation: Theory and Implementation.* Marcel-Dekker.

J. Åkerberg, M. Gidlund, and M. Björkman. 2011. Future research challenges in wireless sensor and actuator networks targeting industrial automation. In Proceedings of the *9th IEEE International Conference on Industrial Informatics.*

M. Aldeen and F. Crusca. 2006. Observer-based fault detection and identification scheme for power systems. *IEE Proceedings–Generation, Transmission and Distribution*, 153(1):71–79.

S. Amin, A. A. Cárdenas, and S. S. Sastry. 2009. Safe and secure networked control systems under denial-of-service attacks. In Rupak Majumdar and Paulo Tabuada, editors, *Hybrid Systems: Computation and Control*, volume 5469 of *Lecture Notes in Computer Science*, pages 31–45. Springer Berlin Heidelberg.

S. Amin, X. Litrico, S. S. Sastry, and A. M. Bayen. 2010. Stealthy deception attacks on water scada systems. In Proceedings of the *13th ACM International Conference on Hybrid Systems: Computation and Control*, CPSWeek.

S. Amin, G. A. Schwartz, and S. S. Sastry. 2013. Security of interdependent and identical networked control systems. *Automatica*, 49(1):186–192.

K. J. Åström. 1970. *Introduction to Stochastic Control Theory.* Academic Press. Republished by Dover Publications, 2006.

K. J. Åström and P. R. Kumar. 2014. Control: A perspective. *Automatica*, 50(1): 3–43.

K. J. Åström and B. Wittenmark. 1997. *Computer-Controlled Systems.* Prentice Hall.

Luigi Atzori, Antonio Iera, and Giacomo Morabito. 2010. The internet of things: A survey. *Computer Networks*, 54(15):2787–2805.

J. Baillieul and P. J. Antsaklis. 2007. Control and communication challenges in networked real-time systems. *Proceedings of the IEEE*, 95(1):9–28.

N. Balu, T. Bertram, A. Bose, V. Brandwajn, G. Cauley, D. Curtice, A. Fouad, L. Fink, M.G. Lauby, B.F. Wollenberg, and J.N. Wrubel. 1992. On-line power system security analysis. *Proceedings of the IEEE*, 80(2):262–282.

B. Bamieh, F. Paganini, and M.A Dahleh. 2002. Distributed control of spatially invariant systems. *IEEE Transactions on Automatic Control*, 47(7):1091–1107.

Y. Bar-Shalom. 2002. Update with out-of-sequence measurements in tracking: exact solution. *IEEE Transactions on Aerospace and Electronic Systems*, 38(3):769–777.

P. Barooah and J.P. Hespanha. 2006. Graph effective resistance and distributed control: Spectral properties and applications. In Proceedings of the *45th IEEE Conference on Decision and Control*, San Diego, CA, USA, December 2006.

A. Barrat, M. Barthlemy, and A. Vespignani. 2008. *Dynamical Processes on Complex Networks.* Cambridge University Press, New York, NY, USA.

T. Basar and P. Bernhard. 1995. *H^∞-Optimal Control and Related Minimax Design Problems: A Dynamic Game Approach.* Birkhäuser, Boston, MA.

G. Basile and G. Marro. 1992. *Controlled and Conditioned Invariants in Linear System theory.* Prentice Hall.

M. Basseville and I. V. Nikiforov. 1993. *Detection of Abrupt Changes: Theory and Application.* Prentice-Hall, Inc., Upper Saddle River, NJ, USA.

J. Bendtsen, K. Trangbaek, and J. Stoustrup. 2013. Plug-and-play control - modifying control systems online. *IEEE Transactions on Control Systems Technology*, 21(1):79–93.

M. Bishop. 2002. *Computer Security: Art and Science.* Addison-Wesley Professional.

M. Blanke, M. Kinnaert, J. Lunze, and M. Staroswiecki. 2006. *Diagnosis and Fault-Tolerant Control*, 2nd edition. Springer-Verlag.

S. Bodenburg and J. Lunze. 2013. Plug-and-play control - theory and implementation. In Proceedings of the *11th IEEE International Conference on Industrial Informatics.*

E. Bompard, C. Gao, R. Napoli, A. Russo, M. Masera, and A. Stefanini. 2009. Risk assessment of malicious attacks against power systems. *IEEE Transactions Systems, Man and Cybernetics, Part A: Systems and Humans*, 39(5):1074–1085.

S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan. 1994. *Linear Matrix Inequalities in System and Control Theory*, volume 15 of *Studies in Applied Mathematics.* SIAM, Philadelphia, PA.

S. Boyd, N. Parikh, E. Chu, B.a Peleato, and J. Eckstein. 2011. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine Learning*, 3(1):1–122.

S. Boyd and L. Vandenberghe. 2004. *Convex Optimization.* Cambridge University Press.

J.C. Campelo, F. Rodriguez, A. Rubio, R. Ors, P.J. Gil, L. Lemus, J.V. Busquets, J. Albaladejo, and J.J. Serrano. 1999. Distributed industrial control systems: a fault-tolerant architecture. *Microprocessors and Microsystems*, 23(2):103 – 112.

A. A. Cárdenas, S. Amin, Z. Lin, Y. Huang, C. Huang, and S. S. Sastry. 2011. Attacks against process control systems: risk assessment, detection, and response. In Proceedings of the *6th ACM Symposium on Information, Computer and Communications Security*, ASIACCS.

A. A. Cárdenas, S. Amin, and S. S. Sastry. 2008a. Secure control: Towards survivable cyber-physical systems. In Proceedings of the *First International Workshop on Cyber-Physical Systems.*

A. A. Cárdenas, S. Amin, B. Sinopoli, A. Giani, A. Perrig, and S. S. Sastry. 2009. Challenges for securing cyber physical systems. In Proceedings of the *Workshop on Future Directions in Cyber-physical Systems Security.* U.S. DHS.

A.A. Cárdenas, S. Amin, and S.S. Sastry. 2008b. Research challenges for the security of control systems. In Proceedings of the *3rd USENIX Workshop on Hot Topics in Security.*

CBSNews. 2009. Cyber war: Sabotaging the system. *CBSNews.* Available at [Online]: `http://www.cbsnews.com/news/cyber-war-sabotaging-the-system-06-11-2009/`.

R. Chabukswar, Y. Mo, and B. Sinopoli. 2011. Detecting integrity attacks on scada systems. In Proceedings of the *18th IFAC World Congress*.

J. Chen and R. J. Patton. 1999. *Robust Model-Based Fault Diagnosis for Dynamic Systems*. Kluwer Academic Publishers.

J. Chen, R. J. Patton, and H. Zhang. 1996. Design of unknown input observers and robust fault detection filters. *International Journal of Control*, 63(1):85–105.

X. Chen, K. Makki, K. Yen, and N. Pissinou. 2009. Sensor network security: A survey. *IEEE Communications Surveys & Tutorials*, 11(2):52–73.

Y. Chompoobutrgool, L. Vanfretti, and M. Ghandhari. 2011. Survey on power system stabilizers control and their prospective applications for power system damping using synchrophasor-based wide-area systems. *European Transactions on Electrical Power*, 21(8):2098–2111.

E. Chow and A. Willsky. 1984. Analytical redundancy and the design of robust failure detection systems. *IEEE Transactions on Automatic Control*, 29(7):603–614.

W.H. Chung and J.L. Speyer. 1998. A game theoretic fault detection filter. *IEEE Transactions on Automatic Control*, 43(2):143–161.

G. Dán and H. Sandberg. 2010. Stealth attacks and protection schemes for state estimators in power systems. In Proceedings of the *First IEEE International Conference on Smart Grid Communications*.

M. A. Demetriou. 2005. Using unknown input observers for robust adaptive fault detection in vector second-order systems. *Mechanical systems and signal processing*, 19(2):291–309.

B. Demirel, Z. Zou, P. Soldati, and M. Johansson. 2014. Modular design of jointly optimal controllers and forwarding policies for wireless control. *IEEE Transactions on Automatic Control*. To appear.

S. X. Ding. 2008. *Model-based Fault Diagnosis Techniques: Design Schemes*. Springer Verlag.

S. X. Ding, P. Zhang, C. Chihaia, W. Li, Y. Wang, and E. L. Ding. 2008. Advanced design scheme for fault tolerant distributed networked control systems. In Proceedings of the *17th IFAC World Congress*.

R. Douglas and J. Speyer. 1995. Robust fault detection filter design. In Proceedings of the *American Control Conference*.

M.J. Ellsworth, L.A. Campbell, R.E. Simons, and R.R.S. Iyengar. 2008. The evolution of water cooling for IBM large server systems: Back to the future. In Proceedings of the *11th Intersociety Conference on Thermal and Thermomechanical Phenomena in Electronic Systems*, ITHERM.

P. Esfahani, M. Vrakopoulou, K. Margellos, J. Lygeros, and G. Andersson. 2010. Cyber attack in a two-area power system: Impact identification using reachability. In Proceedings of the *American Control Conference*.

Hugh Everett III. 1963. Generalized lagrange multiplier method for solving problems of optimum allocation of resources. *Operations research*, 11(3):399–417.

N. Falliere, L. Murchu, and E. Chien. 2011. W32.Stuxnet dossier. *Symantec*. Available at [Online]: `www.symantec.com/content/en/us/enterprise/media/security_response/whitepapers/w32_stuxnet_dossier.pdf`. Last accessed: 10 Sep. 2014.

H. Fawzi, P. Tabuada, and S. Diggavi. 2012. Security for control systems under sensor and actuator attacks. In Proceedings of the *51st IEEE Conference on Decision and Control*.

FERC. 2003. Final report on price manipulation in western markets. Available at [Online]: `www.ferc.gov/industries/electric/indus-act/wec.asp`. Last accessed: 10 Sep. 2014.

R. Ferrari, T. Parisini, and M.M. Polycarpou. 2009. Distributed fault diagnosis with overlapping decompositions: An adaptive approximation approach. *IEEE Transactions on Automatic Control*, 54:794–799.

P. M. Frank and X. Ding. 1997. Survey of robust residual generation and evaluation methods in observer-based fault detection systems. *Journal of process control*, 7 (6):403–424.

B. Galloway and G. P. Hancke. 2013. Introduction to industrial control networks. *IEEE Communications Surveys & Tutorials*, 15(2):860–880.

Z. Gao and P. J. Antsaklis. 1991. Stability of the pseudo-inverse method for reconfigurable control systems. *International Journal of Control*, 53(3):717–729.

I. Garitano, R. Uribeetxeberria, and U. Zurutuza. 2011. A review of scada anomaly detection systems. In E. Corchado, V. Snášel, J. Sedano, A. E. Hassanien, J. L. Calvo, and D. Ślęzak, editors, *6th International Conference on Soft Computing Models in Industrial and Environmental Applications*, volume 87 of *Advances in Intelligent and Soft Computing*, pages 357–366. Springer Berlin Heidelberg.

E. Ghadimi, A. Teixeira, M. Rabbat, and M. Johansson. 2014. The ADMM algorithm for distributed averaging: convergence rates and optimal parameter selection. In Proceedings of the *48th Asilomar Conference on Signals, Systems, and Computers*, Pacific Grove, CA, USA. To appear.

A. Giani, E. Bitar, M. Garcia, M. McQueen, P. Khargonekar, and K. Poolla. 2013. Smart grid data integrity attacks. *IEEE Transactions on Smart Grid*, 4(3):1244–1253.

A. Giani, S. Sastry, K. H. Johansson, and H. Sandberg. 2009. The VIKING project: an initiative on resilient control of power networks. In Proceedings of the *2nd International Symposium on Resilient Control Systems.*

S. Gorman. 2009. Electricity grid in U.S. penetrated by spies. *The Wall Street Journal.* Available at [Online]: `http://online.wsj.com/articles/SB123914805204099085`. Last accessed: 10 Sep. 2014.

F. Grandoni. 2006. A note on the complexity of minimum dominating set. *J. Discrete Algorithms*, 4(2):209–214.

A. Gupta, C. Langbort, and T. Başar. 2010. Optimal control in the presence of an intelligent jammer with limited actions. In Proceedings of the *49th IEEE Conference on Decision and Control.*

V. Gupta, B. Hassibi, and R. M. Murray. 2007. Optimal LQG control across packet-dropping links. *Systems & Control Letters*, 56(6):439–446.

Z. Han, W. Li, and S. L. Shah. 2005. Fault detection and isolation in the presence of process uncertainties. *Control engineering practice*, 13(5):587–599.

O. Härkegård and S. T. Glad. 2005. Resolving actuator redundancy: optimal control vs. control allocation. *Automatica*, 41(1):137–144.

J.P. Hespanha, P. Naghshtabrizi, and Yonggang Xu. 2007. A survey of recent results in networked control systems. *Proceedings of the IEEE*, 95(1):138–162.

J.B. Hiriart-Urruty. 2001. Global optimality conditions in maximizing a convex quadratic function under convex quadratic constraints. *Journal of Global Optimization*, 21(4):443–453.

Z. G. Hou, L. Cheng, and M. Tan. 2009. Decentralized robust adaptive control for the multiagent system consensus problem using neural networks. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, 39(3):636–647.

I. Hwang, S. Kim, Y. Kim, and C. E. Seah. 2010. A survey of fault detection, isolation, and reconfiguration methods. *IEEE Transactions on Control Systems Technology*, 18(3):636–653.

ICS-CERT. 2010. ICS-ALERT-10-301-01: Control system internet accessibility. Available at [Online]: `https://ics-cert.us-cert.gov/alerts/ICS-ALERT-10-301-01`. Last accessed: 10 Sep. 2014.

R. Isermann. 2004. Model-based fault detection and diagnosis: status and applications. In Proceedings of the *Proceedings of the 16th IFAC Symposium on Automatic Control in Aerospace.*

R. Isermann. 2006. *Fault-Diagnosis Systems: An Introduction from Fault Detection to Fault Tolerance.* Springer.

H. Ishii and B. A. Francis. 2002. *Limited Data Rate in Control Systems with Networks*, volume 275 of *Lecture Notes in Control and Information Sciences.* Springer-Verlag.

A. Isidori. 1995. *Nonlinear Control Systems*, 3rd edition. Springer-Verlag.

S Jiang, PG Voulgaris, and N Neogi. 2007. Failure-robust distributed controller architectures. *International Journal of Control*, 80(9):1367–1378.

X. Z. Jin and G. H. Yang. 2009. Distributed fault-tolerant control systems design against actuator faults and faulty interconnection links: An adaptive method. In Proceedings of the *American Control Conference.*

T. A. Johansen and T. I. Fossen. 2013. Control allocation - a survey. *Automatica*, 49(5):1087–1103.

B. Johansson. 2008. *On Distributed Optimization in Networked Systems.* PhD thesis, KTH, Automatic Control.

K.H. Johansson. 2000. The quadruple-tank process: a multivariable laboratory process with an adjustable zero. *IEEE Transactions on Control Systems Technology*, 8(3):456–465.

S. Jokar and M. E. Pfetsch. 2008. Exact and approximate sparse solutions of underdetermined linear equations. *SIAM Journal on Scientific Computing*, 31 (1):23–44.

S. Kaplan and B. J. Garrick. 1981. On the quantitative definition of risk. *Risk Analysis*, 1(1):11–27.

U.A Khan and J.M.F. Moura. 2008. Distributing the kalman filter for large-scale systems. *IEEE Transactions on Signal Processing*, 56(10):4919–4935.

A. Khanafer, B. Touri, and T. Başar. 2012. Consensus in the presence of an adversary. In Proceedings of the *3rd IFAC Workshop on Estimation and Control of Networked Systems*, NecSys.

T.T. Kim and H.V. Poor. 2011. Strategic protection against data injection attacks on power grids. *IEEE Transactions on Smart Grid*, 2(2):326–333.

I. Koren and C. M. Krishna. 2010. *Fault-tolerant systems.* Morgan Kaufmann.

O. Kosut, L. Jia, R. Thomas, and L. Tong. 2010. Malicious data attacks on smart grid state estimation: Attack strategies and countermeasures. In Proceedings of the *First IEEE International Conference on Smart Grid Communications*.

O. Kosut, L. Jia, R. J. Thomas, and L. Tong. 2011. Malicious data attacks on the smart grid. *IEEE Transactions on Smart Grid*, 2(4):645–658.

R. Krutz. 2006. *Securing SCADA Systems*. Wiley Publishing, Inc.

P. Kundur. 1994. *Power System Stability and Control*. McGraw-Hill Professional.

D. Kushner. 2013. The real story of stuxnet. *IEEE Spectrum*. Available at [Online]: `spectrum.ieee.org/telecom/security/the-real-story-of-stuxnet`. Last accessed: 10 Sep. 2014.

C. Langbort, R.S. Chandra, and R. D'Andrea. 2004. Distributed control design for systems interconnected over an arbitrary graph. *IEEE Transactions on Automatic Control*, 49(9):1502–1519.

H.J. LeBlanc, H. Zhang, X. Koutsoukos, and S. Sundaram. 2013. Resilient asymptotic consensus in robust networks. *IEEE Journal on Selected Areas in Communications*, 31(4):766–781.

J. H. Lee, W. H. Kwon, and J.-W. Lee. 1996. Quadratic stability and stabilization of linear systems with Frobenius norm-bounded uncertainties. *IEEE Transactions on Automatic Control*, 41(3):453–456.

Y. Liu, M. K. Reiter, and P. Ning. 2009. False data injection attacks against state estimation in electric power grids. In Proceedings of the *16th ACM Conference on Computer and Communications Security*.

J. Lunze. 1992. *Feedback Control of Large Scale Systems*. Prentice Hall PTR, Upper Saddle River, NJ, USA.

J. Lunze and J. H. Richter. 2008. Reconfigurable fault-tolerant control: a tutorial introduction. *European Journal of Control*, 14(5):359–386.

J. Lunze and T. Steffen. 2006. Control reconfiguration after actuator failures using disturbance decoupling methods. *IEEE Transactions on Automatic Control*, 51 (10):1590–1601.

N.A. Lynch. 1997. *Distributed Algorithms*, 1st edition. Morgan Kaufmann.

J. Machowski, J. W. Bialek, and J. R. Bumby. 2008. *Power System Dynamics: Stability and Control*. John Wiley & Sons.

J.M. Maciejowski. 1997. Reconfigurable control using constrained optimization. In Proceedings of the *European Control Conference*, pages 107–130.

R. T. Marler and J. S. Arora. 2004. Survey of multi-objective optimization methods for engineering. *Structural and Multidisciplinary Optimization*, 26(6):369–395.

M. A. Massoumnia and G. C. Verghese. 1989. Failure detection and identification. *IEEE Transactions on Automatic Control*, 34:316–321.

John C. Matherly. 2009. Shodan: the computer search engine. Available at [Online]: http://www.shodanhq.com/help.

S.T. McCormick. 1983. *A combinatorial approach to some sparse matrix problems*. PhD thesis, Stanford University.

V.A. Megna and K.J. Szalai. 1977. Multi-flight computer redundancy management for digital fly-by-wire aircraft control. In Proceedings of the *COMPCON*.

J. Meserve. 2007. Staged cyber attack reveals vulnerability in power grid. *CNN*. Available at [Online]: http://edition.cnn.com/2007/US/09/26/power.at.risk/index.html. Last accessed: 10 Sep. 2014.

F. Miao, M. Pajic, and G.J. Pappas. 2013. Stochastic game approach for replay attack detection. In Proceedings of the *IEEE 52nd Conference on Decision and Control*.

Y. Mo and B. Sinopoli. 2009. Secure control against replay attack. In Proceedings of the *47th Annual Allerton Conference on Communication, Control, and Computing*.

Y. Mo and B. Sinopoli. 2012. Integrity attacks on cyber-physical systems. In Proceedings of the *1st International Conference on High Confidence Networked Systems, CPSWeek 2012*.

A. Monticelli. 1999. *State Estimation in Electric Power Systems: A Generalized Approach*. Kluwer Academic Publishers.

K.J. Morrisse, G.F. Solimini, and U.A. Khan. 2012. Distributed control schemes for wind-farm power regulation. In Proceedings of the *North American Power Symposium (NAPS), 2012*.

B. Nabet, N. E. Leonard, I. D. Couzin, and S. A. Levin. 2009. Dynamics of decision making in animal group motion. *Journal of Nonlinear Science*, 19(4):399–435.

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

H. Nishino and H. Ishii. 2014. Distributed detection of cyber attacks and faults for power systems. In Proceedings of the *19th IFAC World Congress*.

NIST. 2012. Special publication 800-30: Guide for conducting risk assessments. *National Institute of Standards and Technology.* Available at [Online]: `csrc.nist.gov/publications/nistpubs/800-30-rev1/sp800_30_r1.pdf`. Last accessed: 10 Sep. 2014.

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

Z.-H. Pang and G.-P. Liu. 2012. Design and implementation of secure networked predictive control systems under deception attacks. *IEEE Transactions on Control Systems Technology*, 20(5):1334–1342.

F. Pasqualetti, A. Bicchi, and F. Bullo. 2007. Distributed intrusion detection for secure consensus computations. In Proceedings of the *46th IEEE Conference on Control and Decision.*

F. Pasqualetti, A. Bicchi, and F. Bullo. 2012. Consensus computation in unreliable networks: A system theoretic approach. *IEEE Transactions on Automatic Control*, 57:90–104.

F. Pasqualetti, F. Dorfler, and F. Bullo. 2011. Cyber-physical attacks in power networks: Models, fundamental limitations and monitor design. In Proceedings of the *50th IEEE Conference on Decision and Control and European Control Conference.*

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

R. J. Patton. 1997. Fault-tolerant control systems: The 1997 situation. In Proceedings of the *IFAC Symposium on Fault Detection Supervision and Safety for Technical Processes*, SAFEPROCESS.

R. J. Patton and J. Chen. 1997. Observer-based fault detection and isolation: robustness and applications. *Control Engineering Practice*, 5(5):671–682.

A.G. Phadke and R.M. de Moraes. 2008. The wide world of wide-area measurement. *IEEE Power and Energy Magazine*, 6(5):52–65.

R. Poovendran, K. Sampigethaya, S. K. S. Gupta, I. Lee, K. V. Prasad, D. Corman, and J. Paunicka. 2012. Special issue on cyber - physical systems. *Proceedings of the IEEE*, 100(1):6 –12.

J. Qin, W. X. Zheng, and H. Gao. 2012. Coordination of multiple agents with double-integrator dynamics under generalized interaction topologies. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, 42(1): 44–57.

C. Ramesh, H. Sandberg, and K.H. Johansson. 2013. Design of state-based schedulers for a network of control loops. *IEEE Transactions on Automatic Control*, 58(8):1962–1975.

W. Ren and E. Atkins. 2007. Distributed multi-vehicle coordinated control via local information exchange. *International Journal of Robust Nonlinear Control*, 17(10-11):1002–1033.

J. Richard. 2003. Time-delay systems: an overview of some recent advances and open problems. *Automatica*, 39(10):1667–1694.

J. H. Richter, W. P. M. H. Heemels, N. van de Wouw, and J. Lunze. 2011. Reconfigurable control of piecewise affine systems with actuator and sensor faults: Stability and tracking. *Automatica*, 47(4):678–691.

T. Rid. 2011. Cyber war will not take place. *Journal of Strategic Studies*, 35(1): 5–32.

C.G. Rieger. 2010. Notional examples and benchmark aspects of a resilient control system. In Proceedings of the *3rd International Symposium on Resilient Control Systems*, ISRCS, pages 64–71.

C.G. Rieger, D.I. Gertman, and M.A. McQueen. 2009. Resilient control systems: Next generation design research. In Proceedings of the *2nd Conference on Human System Interactions*, pages 632–636.

S. Riverso, M. Farina, and G. Ferrari-Trecate. 2013. Plug-and-play decentralized model predictive control for linear systems. *IEEE Transactions on Automatic Control*, 58(10):2608–2614.

T. Samad and A.M. Annaswamy, editors. 2011. *The Impact of Control Technology.* IEEE Control Systems Society. Available at [Online]: `http://www.ieeecss.org/general/impact-control-technology`.

T. Samad, P. McLaughlin, and J. Lu. 2007. System architecture for process automation: Review and trends. *Journal of Process Control*, 17(3):191–201.

H. Sandberg, A. Teixeira, and K. H. Johansson. 2010. On security indices for state estimators in power networks. In Proceedings of the *First Workshop on Secure Control Systems, CPSWeek*, Stockholm, Sweden, April 2010.

L. Schenato. 2009. To zero or to hold control inputs with lossy links? *IEEE Transactions on Automatic Control*, 54(5):1093–1099.

L. Schenato, B. Sinopoli, M. Franceschetti, K. Poolla, and S. Sastry. 2007. Foundations of control and estimation over lossy networks. *Proceedings of the IEEE*, 95(1):163–187.

E. Scholtz and B.C. Lesieutre. 2008. Graphical observer design suitable for large-scale DAE power systems. In Proceedings of the *47th IEEE Conference on Decision and Control*.

Alexander Schrijver. 1986. *Theory of Linear and Integer Programming*. John Wiley & Sons, Inc., New York, NY, USA.

F. C. Schweppe and J. Wildes. 1970. Power system static-state estimation, part I: Exact model. *IEEE Transactions on Power Apparatus and Systems*, 89(1): 120–125.

M. Shahidehpour, F. Tinney, and Y. Fu. 2005. Impact of security on power systems operation. *Proceedings of the IEEE*, 93(11):2013–2025.

W. Shefte, S. Al-Jamea, and R. O'Harrow. 2012. Cyber search engine shodan exposes industrial control systems to new risks. *The Washington Post*. Available at [Online]: `http://www.washingtonpost.com/investigations/cyber-search-engine-exposes-vulnerabilities/2012/06/03/gJQAIK9KCV_story.html`. Last accessed: 10 Sep. 2014.

N. Z. Shor, K. C. Kiwiel, and A. Ruszcaynski. 1985. *Minimization Methods for Non-differentiable Functions*. Springer-Verlag New York, Inc.

D. D. Siljak. 1991. *Decentralized Control of Complex Systems*. Academic Press, New York, USA.

Sigurd Skogestad and Ian Postlethwaite. 1996. *Multivariable Feedback Control: Analysis and Design*. John Wiley & Sons.

R. Smith. 2011. A decoupled feedback structure for covertly appropriating networked control systems. In Proceedings of the *18th IFAC World Congress*.

T. Sommestad, M. Ekstedt, and H. Holm. 2013. The cyber security modeling language: A tool for assessing the vulnerability of enterprise system architectures. *IEEE Systems Journal*, 7(3):363–373.

K. C. Sou, H. Sandberg, and K. H. Johansson. 2013a. Data attack isolation in power networks using secure voltage magnitude measurements. *IEEE Transactions on Smart Grid*, 5(1):14–28.

K. C. Sou, H. Sandberg, and K.H. Johansson. 2013b. On the exact solution to a smart grid cyber-security analysis problem. *IEEE Transactions on Smart Grid*, 4(2):856–865.

S. Sridhar, A. Hahn, and M. Govindarasu. 2012. Cyber-physical system security for the electric power grid. *Proceedings of the IEEE*, 100(1):210–224.

M. Staroswiecki. 2005. Fault tolerant control : The pseudo-inverse method revisited. In Proceedings of the *16th IFAC World Congress*.

M. Staroswiecki and D. Berdjag. 2010. A general fault tolerant linear quadratic control strategy under actuator outages. *International Journal of Systems Science*, 41(8):971–985.

M. Staroswiecki and F. Cazaurang. 2008. Fault recovery by nominal trajectory tracking. In Proceedings of the *American Control Conference*.

M. Staroswiecki, H. Yang, and B. Jiang. 2007. Progressive accommodation of parametric faults in linear quadratic control. *Automatica*, 43(12):2070–2076.

S. Sundaram and C.N. Hadjicostis. 2011. Distributed function calculation via linear iterative strategies in the presence of malicious agents. *IEEE Transactions on Automatic Control*, 56(7):1495–1508.

S. Sundaram, S. Revzen, and G. Pappas. 2012. A control-theoretic approach to disseminating values and overcoming malicious links in wireless networks. *Automatica*, 48(11):2894–2901.

Symantec. 2011. W32.Duqu: The precursor to the next stuxnet. *Symantec*.

Symantec. 2012. Flamer: Highly sophisticated and discreet threat targets the middle east. *Symantec*. Available at [Online]: `www.symantec.com/connect/blogs/flamer-highly-sophisticated-and-discreet-threat-targets-middle-east`. Last accessed: 10 Sep. 2014.

Symantec. 2014. Dragonfly: Cyberespionage attacks against energy suppliers. *Symantec*. Available at [Online]: `http://www.symantec.com/content/en/us/enterprise/media/security_response/whitepapers/Dragonfly_Threat_Against_Western_Energy_Suppliers.pdf`. Last accessed: 10 Sep. 2014.

A. Tanenbaum and D. J. Wetherall. 2010. *Computer Networks*, 5th edition. Prentice Hall.

G. Tao, S. Chen, and S. M. Joshi. 2002. An adaptive control scheme for systems with unknown actuator failures. *Automatica*, 38(6):1027–1034.

A. Teixeira, G. Dán, H. Sandberg, and K. H. Johansson. 2011. Cyber security study of a SCADA energy management system: stealthy deception attacks on the state estimator. In Proceedings of the *18th IFAC World Congress*.

A. Teixeira, H. Sandberg, G. Dán, and K. H. Johansson. 2012a. Optimal power flow: closing the loop over corrupted data. In Proceedings of the *American Control Conference*.

A. Teixeira, I. Shames, H. Sandberg, and K. H. Johansson. 2012b. Revealing stealthy attacks in control systems. In Proceedings of the *50th Annual Allerton Conference on Communication, Control, and Computing*.

IBM. IBM ILOG CPLEX Optimizer. Available at [Online]: `www-01.ibm.com/software/integration/optimization/cplex-optimizer/`. Last accessed on: 10 Sep. 2014.

A. M. Tillmann and M. E. Pfetsch. 2012. The computational complexity of the restricted isometry property, the nullspace property, and related concepts in compressed sensing.

J. Tokarzewski. 2006. *Finite zeros in discrete time control systems*, volume 338 of *Lecture notes in control and information sciences.* Springer Berlin Heidelberg.

J. Tsitsiklis and D. Bertsimas. 1997. *Introduction to Linear Optimization.* Athena Scientific.

U.S.-Canada PSOTF. 2004. Final report on the August 14th blackout in the United States and Canada. Technical report, U.S.-Canada Power System Outage Task Force.

U.S. DHS. 2011. Risk management fundamentals. *U.S. Department of Homeland Security.* Available at [Online]: `www.dhs.gov/xlibrary/assets/rma-risk-management-fundamentals.pdf`. Last accessed: 10 Sep. 2014.

U.S. GAO. 2004. Critical infrastructure protection: Challenges and efforts to secure control systems. *U.S. GAO.* Available at [Online]: `http://www.gao.gov/assets/250/241726.pdf`. Last accessed: 10 Sep. 2014.

O. Vukovic, K. C. Sou, G. Dán, and H. Sandberg. 2012. Network-aware mitigation of data integrity attacks on power system state estimation. *IEEE Journal on Selected Areas in Communications*, 30(6):1108–1118.

X. Wang and M.D. Lemmon. 2011. Event-triggering in distributed networked control systems. *IEEE Transactions on Automatic Control*, 56(3):586–601.

D. Wei and K. Ji. 2010. Resilient industrial control system (RICS): Concepts, formulation, metrics, and insights. In Proceedings of the *3rd International Symposium on Resilient Control Systems.*

A. S. Willsky. 1976. A survey of design methods for failure detection in dynamic systems. *Automatica*, 12(6):601–611.

F.F. Wu, K. Moslehi, and A. Bose. 2005. Power system control centers: Past, present, and future. *Proceedings of the IEEE*, 93(11):1890–1908.

N.Eva Wu, Kemin Zhou, and Gregory Salomon. 2000. Control reconfigurability of linear time-invariant systems. *Automatica*, 36(11):1767–1771.

L. Xie, Y. Mo, and B. Sinopoli. 2010. False data injection attacks in electricity markets. In Proceedings of the *First IEEE International Conference on Smart Grid Communications.*

H. Yang, B. Jiang, and M. Staroswiecki. 2009. Supervisory fault tolerant control for a class of uncertain nonlinear systems. *Automatica*, 45(10):2319–2324.

I. Yang, D. Kim, and D. Lee. 2010. Fault-tolerant control strategy based on control allocation using smart actuators. In Proceedings of the *Conference on Control and Fault-Tolerant Systems*, SysTol.

G. Zames. 1981. Feedback and optimal sensitivity: Model reference transformations, multiplicative seminorms, and approximate inverses. *IEEE Transactions on Automatic Control*, 26(2):301–320.

Q. Zhang and X. Zhang. 2012. Distributed fault detection and isolation for multimachine power systems. In Proceedings of the *IEEE/ASME International Conference on Mechatronics and Embedded Systems and Applications.*

W. Zhang, M. S. Branicky, and S. M. Phillips. 2001. Stability of networked control systems. *IEEE Control Systems Magazine*, 21:84–99.

W. Zhang, Q. Yang, and Y. Geng. 2009. A survey of anomaly detection methods in networks. In Proceedings of the *International Symposium on Computer Network and Multimedia Technology.*

X. Zhang, M. Polycarpou, and T. Parisini. 2002. A robust detection and isolation scheme for abrupt and incipient faults in nonlinear systems. *IEEE Transactions on Automatic Control*, 47(4):576–593.

Y. Zhang and J. Jiang. 2008. Bibliographical review on reconfigurable fault-tolerant control systems. *Annual Reviews in Control*, 32(2):229–252.

K. Zhou, J. C. Doyle, and K. Glover. 1996. *Robust and Optimal Control.* Prentice-Hall, Inc., Upper Saddle River, NJ, USA.

M. Zhu and S. Martinez. 2012. On resilient consensus against replay attacks in operator-vehicle networks. In Proceedings of the *American Control Conference.*

Q. Zhu and T. Başar. 2012. A dynamic game-theoretic approach to resilient control system design for cascading failures. In Proceedings of the *1st International Conference on High Confidence Networked Systems, CPSWeek.*

R. D. Zimmerman, C. E. Murillo-Sánchez, and R. J. Thomas. 2009. MATPOWER's extensible optimal power flow architecture. In Proceedings of the *IEEE Power and Energy Society General Meeting.*