UNIVERSITY OF CALIFORNIA
Santa Barbara

Secure Control Systems: A Control-Theoretic Approach to Cyber-Physical Security

A Dissertation submitted in partial satisfaction
of the requirements for the degree of

Doctor of Philosophy

in

Mechanical Engineering

by

Fabio Pasqualetti

Committee in Charge:

Professor Francesco Bullo, Chair

Professor Bassam Bamieh

Professor Jeff Moehlis

Professor João Hespanha

September 2012

The Dissertation of
Fabio Pasqualetti is approved:

Professor Bassam Bamieh

Professor Jeff Moehlis

Professor João Hespanha

Professor Francesco Bullo, Committee Chairperson

September 2012

Secure Control Systems: A Control-Theoretic Approach to Cyber-Physical Security

by

Fabio Pasqualetti

Alla mia famiglia, per il consiglio,

il sostegno, e l'approvazione a me offerti.

Acknowledgements

My first and deepest thanks go to my advisor Francesco Bullo:

> "Tu se' lo mio maestro e 'l mio autore,
> tu se' solo colui da cu' io tolsi
> lo bello stilo che m'ha fatto onore."[1]

Francesco has always been very patient, supportive, and encouraging, both from a scientific and a human point of view. His guidance and enthusiasm have allowed me to discover the joy and gratification of scientific research. I am truly grateful to him for making my graduate studies an unforgettable experience.

I am indebted to Antonio Bicchi, who advised me during my studies in Pisa. Antonio is an inspired teacher, whose perseverance and dedication are extraordinary and have been a model for me since we first met. I am thankful to him for his revealing advice, which made it possible for me to start my graduate studies.

I am grateful to my committee members, Bassam Bamieh, Joao Pedro Hespanha, and Jeff Moehlis for our constructive interactions. Their questions and feedback shaped my research and helped me identifying interesting directions.

I would like to thank my collaborators, who helped me complete my research achievements, and encouraged me towards compelling problems. Thank you Antonio Franchi for all your work on multi-agent patrolling and surveillance. Thank

[1] Dante Alighieri, "The Divine Comedy", Inferno, Canto I, 85-87, (1321). Thou art my master, and my author thou, Thou art alone the one from whom I took, The beautiful style that has done honour to me.

you Ruggero Carli for your insight and contribution in distributed monitoring via iterative projections. Thank you Vaibhav Srivastava for our endless discussions about continuous and discrete mathematics, and for allowing me to be part of your stochastic patrolling project. Thank you Joseph W. Durham for teaching me how to implement our patrolling algorithms on real robots. Thank you Markus Spindler, Jeffrey R. Peters, Filippo Zanella, and Domenica Borra for your help in cameras surveillance. A special thanks to Florian Dörfler. Florian not only contributed to many results contained in this thesis, he also encouraged me to strive every time for the very best results that could be achieved. I have been very fortunate to work with you all, and to enjoy your company outside our lab.

My gratitude also goes to the past and present lab members for numerous stimulating discussions, and for organizing our strenuous mountain hikes.

During my years at UCSB I have been lucky to meet several people that soon became my friends. I shared countless enriching and cheerful moments with you, and you made my stay in Santa Barbara very exciting. Thank you Corrado, for the numerous pool games, for introducing me to salsa dancing, and for many insightful conversations. Thank you Luca, for our surfing trips and for our beach volleyball games. Thank you Ludovico, Greg, and Nicolas, for your friendship and help in many occasions. Thank you Tammy and Mary, for trying to smooth my Italian accent, and for our periodic reunions. Thank you Emily, Laura, Lily, and

Lisa for bearing with my dancing skills. Finally, thank you Aina, Fanny, Dimitri, Bahar, Lise, Misael, Joana, Till, Benjamin, Marco, Elisa, Lara, Val, and Katia.

My thanks for my family will never be enough; this achievement and many past ones live on the support, the comforting words, and the care of my dear ones.

Finally thank you Daniela, for your incredible patience, your tireless perseverance, your sweet smile, and for the unique flavor you give to my life.

Curriculum Vitae of Fabio Pasqualetti

Education and Qualifications

Oct 2007 **Laurea Magistrale** (M.Sc. equivalent)
Electrical and Computer Engineering, University of Pisa, Pisa, Italy
Advisor: Antonio Bicchi
Thesis: Distributed Intrusion Detection for Secure Consensus Computations

Jul 2005 **Mittelstufe Zertificat, Deutsch**
Goethe Institut, Pisa, Italy

Aug 2004 **Laurea** (B.Sc. equivalent)
Computer Engineering, University of Pisa, Pisa, Italy

Professional Affiliations

Member Institute for Electrical and Electronics Engineers (IEEE)
IEEE Control Systems Society (IEEE CSS)

Research Interests

1 Distributed control and estimation of large scale systems.
2. Security, reliability, and trust management in cyber-physical systems.
3. Mobile robotics, environmental patrolling, and persistent surveillance.
4. Combinatorial optimization, complexity theory, and algorithms.

Research Experience

Jan 2008 - **Graduate Student Researcher**, Mechanical Engineering, University of California, Santa Barbara

Two key security problems have been investigated, namely (i) *Secure control systems*, and (ii) *Distributed area patrolling and persistent surveillance*. For the fist problem, research focused on understanding and characterizing fundamental vulnerabilities of control systems, including cyber-physical systems, and on developing centralized, decentralized, and distributed monitoring algorithms. For the second problem, research focused on designing open-loop and feedback strategies to optimally coordinate a team of autonomous agents for the patrolling of an environment, as well on characterizing the computational complexity of designing optimal patrolling strategies.

Sep 2006 - Jun 2007 **Visiting Scholar**, Mechanical Engineering, University of California, Santa Barbara

Investigated the applicability of geometric fault detection and isolation techniques to distributed consensus systems

Teaching Experience and Mentoring

2010 - 2011 **Teaching Assistant**
Course: *Control System Design*,
Mechanical Engineering, University of California, Santa Barbara

Sep 2010 - Jun 2011 **Graduate Student Mentor**
Student: Markus Spindler
Thesis: *Distributed Multi-camera Synchronization for Smart-intruder Detection*

Honors and Awards

Jun 2012 **Excellence Fellowship**,
Department of Mechanical Engineering, UCSB, Santa Barbara, USA

Jun 2011 **Best Presentation in Session**,
American Control Conference

Dec 2009 **General Chairs' Recognition Award for Interactive Papers**,
Conference on Decision and Control

Research awards

I have contributed to the writing of the following awards:

1. NSF CPS Medium: The Cyber-Physical Challenges of Transient Stability and Security in Power Grids
2. NSF CPS Medium: Dynamic Routing and Robotic Coordination for Oceanographic Adaptive Sampling

Professional Service

Technical Reviewer

Journal
Automatica
IEEE Transactions on Automatic Control
IEEE Transactions on Robotics
IEEE Transactions on Systems, Man, and Cybernetics, PartB

Conference
American Control Conference
IEEE Conference on Decision and Control
IEEE International Conference on Robotics and Automation
IFAC Workshop on Distributed Estimation and Control in Networked Systems
IFAC World Congress

Co - Organizer

Workshop
2011 Santa Barbara Control Workshop

Invited Workshops and Tutorials

Dec 2011 **Workshop on Control Systems Security: Challenges and Directions**
Conference: IEEE CDC, Orlando, FL

Invited Talks

Feb 2012 Cyber-Physical Systems Laboratory, University of California at Los Angeles

Aug 2010 Interdepartmental Research Center E. Piaggio, University of Pisa

Aug 2010 Department of Information Engineering, University of Padova

Publications

Journal articles

1. D. Borra, F. Pasqualetti, and F. Bullo. Continuous Graph Partitioning for Camera Network Surveillance. *Automatica*, 2012. Submitted.
2. V. Srivastava, F. Pasqualetti, and F. Bullo. Stochastic Surveillance Strategies for Spatial Quickest Detection: Theory and Experiments. *International Journal of Robotics Research*, 2012. Submitted.
3. F. Pasqualetti, F. Dörfler, and F. Bullo. Attack Detection and Identification in Cyber-Physical Systems. *IEEE Transactions on Automatic Control*, 2012. Submitted.
4. F. Dörfler, F. Pasqualetti, and F. Bullo. Continuous-Time Distributed Estimation with Discrete Communication. *Journal of Selected Topics in Signal Processing*, 2012. Submitted.
5. F. Pasqualetti, J. W. Durham, and F. Bullo. Cooperative patrolling via weighted tours: Performance analysis and distributed algorithms. *IEEE Transactions on Robotics*. To appear.
6. F. Pasqualetti, R. Carli, and F. Bullo. Distributed estimation via iterative projections with application to power network monitoring. *Automatica*, May. 2012.
7. F. Pasqualetti, A. Franchi, and F. Bullo. On cooperative patrolling: Optimal trajectories, complexity analysis and approximation algorithms. *IEEE Transactions on Robotics*, Jun. 2012.
8. F. Pasqualetti, A. Bicchi, and F. Bullo. Consensus computation in unreliable networks: A system theoretic approach. *IEEE Transactions on Automatic Control*, Jan. 2012.

Conference articles

1. D. Borra, F. Pasqualetti, and F. Bullo. Continuous Graph Partitioning for Camera Network Surveillance. In *IFAC Workshop on Distributed Estimation and Control in Networked Systems*, Santa Barbara, CA, USA, Sep. 2012..

1. F. Zanella, F. Pasqualetti, and F. Bullo. Simultaneous Boundary Partitioning and Cameras Synchronization for Optimal Video Surveillance. In *IFAC Workshop on Distributed Estimation and Control in Networked Systems*, Santa Barbara, CA, USA, Sep. 2012.

3. F. Pasqualetti, F. Dörfler, and F. Bullo. Cyber-physical security via geometric control: Distributed monitoring and malicious attacks. In *IEEE Conference on Decision and Control*, Maui, HI, USA, Dec. 2012.

4. M. Spindler, F. Pasqualetti, and F. Bullo. Distributed multi-camera synchronization for smart-intruder detection. In *American Control Conference*, Montreal, Canada, Jun. 2012.

5. F. Dörfler, F. Pasqualetti, and F. Bullo. Distributed detection of cyber-physical attacks in power networks: A waveform relaxation approach. In *Allerton Conf. on Communications, Control and Computing*, pages 1486–1491, Monticello, Il, USA, Sep. 2011.

6. F. Pasqualetti, A. Bicchi, and F. Bullo. A graph-theoretical characterization of power network vulnerabilities. In *American Control Conference*, pages 3918–3923, San Francisco, CA, USA, Jun. 2011.

7. F. Pasqualetti, R. Carli, and F. Bullo. A distributed method for state estimation and false data detection in power networks. In *IEEE International Conference on Smart Grid Communications*, Brussels, Belgium, Oct. 2011.

8. F. Pasqualetti, F. Dörfler, and F. Bullo. Cyber-physical attacks in power networks: Models, fundamental limitations and monitor design. In *IEEE Conference on Decision and Control and European Control Conference*, Orlando, FL, USA, Dec. 2011.

9. F. Pasqualetti, R. Carli, A. Bicchi, and F. Bullo. Distributed estimation and detection under local information. In *IFAC Workshop on Distributed Estimation and Control in Networked Systems*, pages 263–268, Annecy, France, Sep. 2010.

10. F. Pasqualetti, R. Carli, A. Bicchi, and F. Bullo. Identifying cyber attacks under local model information. In *IEEE Conference on Decision and Control*, pages 5961–5966, Atlanta, GA, USA, Dec. 2010.

11. F. Pasqualetti, A. Franchi, and F. Bullo. On optimal cooperative patrolling. In *IEEE Conference on Decision and Control*, pages 7153–7158, Atlanta, GA, USA, Dec. 2010.

12. F. Pasqualetti, A. Bicchi, and F. Bullo. On the security of linear consensus networks. In *IEEE Conference on Decision and Control and Chinese Control Conference*, pages 4894–4901, Shanghai, China, Dec. 2009.

13. F. Pasqualetti, S. Martini, and A. Bicchi. Steering a leader-follower team via linear consensus. In *Hybrid Systems: Computation and Control*, Saint Louis, MO, USA, May. 2008.

14. F. Pasqualetti, A. Bicchi, and F. Bullo. Distributed intrusion detection for secure consensus computations. In *IEEE Conference on Decision and Control*, pages 5594–5599, New Orleans, LA, USA, Dec. 2007.

Abstract

Secure Control Systems: A Control-Theoretic Approach to Cyber-Physical Security

Fabio Pasqualetti

Cyber-physical systems and networks form a ubiquitous computing substrate that underlies much of modern technological society. Examples include embedded systems, such as medical devices, communication peripherals, smart vehicles, and large-scale systems, such as transportation networks, power generation grids, and water distribution systems. Researchers and hackers have recently shown that cyber-physical systems are vulnerable to remote attacks targeting their physical infrastructure or their data management and communication layer. Due to the crucial role of cyber-physical systems in everyday life, the development of advanced security monitors is of utmost importance.

This thesis addresses problems concerning security of cyber-physical systems. Our contribution is threefold. First, we propose a unified modeling framework for cyber-physical systems, monitors, and attacks. For our model we define the notions of *detectability* and *identifiability* of an attack by its effect on output measurements, and we characterize fundamental monitoring limitations. Additionally, we provide algebraic and graph-theoretic tests for the existence of undetectable and unidentifiable attacks in cyber-physical systems. Second, we design centralized and distributed monitors for the detection and identification of attacks from output measurements. Our monitors leverage on tools from control theory and distributed computing, such as *conditioned invariant subspaces* and *waveform relaxation techniques*. Our monitors are provably correct, and effective against attacks targeting both the physical infrastructure and the communication layer. Third, we exploit our findings to design undetectable attack strategies. Our attack design method relies upon the control-theoretic notion of *controlled invariant subspace*. Our attack strategy is specific, in the sense that the attack signal is cast to alter the system functionality in a pre-specified manner. Finally, we present several illustrative examples. Besides showing the effectiveness of our methods for the analysis of systems vulnerabilities, the design of security monitors, and the synthesis of attack strategies, our numerical examples confirm that our methods are effective also in the presence of system noise and unmodeled dynamics.

Professor Francesco Bullo
Dissertation Committee Chair

Contents

List of Figures

List of Tables

Chapter 1

Introduction

> *"... fare come gli arcieri prudenti, e quali parendo el loco dove disegnano ferire troppo lontano e conoscendo fino a quanto va la virtù del loro arco, pongono la mira assai più alta che il loco destinato, non per aggiungere con la loro freccia a tanta altezza, ma per potere con l'aiuto di sì alta mira pervenire al disegno loro."*[1]
>
> Niccolò Machiavelli, "*De Principatibus*" (1532)

Cyber-physical systems arise from the tight integration of physical processes, computational resources, and communication capabilities: processing units monitor and control physical processes by means of sensor and actuator networks. Examples of cyber-physical systems include transportation networks, power generation and distribution networks, water and gas distribution networks, and advanced communication systems. Due to the crucial role of cyber-physical systems in everyday life, cyber-physical security needs to be promptly addressed.

[1] ... to act like the clever archers who, designing to hit the mark which yet appears too far distant, and knowing the limits to which the strength of their bow attains, take aim much higher than the mark, not to reach by their strength or arrow to so great a height, but to be able with the aid of so high an aim to hit the mark they wish to reach.

Besides failures and attacks on the physical infrastructure, cyber-physical systems are also prone to cyber attacks on their data management and communication layer. Recent studies and real-world incidents have demonstrated the inability of existing security methods to ensure a safe and reliable functionality of cyber-physical infrastructures against unforeseen failures and, possibly, external attacks [15, 64, 101, 103]. The protection of critical infrastructures is, as of today, one of the main focuses of the Department of Homeland Security [4].

Concerns about security of control systems are not new, as the numerous manuscripts on systems fault detection, isolation, and recovery testify; see for example [7, 28]. Cyber-physical systems, however, suffer from specific vulnerabilities which do not affect classical control systems, and for which appropriate detection and identification techniques need to be developed. For instance, the reliance on communication networks and standard communication protocols to transmit measurements and control packets increases the possibility of intentional and worst-case (cyber) attacks against physical plants. On the other hand, information security methods, such as authentication, access control, message integrity, and cryptography methods, appear inadequate for satisfactory protection of cyber-physical systems. Indeed, these security methods do not exploit the compatibility of the measurements with the underlying physical process and control mechanism, which are the ultimate objective of a protection scheme [16]. Moreover, such information security methods are not effective against insider attacks carried out by authorized entities, as in the famous Maroochy Water Breach case [101], and they also fail against attacks targeting directly the physical dynamics [26].

1.1 Literature Synopsis

In this section we review the existing literature in the area of cyber-physical security, distributed estimation, and secure consensus computation. This will allow for a more concrete statement of the contributions of this thesis.

1.1.1 Cyber-physical security

The analysis of vulnerabilities of cyber-physical systems to external attacks has received increasing attention in the last years. The general approach has been to study the effect of specific attacks against particular systems. For instance, in [2] *deception* and *denial of service* attacks against a networked control system are introduced, and, for the latter ones, a countermeasure based on semi-definite programming is proposed. Deception attacks refer to the possibility of compromising the integrity of control packets or measurements, and they are cast by altering the behavior of sensors and actuators. Denial of service attacks, instead, compromise the availability of resources by, for instance, jamming the communication channel. In [56] *false data* injection attacks against static state estimators are introduced. False data injection attacks are specific deception attacks in the context of static estimators. It is shown that undetectable false data injection attacks can be designed even when the attacker has limited resources. In a similar fashion, *stealthy deception attacks* against the Supervisory Control and Data Acquisition system are studied, among others, in [3, 110]. In [69] the effect of *replay attacks* on a control system is discussed. Replay attacks are cast by hijacking the sensors, recording the readings for a certain amount of time, and repeating such readings while injecting an exogenous signal into the system. It is shown that this type of attack can be detected by injecting a signal unknown to the attacker into the system. In [102] the effect of *covert attacks* against networked control systems is investigated. Specifically, a parameterized decoupling structure allows a covert agent to alter the behavior of the physical plant while remaining undetected from the original controller. In [121] a resilient control problem is studied, in which control packets transmitted over a network are corrupted by a human adversary. A receding-horizon Stackelberg control law is proposed to stabilize the control system despite the attack. Recently the problem of estimating the state of a linear system with corrupted measurements has been studied [40]. More precisely, the maximum number of faulty sensors that can be tolerated is characterized, and a decoding algorithm is proposed to detect corrupted measurements. Finally, security issues of some specific cyber-physical systems have received considerable attention, such as power networks [25,26,64,70,77,80,97,103,110], linear networks with misbehaving components [78,105], and water networks [3,32,101,102].

1.1.2 Distributed estimation and false data detection

Starting from the eighties, the problem of distributed estimation has attracted intense attention from the scientific community, generating through the years a very rich literature. More recently, because of the advent of highly integrated and low-cost wireless devices as key components of large autonomous networks, the interest for this classical topic has been renewed. For a wireless sensor network, novel applications requiring efficient distributed estimation procedures include, for instance, environment monitoring, surveillance, localization, and target tracking. Considerable effort has been devoted to the development of distributed and adaptive filtering schemes, which generalize the notion of adaptive estimation to a setup involving networked sensing and processing devices [18]. In this context, relevant methods include incremental Least Mean-Square [57], incremental Recursive Least-Square [94], Diffusive Least Mean-Square [94], and Diffusive Recursive Least-Square [18]. Diffusion Kalman filtering and smoothing algorithms are proposed, for instance, in [17, 19], and consensus based techniques in [95, 96]. We remark that the strategies proposed in the aforementioned references could be adapted for the solution of our estimation problems. Their performance, however, appears not to be well suited in our context for the following reasons. First, the convergence of the above estimation algorithms is only asymptotic, and it depends upon the communication topology. As a matter of fact, for many communication topologies, such as Cayley graphs and random geometric graphs, the convergence rate is very slow and scales badly with the network dimension. Such slow convergence rate is clearly undesirable because a delayed state estimation could lead the power plant to instability. Second, approaches based on Kalman filtering require the knowledge of the global state and observation model by all the components of the network, and they therefore violate our assumptions. An exception is constituted by [104], where an estimation technique based on local Kalman filters and a consensus strategy is developed. This latter method, however, besides exhibiting asymptotic convergence, does not offer guarantees on the final estimation error. Third and finally, the application of these methods to the detection of cyber attacks, which is also our goal, is not straightforward, especially when detection guarantees are required.

The estimation technique we developed here belongs to the family of Kaczmarz (row-projection) methods for the solution of a linear system of equations. See [20, 38, 44, 108] for a detailed discussion. Differently from the existing row-action methods, our algorithms exhibit finite time convergence, and they can be used to compute the weighted least squares solution to a system of linear equations with arbitrary weights.

1.1.3 Secure consensus computation

Distributed systems and networks have received much attention in the last years because of their flexibility and computational performance. One of the most frequent tasks to be accomplished by autonomous agents is to agree upon some parameters. Agreement variables represent quantities of interest such as the work load in a network of parallel computers, the clock speed for wireless sensor networks, the velocity, the rendezvous point, or the formation pattern for a team of autonomous vehicles; e.g., see [12, 73, 89].

Several algorithms achieving consensus have been proposed and studied in the computer science community [61]. In this work, we consider linear consensus iterations, where, at each time instant, each node updates its state as a weighted combination of its own value and those received from its neighbors [43, 74]. The choice of algorithm weights influences the convergence speed toward the steady state value [118].

Because of the lack of a centralized entity that monitors the activity of the nodes of the network, distributed systems are prone to attacks and component failure, and it is of increasing importance to guarantee trustworthy computation even in the presence of misbehaving parts [66]. Misbehaving agents can interfere with the nominal functions of the network in different ways. In this thesis, we consider two extreme cases: that the deviations from their nominal behavior are due to genuine, random faults in the agents; or that agents can instead craft messages with the purpose of disrupting the network functions. In the first scenario, faulty agents are unaware of the structure and state of the network and ignore the presence of other faults. In the second scenario, the worst-case assumption is made that misbehaving agents have knowledge of the structure and state of the network, and may collude with others to produce the biggest damage. We refer to the first case as non-colluding, or faulty; to the second case as malicious, or Byzantine.

Reaching unanimity in an unreliable system is an important problem, well studied by computer scientists interested in distributed computing. A first characterization of the resilience of distributed systems to malicious attacks appears in [50], where the authors consider the task of agreeing upon a binary message sent by a "Byzantine general," when the communication graph is complete. In [30] the resilience of a partially connected[2] network seeking consensus is analyzed, and

[2] The connectivity of a graph is the maximum number of disjoint paths between any two vertices of the graph. A graph is complete if it has connectivity $n-1$, where n is the number of vertices in the graph.

it is shown that the well-behaving agents of a network can always agree upon a parameter if and only if the number of malicious agents

(i) is less than 1/2 of the network connectivity, and

(ii) is less than 1/3 of the number of processors.

This result has to be regarded as a fundamental limitation of the ability of a distributed consensus system to sustain arbitrary malfunctioning: the presence of misbehaving Byzantine processors can be tolerated only if their number satisfies the above threshold, independently of whatever consensus protocol is adopted.

We consider linear consensus algorithms in which every agent, including the misbehaving ones, are assumed to send the same information to all their neighbors. This assumption appears to be realistic for most control scenarios. In a sensing network for instance, the data used in the consensus protocol consist of the measurements taken directly by the agents, and (noiseless) measurements regarding the same quantity coincide. Also, in a broadcast network, the information is transmitted using broadcast messages, so that the content of a message is the same for all the receiving nodes. The problem of characterizing the resilience properties of linear consensus strategies has been partially addressed in recent works [76, 106, 107], where, for the malicious case, it is shown that, despite the limited abilities of the misbehaving agents, the resilience to external attacks is still limited by the connectivity of the network. In [76] the problem of detecting and identifying misbehaving agents in a linear consensus network is first introduced, and a solution is proposed for the single faulty agent case. In [106, 107], the authors provide one policy that k malicious agents can follow to prevent some of the nodes of a $2k$-connected network from computing the desired function of the initial state, or, equivalently, from reaching an agreement. On the contrary, if the connectivity is $2k + 1$ or more, then the authors show that generically the set of misbehaving nodes is identified independent of its behavior, so that the desired consensus is eventually reached.

1.2 Contributions of this Thesis

The main contributions of each chapter are as follows.

Chapter 2 In this chapter we introduce the notation, some preliminary definitions, and some important results about control theory, algebraic graph theory, and distributed computing. These notions will be intensively applied in the subsequent chapters for our analysis.

Chapter 3 In this chapter we formally model power networks, water networks, and sensor networks. These prototypical examples of cyber-physical systems will be used in the subsequent chapters to illustrate our findings and techniques.

Chapter 4 The contributions of this chapter are threefold. First, we describe a unified modeling framework for cyber-physical systems and attacks. Motivated by existing cyber-physical systems and proposed attack scenarios, we model a cyber-physical system under attack as a descriptor system subject to unknown inputs affecting the state and the measurements. For our model, we define the notions of *detectability* and *identifiability* of an attack by its effect on output measurements. Our framework is general, and it includes the scenarios in [2,25,56,68,69,102,110] as special cases. Second, we characterize fundamental limitations of static, dynamic, and active detection and identification procedures. Specifically, we show that static detection procedures are unable to detect any attack affecting the dynamics, and that attacks corrupting the measurements can be easily designed to be undetectable. On the contrary, we show that undetectability in a dynamic setting is much harder to achieve for an attacker: a cyber-physical attack is undetectable if and only if the attackers' signal excites uniquely the zero dynamics of the input/output system. Additionally, we show that active monitors capable of injecting test signals are as powerful as dynamic (passive) monitors, since an attacker can design undetectable and unidentifiable attacks without knowing the signal injected by the monitor into the system. Third, we provide a graph theoretic characterization of undetectable attacks. Specifically, we borrow some tools from the theory of structured systems, and we identify conditions on the system interconnection structure for the existence of undetectable attacks. These conditions are *generic*, in the sense that they hold for almost all numerical systems with the same structure, and they can be efficiently verified. Finally, we illustrate the potential impact of our theoretical findings through compelling examples.

Chapter 5 The contributions of this chapter are threefold. First, we adopt the static state network estimation model, in which the state vector is linearly related to the network measurements. We develop two methods for a group of interconnected control centers to compute an optimal estimate of the system state via distributed computation. Our first estimation algorithm assumes an *incremental* mode of cooperation among the control centers, while our second estimation algorithm is based upon a *diffusive* strategy. Both methods are shown to converge in a finite number of iterations, and to require only local information for their implementation. Differently from [86], our estimation procedures assume neither the measurement error covariance nor the measurements matrix to be diagonal. Furthermore, our algorithms are advantageous from a communication perspective, since they reduce the distance between remote terminal units and the associated control center, and from a computational perspective, since they distribute the measurements to be processed among the control centers. Second, we describe a finite-time algorithm to detect via distributed computation if the measurements have been corrupted by a malignant agent. Our detection method is based upon our state estimation technique, and it inherits its convergence properties. Notice that, since we assume the measurements to be corrupted by noise, the possibility exists for an attacker to compromise the network measurements while remaining undetected (by injecting for instance a vector with the same noise statistics). With respect to this limitation, we characterize the class of corrupted vectors that are guaranteed to be detected by our procedure, and we show optimality with respect to a centralized detection algorithm. Third, we study the scalability of our methods in networks of increasing dimension, and we derive a finite-memory approximation of our diffusive estimation strategy. For this approximation procedure we show that, under a reasonable set of assumptions and independently of the network dimension, each control center is able to recover a good approximation of the state of a certain subnetwork through little computation. Moreover, we provide bounds on the approximation error for each subnetwork. Finally, we illustrate the effectiveness of our procedures on the IEEE 118 bus system.

Chapter 6 The main contributions of this chapter are as follows. First, for our differential-algebraic model of cyber-physical systems under attacks developed in Chapter 4, we design centralized monitors for attack detection and identification. With respect to the existing solutions, in this thesis we propose attack detection and identification filters that are effective for both state and output attacks against linear continuous-time differential-algebraic cyber-physical systems. Our monitors are designed by using tools from geometric control theory; they extend the construction of [62] to descriptor systems with direct feedthrough matrix, and they are guaranteed to achieve optimal performance, in the sense that they detect (respectively identify) every detectable (respectively identifiable) attack. Second, we develop a fully distributed attack detection filter with optimal (centralized) performance. Specifically, we provide a distributed implementation of our centralized attack detection filter based upon iterative local computations by using the Gauss-Jacobi waveform relaxation technique. For the implementation of this method, we rely upon cooperation among geographically deployed control centers, each one responsible for a part of the system. In particular, we require each control center to have access to the measurements of its local subsystem, synchronous communication among neighboring control centers at discrete time instants, and ability to perform numerical integration. Third, we show that the attack identification problem is inherently computationally hard. Consequently, we design a distributed identification method that achieves identification, at a low computational cost and for a class of attacks, which can be characterized accurately. Our distributed identification methods is based upon a *divide and conquer* procedure, in which first corrupted regions and then corrupted components are identified by means of local identification procedures and cooperation among neighboring regions. Due to cooperation, our distributed procedure provably improves upon the fully decoupled approach advocated in decentralized control [93]. Fourth, we present several illustrative examples, which show the robustness of our methods in the presence of system noise, nonlinearities, and modeling uncertainties.

Chapter 7 In this chapter we exploit our previous findings to cast malicious attacks. In particular, we use the geometric notion of *controlled invariant subspace* to design attack signals, which are undetectable at some observing stations. We illustrate this technique in a competitive power generation scenario [26], in which a coalition of generators aim to destabilize other machines in the network.

Chapter 8 In this chapter we study linear consensus networks with misbehaving agents. By recasting the problem of linear consensus computation in an unreliable system into a system theoretic framework, we provide alternative and constructive system-theoretic proofs of existing bounds on the number of identifiable misbehaving agents in a linear network, i.e., k Byzantine agents can be detected and identified if the network is $(2k+1)$-connected, and they cannot be identified if the network is $2k$-connected or less. Moreover, we exhaustively describe the strategies that misbehaving nodes can follow to disrupt a linear network that is not sufficiently connected. We provide a novel and comprehensive analysis on the detection and identification of non-colluding agents. We show that k faulty agents can be identified if the network is $(k+1)$-connected, and cannot if the network is k-connected or less. For both the cases of Byzantine and non-colluding agents, we prove that the proposed bounds are generic with respect to the network communication weights, i.e., given an (unweighted) consensus graph, the bounds hold for almost all (consensus) choices of the communication weights. In other words, if we are given a $(k + 1)$-connected consensus network for which k faulty agents cannot be identified, then a random and arbitrary small change of the communication weights (within the space of consensus weights) make the misbehaving agents identifiable with probability one. In the last part of this chapter, we discuss the problem of detecting and identifying misbehaving agents when either the partial knowledge of the network or hardware limitations make it impossible to implement an exact identification procedure. We introduce a notion of network decentralization in terms of relatively weakly connected subnetworks. We derive a sufficient condition on the consensus matrix that allows one to identify a certain class of misbehaving agents under local network model information. Finally, we describe a local algorithm to promptly detect and identify corrupted components.

Chapter 9 This chapter concludes the thesis and discuss some aspects for future research in the area of secure control systems and secure distributed computing.

Chapter 2

Preliminaries in Control Theory, Algebraic Graph Theory, and Distributed Computing

In this chapter we review some basic results in linear dynamical systems, graph theory, and distributed computing. The notation introduced in this chapter will be used consistently throughout the remaining chapters.

2.1 Control Theory and Graph Definitions

2.1.1 Linear dynamical systems

Let $\mathbb{R}$, $\mathbb{R}_{\geq 0}$, $\mathbb{C}$, and $\mathbb{N}$ denote the set of real numbers, the set of non-negative real numbers, the set of complex numbers, and the set of positive integers, respectively. A *continuous time invariant system* is defined by the equations

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

where $x : \mathbb{R}_{\geq 0} \to \mathbb{R}^n$, $u : \mathbb{R}_{\geq 0} \to \mathbb{R}^m$, $y : \mathbb{R}_{\geq 0} \to \mathbb{R}^p$, and A, B, C, and D are constant matrices of appropriate dimensions. The signals x, u, and y are called the *state*, *input*, and *output* of the system, respectively. Analogously, a *discrete time invariant system* is defined by the equations

$$\begin{aligned} x(t+1) &= Ax(t) + Bu(t), \\ y(t) &= Cx(t) + Du(t), \end{aligned} \tag{2.2}$$

where, differently than the continuous time case, the domain of signals is $\mathbb{N}$ instead of $\mathbb{R}_{\geq 0}$. Equations (2.1) express an *input-output* relationship between the input u and the output y. In particular, for $t \in \mathbb{R}_{\geq 0}$ it holds

$$\begin{aligned} x(t) &= \exp^{At} x(0) + \int_0^t \exp^{A(t-\tau)} Bu(\tau) d\tau, \\ y(t) &= C \exp^{At} x(0) + \int_0^t C \exp^{A(t-\tau)} Bu(\tau) d\tau, \end{aligned}$$

where $x(0)$ is referred to as the system *initial condition* or *initial state*, and $\exp^A$ denotes the matrix exponential function on A. Analogously, for the system (2.2) and for $t \in \mathbb{N}$ it holds

$$
\begin{aligned}
x(t) &= A^t x(0) + \sum_{\tau=0}^{t-1} A^{t-\tau} Bu(\tau),\\
y(t) &= CA^t x(0) + \sum_{\tau=0}^{t-1} CA^{t-\tau} Bu(\tau).
\end{aligned}
$$

The input-output relation for the systems (2.1) can be equivalently written in the Laplace domain. In particular,

$$
\begin{aligned}
X(s) &= (sI - A)^{-1} BU(s),\\
Y(s) &= C(sI - A)^{-1} BU(s),
\end{aligned}
$$

where $X(s) = \mathscr{L}(x(t))$, $U(s) = \mathscr{L}(u(t))$, and $Y(s) = \mathscr{L}(y(t))$. Analogous expressions are obtained for the discrete time case by using the Z-transform.

2.1.2 Basic linear algebra definitions

In the field of geometric control theory for linear dynamical systems, system properties are expressed in terms of subspaces. Likewise, analysis and synthesis algorithms rely on operations on subspaces, such as sum, intersection, and orthogonal complement. We will only be dealing with finite-dimensional spaces. Let A be a matrix describing a linear map between two subspaces $\mathcal{X}$ and $\mathcal{Y}$, i.e., $A : \mathcal{X} \to \mathcal{Y}$. The *kernel* or *null space* of A is defined as

$$
\mathrm{Ker}(A) := \{x \in \mathcal{X} \ : \ Ax = 0\},
$$

and the *image* or *range space* of A is defined as

$$\mathrm{Im}(A) := \{Ax \; : \; x \in \mathcal{X}\}.$$

We say that A is surjective if $\mathrm{Im}(A) = \mathcal{Y}$ and injective if $\mathrm{Ker}(A) = 0$. Also, the map A is called bijective (or invertible) if A is injective and surjective. In this case, the map A has an inverse map, usually denoted by A^{-1}. In general, if A is a (not necessarily invertible) linear map and if $\mathcal{V}$ is a subspace of $\mathcal{Y}$, then the inverse image of $\mathcal{V}$ through A is the subspace of X defined by

$$A^{-1}\mathcal{V} := \{x \in \mathcal{X} \; : \; Ax \in \mathcal{V}\}.$$

A subspace $\mathcal{V} \subseteq \mathcal{X}$ is A*-invariant* if $A\mathcal{V} \subseteq \mathcal{V}$. A matrix V is a *basis* of a subspace $\mathcal{V}$ if the columns of V span the subspace $\mathcal{V}$, i.e., if $\mathrm{Im}(V) = \mathcal{V}$. Let $\mathcal{V}$ be A-*invariant*, let $T = [V \, \bar{V}]$, with $\mathrm{Im}(V) = \mathcal{V}$, and $\bar{V}$ such that T is invertible. Then, because of the invariance of $\mathcal{V}$, the *change of coordinates* via T yields

$$T^{-1}AT = \begin{bmatrix} A_{11} & A_{12} \\ 0 & A_{22} \end{bmatrix}.$$

The matrix A_{11} is also denoted as $A \,|\, \mathcal{V}$, while the matrix A_{22} as $A \,|\, \mathcal{X} \setminus \mathcal{V}$. For a matrix A, $\lambda \in \mathbb{C}$ is an *eigenvalue* of A if there exists a nonzero vector $v \in \mathcal{X}$ such that $Av = \lambda v$. The set of eigenvalues, which contains at most n elements, is called the *spectrum* of A and is denoted by $\sigma(A)$. The matrix A is *Hurwitz stable* if $\mathrm{Real}(\lambda) < 0$ for all $\lambda \in \sigma(A)$; the matrix A is *Schur stable* if $|\lambda| < 1$ for all $\lambda \in \sigma(A)$. An invariant subspace $\mathcal{V}$ is called *internally Hurwitz stable* (resp. *internally Schur stable*) if the matrix $A\,|\,\mathcal{V}$ is Hurwitz stable (resp. Schur stable). An invariant subspace $\mathcal{V}$ is called *externally Hurwitz stable* (resp. *externally Schur stable*) if the matrix $A\,|\,\mathcal{X} \setminus \mathcal{V}$ is Hurwitz stable (resp. Schur stable).

2.1.3 Controlled and conditioned invariant subspaces

Consider the time invariant system described by the matrices (A, B, C) $(D = 0)$, where $A : \mathbb{R}^n \to \mathbb{R}^n$, $B : \mathbb{R}^m \to \mathbb{R}^n$, $C : \mathbb{R}^n \to \mathbb{R}^p$, and $D : \mathbb{R}^m \to \mathbb{R}^p$. A subspace $\mathcal{V} \subseteq \mathbb{R}^n$ is an $(A, \mathrm{Im}(B))$-*controlled invariant* subspace [6, Chapter 4] if

$$A\mathcal{V} \subseteq \mathcal{V} + \mathrm{Im}\, B,$$

or, equivalently, if there exists a matrix F such that

$$(A + BF)\mathcal{V} \subseteq \mathcal{V}.$$

The notion of controlled invariant subspace refers to the possibility of confining the state trajectory of the system (A, B, C) within a subspace. Specifically, a subspace $\mathcal{V} \subseteq \mathbb{R}^{n\times n}$ is an $(A, \mathcal{B})$-controlled invariant if, for every initial state $x(0) \in \mathcal{V}$, there exists a control input u such that $x(t) \in \mathcal{V}$ at all times $t \in\ \mathbb{R}_{\geq 0}$. For instance, the controllability subspace $\mathrm{Im}([B\, AB\, \cdots A^{n-1}B])$ is an $(A, \mathcal{B})$-controlled invariant subspace. The set of controlled invariant subspaces contained in $\mathcal{E} \subseteq \mathbb{R}^{n\times n}$ admits a supremum $\mathcal{V}^*$, i.e., an $(A, \mathrm{Im}(B))$-controlled invariant subspace satisfying $\mathcal{V} \subseteq$

$\mathcal{V}^* \subseteq \mathcal{E}$, for any $(A, \text{Im}(B))$-controlled invariant subspace $\mathcal{V}$. If $\mathcal{E} = \text{Ker}(C)$, then $\mathcal{V}^*$ contains all the state trajectories driven by the input u and resulting in the output y being identically zero. In the case $D \neq 0$, the largest controlled invariant subspace $\mathcal{V}^*$ contained in $\text{Ker}(C)$ also needs to satisfy $C\mathcal{V} \subseteq \text{Im}(D)$.

A subspace $\mathcal{S} \subseteq \mathbb{R}^n$ is an $(A, \text{Ker}(C))$-*conditioned invariant* subspace [6, Chapter 4] if

$$A(\mathcal{S} \cap \text{Ker}(C)) \subseteq \mathcal{S},$$

or, equivalently, if there exists a matrix G such that

$$(A + GC)\mathcal{S} \subseteq \mathcal{S},$$

Condition invariant subspaces arise in the context of state estimation. Specifically, the subspace $\mathcal{S}$ is an $(A, \mathcal{C})$-conditioned invariant if it is possible to estimate the trajectory $x \setminus \mathcal{S}$ by processing the initial condition $x(0) \setminus \mathcal{S}$, the input u and the measurements y through an observer [112, Chapter 5]. For instance, the unobservability subspace $\text{Ker}([C^\mathsf{T}\ A^\mathsf{T}C^\mathsf{T} \cdots (A^{n-1})^\mathsf{T}C^\mathsf{T}]^\mathsf{T})$ is an $(A, \text{Ker}(C))$-conditioned invariant subspace. The set of conditioned invariant subspaces containing $\mathcal{E} \subseteq \mathbb{R}^{n \times n}$ admits an infimum $\mathcal{S}^*$, i.e., an $(A, \text{Ker}(C))$-conditioned invariant subspace satisfying $\mathcal{E} \subseteq \mathcal{S}^* \subseteq \mathcal{S}$, for any $(A, \text{Ker}(C))$-conditioned invariant subspace $\mathcal{S}$. If $\mathcal{E} = \text{Im}(B)$, then $\mathcal{S}^*$ defines the largest subspace of the state space that can be estimated in the presence of an unknown input signal u. In the case $D \neq 0$, the smallest conditioned invariant subspace containing $\mathrm{Im}(B)$ also needs to satisfy $\mathrm{Ker}(D) \subseteq B^{-1}\mathcal{S}$.

2.1.4 Invariant zeros, zero dynamics, and left-invertibility

Consider the time invariant system described by the matrices (A, B, C, D), and the associated Rosenbrock matrix

$$P(s) = \begin{bmatrix} sI - A & -B \\ C & D \end{bmatrix}.$$

The invariant zeros of (A, B, C, D) are the complex values $z \in \mathbb{C}$ satisfying $\mathrm{Rank}(P(z)) < \max_{s \in \mathbb{C}} \mathrm{Rank}(P(s))$. Let z be an invariant zeros, and let x_0 and u_0 be such that $(zI - A)x_0 - Bu_0 = 0$ and $Cx_0 + Du_0 = 0$. Then, x_0 and u_0 are referred to as *state-zero direction* and *input-zero direction*. Moreover, z, x_0, and u_0 can be used to generate a state trajectory x yielding $y(t) = 0$ at all times t. For instance, in the case of continuous time systems, the system (A, B, C, D) with input $t \to e^{zt}u_0$ and initial state x_0 yields the state trajectory $x(t) = \exp^{zt} x_0$, with $t \in \mathbb{R}_{\geq 0}$, and output $y(t) = 0$ at all times. The trajectory x is called *zero dynamics*. A system is *left-invertible* if it has a finite number of invariant zeros.

2.1.5 Geometric fault detection and isolation

In the classical Fault Detection and Isolation (FDI) setup, the presence of sensor failures and actuator malfunctions is modeled by adding unknown and unmeasurable inputs $f_i : \mathbb{R}_{\geq 0} \to \mathbb{R}^{m_i}$ to the nominal system. Hence we have

$$\begin{aligned} \dot{x}(t) &= Ax(t) + \sum_{i=1}^{k} B_i f_i(t), \\ y(t) &= Cx(t), \end{aligned}$$

where $k \in \mathbb{N}$ denotes the number of possible actuators and sensors failures, and $B_i \in \mathbb{R}^{n \times m_i}$, $i \in \{1, \dots, k\}$, are known matrices reflecting the failures input directions. The FDI problem is to design, for each failure i, a filter of the form

$$\begin{aligned} \dot{w}_i(t) &= F_i w_i(t) + E_i y(t), \\ r_i(t) &= M_i w(t) + H_i y(t), \end{aligned} \tag{2.3}$$

also known as residual generator, that takes the observables y (and the known input u if present) and generates a residual vector r_i that allows to uniquely identify if f_i becomes nonzero, i.e., if the failure i occurred in the system. As a result of [6, 62], the i-th failure can be correctly identified if and only if

$$\mathrm{Im}(B_i) \cap (\mathcal{V}^*_{K\setminus\{i\}} + \mathcal{S}^*_{K\setminus\{i\}}) = \emptyset,$$

where $\mathcal{V}^*_{K\setminus\{i\}}$ and $\mathcal{S}^*_{K\setminus\{i\}}$ are the maximal controlled and minimal conditioned invariant subspaces associated with the system $(A, [B_1 \cdots B_{i-1} \;\; B_{i+1} \cdots B_k], C)$.

2.1.6 Basic graph definitions

A directed graph $G = (\mathcal{V}_G, \mathcal{E}_G)$ consists of a set of vertices $\mathcal{V}_G$ and a set of directed edges $\mathcal{E}_G \subseteq \mathcal{V}_G \times \mathcal{V}_G$. An edge $(v, w) \in \mathcal{E}_G$ is directed from vertex v to vertex w. A subgraph of a graph $G = (\mathcal{V}_G, \mathcal{E}_G)$ is a graph $H = (\mathcal{V}_H, \mathcal{E}_H)$ such that $\mathcal{V}_H \subseteq \mathcal{V}_G$ and $\mathcal{E}_H \subseteq \mathcal{E}_G$. A graph is undirected if $(v, w) \in \mathcal{E}_G$ implies that $(w, v) \in \mathcal{E}_G$, and in this case we write $\{v, w\} \in \mathcal{E}_G$. For a vertex $v \in \mathcal{V}_G$, the set of in-neighbors of v is defined as $\mathcal{N}_v^{\text{in}} = \{w \in \mathcal{V}_G \ : \ (w, v) \in \mathcal{E}_G\}$, and the set of out-neighbors as $\mathcal{N}_v^{\text{out}} = \{w \in \mathcal{V}_G \ : \ (v, w) \in \mathcal{E}_G\}$. The in-degree of $v \in \mathcal{V}_G$ equals $|\mathcal{N}_v^{\text{in}}|$, whereas the out-degree of $v \in \mathcal{V}_G$ equals $|\mathcal{N}_v^{\text{out}}|$. The complete graph is an undirected graph G such that for every $u, v \in V(G)$, it holds $u, v \in E(G)$.

A path in G is a subgraph $P = (\{v_1, \ldots, v_{k+1}\}, \{e_1, \ldots, e_k\})$ such that $v_i \neq v_j$ for all $i \neq j$, and $e_i = (v_i, v_{i+1})$ for each $i \in \{1, \ldots, k\}$. We say that the path starts at v_1 and ends at v_{k+1}, and at times we will simply identify a path by its vertex sequence $v_1, \ldots, v_{k+1}$. A cycle or closed path is a path in which the first and last vertex in the sequence are the same, i.e., $v_1 = v_{k+1}$. A graph G is acyclic if it contains no cycles. A weighted graph is a graph G in which each edge $(v, w) \in \mathcal{E}_G$ is associated with the weight $z_{vw} \in \mathbb{R}$.

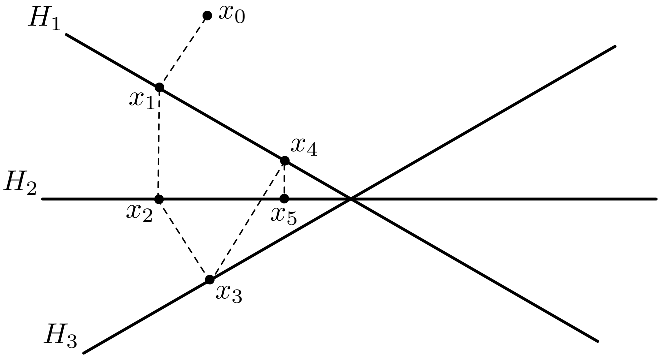


Figure 2.1: Illustration of the Kaczmarz method in iteration (2.4).

2.2 Distributed Computing

2.2.1 The Kaczmarz method

The Kaczmarz method is an iterative algorithm for solving systems of linear equations of the form $Ax = b$, with $A \in \mathbb{R}^{n\times m}$ and $b \in \mathbb{R}^n$ [44]. Assume that the system $Ax = b$ is consistent ($b \in \mathrm{Im}(A)$). Then, the solution to $Ax = b$ is computed as the limit of the iteration

$$x^{(k+1)} = x^{(k)} + \frac{b_i - a_i^\mathsf{T} x^{(k)}}{\|a_i\|_2^2} a_i, \quad k = \{0, 1, \dots\}, \tag{2.4}$$

where $x^{(0)} \in \mathbb{R}^m$ is an arbitrary vector, $i = \mathrm{mod}(k, n) + 1$, a_i^T denotes the i-th row of A, and b_i denotes the i-th component of b. The iteration (2.4) has a geometric interpretation which is next illustrated. Given $x^{(k)}$ and the hyperplane $H_{i_k} = \{x \in \mathbb{R}^m \ : \ a_{i_k}^\mathsf{T} x = b_{i_k}\}$, with $i_{i_k} = \mathrm{mod}(k, n) + 1$, the vector $x^{(k+1)}$ is the orthogonal projection of $x^{(k)}$ onto H_{i_k} (see Fig. 2.1).

In the case the system $Ax = b$ is consistent and admits several solutions, then the Kaczmarz iteration (2.4) converges to the minimum norm solution $\hat{x}$ to $Ax = b$, provided that $x(0) \in \mathrm{Im}(A^\mathsf{T})$. In other words, if $x(0) \in \mathrm{Im}(A^\mathsf{T})$, then

$$\lim_{k \to \infty} x^{(k)} = \hat{x},$$

with $A\hat{x} = b$ and $\hat{x} \perp \mathrm{Ker}(A)$.

2.2.2 The Jacobi method for linear systems

The Jacobi method is an iterative algorithm for solving systems of linear equations of the form $Ax = b$, with $A \in \mathbb{R}^{n \times n}$ and $b \in \mathbb{R}^n$. At each iteration k, the i-th component of the unknown vector is updated as

$$x_i^{(k+1)} = \frac{1}{a_{ii}} \left(b_i - \sum_{j \neq i} a_{ij} x_j^{(k)} \right), \ i \in \{1, \dots, (\}n),$$

where a_{ij} is the (i, j) entry of A, and $x(0)$ is an arbitrary initial vector. Let $A = D + R$, where D is a diagonal matrix with diagonal elements $\{a_{11}, \dots, a_{nn}\}$. The Jacobi iteration in vector form reads as

$$x^{(k+1)} = -D^{-1} R x^{(k)} + D^{-1} b.$$

Hence, the Jacobi iteration is convergent if and only if $\rho(D^{-1}R) < 1$, that is, if and only if the matrix $D^{-1}R$ is Schur stable.

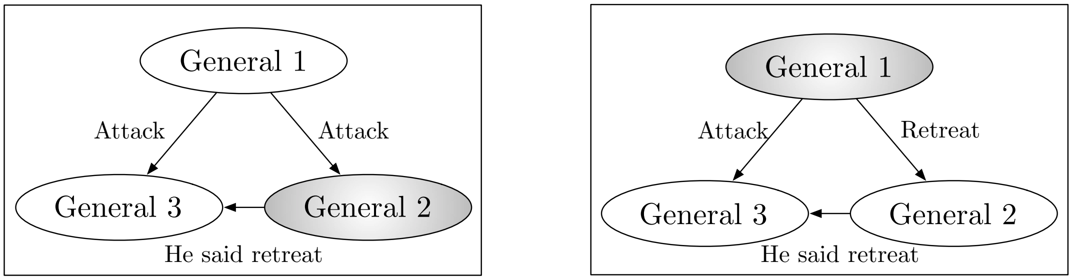


Figure 2.2: Generals can agree on a common plan only if more than two-thirds of the generals are loyal.

2.2.3 The Byzantine Generals problem

Any distributed system relies on the assumption that each single entity involved in the computation behaves as expected. It is often the case, however, that components fail, either due to genuine malfunction, or due to malicious tampering. An important problem studied first by computer scientists is to guarantee trustworthy computation in the face of components misbehaviors.

At a more abstract level, this situation can be expressed in terms of a group of generals of the *Byzantine* army camped with their troops around an enemy city [30, 50]. Generals communicate only via messages exchanged through a communication network G, and they aim at agreeing upon a common battle plan. The assumption is made that one or more of them may be traitors who try to confuse the others. Despite its simplicity, the Byzantine Generals problem has important implications in the field of secure distributed computing.

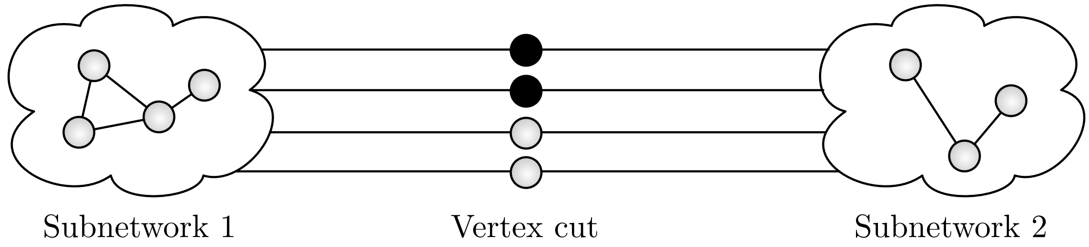


Figure 2.3: Generals can agree on a common plan only if the traitors are less than one-half of the connectivity of the communication network. The black nodes represent traitor generals.

A simple counterexample shows that generals can agree (through some algorithm) on a common plan *only if* more than two-thirds of generals are loyal. To see this, consider the scenarios depicted in Fig. 2.2. Notice that General 3 cannot decide from the received messages whether the General 1 is loyal, and, consequently, it cannot decide upon a battle plan.

Let G be the communication graph among the generals. Recall that the *connectivity* of a graph equals the cardinality of the smallest set of nodes whose removal disconnect the graph. Such set of vertices is called *vertex cut*. Then, an agreement among the generals can be achieved *only if* the number of traitors is less than one-half of the connectivity of the communication graph. To see this, consider now the scenario in Fig. 2.3, where the graph connectivity is four, and where half nodes in a vertex cut represent traitor generals (black nodes). Suppose that the generals in Subnetwork 1 agree on "attack", and that all the traitors deliver the message “retreat” to the generals in Subnetwork 2. Notice that the generals in Subnetwork 2 cannot decide which general on the vertex cut is loyal, and hence cannot agree on the correct battle plan.

In summary, the loyal generals can agree on a common plan only if (i) more than two-thirds of generals are loyal, and (ii) the number of traitors is less than one-half of the connectivity of the communication graph. This result has to be regarded as a fundamental limitation of the ability of a distributed consensus system to sustain arbitrary malfunctioning: the presence of misbehaving Byzantine processors can be tolerated only if their number satisfies the above threshold, independently of whatever agreement protocol is adopted. Finally, the references [30, 50] contain agreement algorithms with optimal resilience to Byzantine failures.

Chapter 3

Examples of Cyber-Physical Systems

In this chapter we describe cyber-physical systems requiring advanced security mechanisms, namely power networks, water networks, and sensor networks. The mathematical models described in this chapter, which will subsequently be used to illustrate our findings, neglect system nonlinearities and the presence of noise in the dynamics and the measurements. Nevertheless, such simplified models have long proven useful in studying stability, faults, and attacks in power networks, water networks, and sensor networks among others. It is our premise that more detailed models are unlikely to change the basic conclusions of this thesis.

3.1 Power Networks

Future power grids will combine physical dynamics with a sophisticated coordination infrastructure. The cyber-physical security of the grid has been identified as an issue of primary concern [64, 103], which has recently attracted the interest of the control and power systems communities, see [25, 70, 77, 80, 97, 110].

During the last decades, a big effort has been devoted to the modeling of the dynamic behavior of a power network, e.g., see [47]. In this thesis, we consider a classical linearized version of the swing model, which we now briefly derive. Consider a connected power network with n generators and $m > n$ buses indexed by $g_1, \dots, g_n$ and $b_1, \dots, b_m$, respectively. Let $b_1, \dots, b_n$ be the generator terminal buses, each one connected to exactly one generator, and let $b_{n+1}, \dots, b_m$ be the load buses. As usual in transient stability studies, the generator dynamics are given by the transient constant-voltage behind reactance model. With the i-th machine, we associate the the voltage modulus E_i, the rotor angle δ_i, the inertia M_i, the damping coefficient D_i, the transient reactance z_i, and the mechanical power input $P_{\mathrm{g},i}$. With the i-th bus we associate the voltage modulus V_i, the phase angle θ_i, the active and the reactive power demands P_i and Q_i, respectively.

With the above notation, the i-th generator dynamics, $i = 1, \ldots, n$, become

$$\begin{aligned} \dot{\delta}_i(t) &= \omega_i(t), \\ M_i \dot{\omega}_i(t) &= P_{\mathrm{g},i}(t) - \frac{E_i V_i}{z_i} \sin(\delta_i(t) - \theta_i(t)) - D_i \omega_i(t). \end{aligned} \tag{3.1}$$

We adopt a ZP load model for every bus, and we denote with G_{ij} and B_{ij} the conductance and susceptance of the transmission line $\{b_i, b_j\}$ [47]. Then, for $k = 1, \ldots, n$ the power flow equation at the k-th generator terminal bus is[1]

$$\begin{aligned} P_k =& \frac{E_k V_k}{z_k} \sin(\theta_k - \delta_k) + \sum_{j=1, j\neq k}^{m} V_k V_j B_{kj} \sin(\theta_k - \theta_j) \\ &+ V_k^2 G_{kk} + \sum_{j=1, j\neq k}^{m} V_k V_j G_{kj} \cos(\theta_k - \theta_j), \\ Q_k =& - \frac{E_k V_k}{z_k} \cos(\theta_k - \delta_k) + \sum_{j=1, j\neq k}^{m} V_k V_j G_{kj} \sin(\theta_k - \theta_j) \\ &- V_k^2 B_{kk} - \sum_{j=1, j\neq k}^{m} V_k V_j B_{kj} \cos(\theta_k - \theta_j) - \frac{1}{x_{di}} V_k^2. \end{aligned} \tag{3.2}$$

Analogously, for $k = n+1, \ldots, m$, the power flow equation at the k-th load bus is

$$\begin{aligned} P_k &= \sum_{j=1, j\neq k}^{m} V_k V_j B_{kj} \sin(\theta_k - \theta_j) + V_k^2 G_{kk} + \sum_{j=1, j\neq k}^{m} V_k V_j G_{kj} \cos(\theta_k - \theta_j), \\ Q_k &= \sum_{j=1, j\neq k}^{m} V_k V_j G_{kj} \sin(\theta_k - \theta_j) - V_k^2 B_{kk} - \sum_{j=1, j\neq k}^{m} V_k V_j B_{kj} \cos(\theta_k - \theta_j). \end{aligned} \tag{3.3}$$

A linear small signal model can be derived from the nonlinear model (3.1) - (3.3) under the usual assumptions that all angular differences are small, that the network is lossless, and that the voltages are close to their nominal rated value. In other words, the assumption is made that for all generators g_i and all pairs of

[1] For brevity, the dependence of the variables on the time t is here omitted.

buses b_j, b_k it holds $|\delta_i - \theta_j| \ll 1$, $|\theta_j - \theta_k| \ll 1$, $G_{jk} = 0$, and $E_i = V_i = 1$. With these assumptions, linearization of equations (3.1) - (3.3) about the (synchronized) network steady state condition yields the dynamic linearized swing equation and the algebraic DC power flow equation[2]

$$\underbrace{\begin{bmatrix} I & 0 & 0 \\ 0 & M & 0 \\ 0 & 0 & 0 \end{bmatrix}}_{E} \begin{bmatrix} \dot{\delta}(t) \\ \dot{\omega}(t) \\ \dot{\theta}(t) \end{bmatrix} = - \underbrace{\begin{bmatrix} 0 & -I & 0 \\ L_{\text{gg}} & D & L_{\text{gl}} \\ L_{\text{lg}} & 0 & L_{\text{ll}} \end{bmatrix}}_{A} \begin{bmatrix} \delta(t) \\ \omega(t) \\ \theta(t) \end{bmatrix}$$

$$+ \underbrace{[\mathbf{0}_n^{\mathsf{T}}, P_{\text{g},1}, \ldots, P_{\text{g},n}, P_1, \ldots, P_m]^{\mathsf{T}}}_{P(t)}, \quad (3.4)$$

where $M = \text{diag}(M_1, \ldots, M_n)$ and $D = \text{diag}(D_1, \ldots, D_n)$. By letting $x = [\delta^{\mathsf{T}} \ \omega^{\mathsf{T}} \ \theta^{\mathsf{T}}]^{\mathsf{T}}$, the model (3.4) can be written as the linear continuous-time descriptor system

$$E\dot{x}(t) = Ax(t) + P(t). \quad (3.5)$$

As a result of the above simplifying assumptions, the matrix $L = \left[\begin{smallmatrix} L_{\text{gg}} & L_{\text{gl}} \\ L_{\text{lg}} & L_{\text{ll}} \end{smallmatrix}\right] \in \mathbb{R}^{(n+m)\times(n+m)}$ is a Laplacian matrix, L_{gg} is diagonal, L_{ll} is invertible, and $L_{\text{lg}} = L_{\text{gl}}^{\mathsf{T}}$.

[2] After linearization, the reactive power equations become independent of the variations of the voltage angles.

3.2 Mass Transport Networks

Mass transport networks, such as gas transmission and distribution networks [75], large-scale process engineering plants [46], and water networks, are prototypical examples of cyber-physical systems. Examples of water networks include open channel flows [55] for irrigation purposes and municipal water networks [11, 13]. The vulnerability of open channel networks to cyber-physical attacks has been studied in [3, 102]. Municipal water networks are also known to be susceptible to attacks on the hydraulics [101], and to biochemical contamination threats [32].

We focus on the hydraulics of a municipal water distribution network [11, 13]. This water network can be modeled as a directed graph with node set consisting of reservoirs, junctions, and storage tanks, and with edge set given by pipes, pumps, and valves that are used to convey water from source points to consumers. The state variables are the pressure head h_i at each node i in the network and the flows Q_{ij} from node i to j. The hydraulic model governing the network dynamics includes constant reservoir heads, flow balance equations at junctions and tanks, and pressure difference equations along all edges:

$$
\begin{aligned}
\text{reservoir } i: &\quad h_i = h_i{}^{\text{reservoir}} = \text{constant}\,,\\
\text{junction } i: &\quad d_i = \sum\nolimits_{j\to i} Q_{ji} - \sum\nolimits_{i\to k} Q_{ik}\,,\\
\text{tank } i: &\quad A_i \dot{h}_i = \sum\nolimits_{j\to i} Q_{ji} - \sum\nolimits_{i\to k} Q_{ik}\,,\\
\text{pipe } (i,j): &\quad Q_{ij} = Q_{ij}(h_i - h_j)\,,\\
\text{pump } (i,j): &\quad h_j - h_i = +\Delta h_{ij}{}^{\text{pump}} = \text{constant}\,,\\
\text{valve } (i,j): &\quad h_j - h_i = -\Delta h_{ij}{}^{\text{valve}} = \text{constant}\,.
\end{aligned}
\tag{3.6}
$$

Here d_i is the demand at junction i, A_i is the (constant) cross-sectional area of storage tank i, and the notation "$j \to i$" denotes the set of nodes j connected to node i. The flow Q_{ij} depends on the pressure drop $h_i - h_j$ along pipe according to the Hazen-Williams equation

$$Q_{ij}(h_i - h_j) = g_{ij}|h_i - h_j|^{1/1.85-1} \cdot (h_i - h_j),$$

where $g_{ij} > 0$ is the pipe conductance.

3.3 Linear Consensus Networks

Networks of autonomous agents or sensors have recently attracted the interest of the computer science and control communities. In networks of autonomous agents, an important task is to reach agreement or *consensus* upon the value of a variable of interest, such as the work load in a network of parallel computers, the clock speed for wireless sensor networks, the velocity, the rendezvous point, or the formation pattern for a team of autonomous vehicles; e.g., see [12, 73, 89].

A consensus algorithm is an algorithm for the agents of a consensus networks to reach the desired consensus. Let the graph $G = (\mathcal{V}, \mathcal{E})$ denote a network of interacting autonomous agents, where $\mathcal{V} = \{1, \dots, n\}$, and $\mathcal{E} \subset \mathcal{V} \times \mathcal{V}$. We let each vertex $j \in \mathcal{V}$ denote an autonomous agent, and we associate a real number x_j with each agent j. Let the vector $x \in \mathbb{R}^n$ contain the values x_j. A linear iteration over G is an update rule for x and is described by the linear discrete time system

$$x(t+1) = Ax(t), \tag{3.7}$$

where the (i, j)-th entry of A is nonzero only if $(j, i) \in \mathcal{E}$. If the matrix A is row stochastic and primitive, then, independent of the initial values of the nodes, the network asymptotically converges to a configuration in which the state of the agents coincides. In the latter case, the matrix A is referred to as a *consensus matrix*, and the system (3.7) is called *consensus system*. The graph G is referred to as the communication graph associated with the consensus system (3.7) or, equivalently, with the consensus matrix A. A detailed treatment of the applications, and the convergence aspects of the consensus algorithm is in [12, 73, 89].

Other interesting examples of cyber-physical systems captured by our modeling framework are dynamic Leontief models of multi-sector economies, mixed gas-power energy networks, and large-scale control systems.

Chapter 4

Fundamental Attack Detection and Identification Limitations

In this chapter we characterize fundamental attack detection and identification limitations from system-theoretic and graph-theoretic perspectives. We start by presenting a framework for cyber-physical systems, monitors, and attacks.

4.1 Mathematical Models

In this section we model cyber-physical systems under attack as linear time-invariant descriptor systems subject to unknown inputs. This modeling framework is very general and includes most of the existing cyber-physical models, attacks, and fault scenarios. Indeed, as shown in Chapter 3, many interesting real-world cyber-physical systems contain conserved physical quantities leading to differential-algebraic system descriptions, and, as we show later, most attacks and faults can be modeled by additive inputs on the state and the measurements.

4.1.1 Model of cyber-physical systems under attack

We consider the linear time-invariant descriptor system[1]

$$\begin{aligned} E\dot{x}(t) &= Ax(t) + Bu(t), \\ y(t) &= Cx(t) + Du(t), \end{aligned} \tag{4.1}$$

where $x : \mathbb{R}_{\geq 0} \to \mathbb{R}^n$, $y : \mathbb{R}_{\geq 0} \to \mathbb{R}^p$, $E \in \mathbb{R}^{n\times n}$, $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$, $C \in \mathbb{R}^{p\times n}$, and $D \in \mathbb{R}^{p\times m}$. Here the matrix E is possibly singular, and the inputs Bu and Du are unknown signals describing disturbances affecting the plant. Besides reflecting the genuine failure of systems components, these disturbances model the effect of an attack against the cyber-physical system (see below for our attack model). For notational convenience and without affecting generality, we assume that each state and output variable can be independently compromised by an attacker. Thus, we let $B = \begin{bmatrix} I, 0 \end{bmatrix}$ and $D = \begin{bmatrix} 0, I \end{bmatrix}$ be partitioned into identity and zero matrices of appropriate dimensions, and, accordingly, $u = \begin{bmatrix} u_x^\mathsf{T}, u_y^\mathsf{T} \end{bmatrix}$. Hence, the *attack* $(Bu, Du) = (u_x, u_y)$ can be classified as *state attack* if it affects the

[1] The results stated in this thesis for continuous-time descriptor systems hold also for discrete-time descriptor systems and nonsingular systems. Moreover, we neglect the presence of known inputs, since, due to the linearity of system (4.1), they do not affect our results on the detectability and identifiability of unknown input attacks.

system dynamics, and as *output attack* if it corrupts directly the measurements vector.

The attack signal $u : \mathbb{R}_{\geq 0} \to \mathbb{R}^{n+p}$ depends upon the specific attack strategy. In particular, if $K \subseteq \{1, \dots, n+p\}$ is the *attack set*, with $|K| = k$, then all (and only) the entries of u indexed by K are nonzero over time, that is, for each $i \in K$, there exists a time t such that $u_i(t) \neq 0$, and $u_j(t) = 0$ for all $j \notin K$ and at all times. To underline this sparsity relation, we sometimes use u_K to denote the *attack mode*, that is the subvector of u indexed by K. Accordingly, the pair (B_K, D_K), where B_K and D_K are the submatrices of B and D with columns in K, to denote the *attack signature*. Hence, $Bu = B_K u_K$, and $Du = D_K u_K$. Since the matrix E may be singular, we make the following assumptions on system (4.1):

(A1) the pair (E, A) is regular, that is, the determinant $|sE - A|$ does not vanish identically,

(A2) the initial condition $x(0) \in \mathbb{R}^n$ is consistent, that is,

$$(Ax(0) + Bu(0)) \in \operatorname{Im}(E);$$

(A3) the input u is smooth.

The regularity assumption (A1) ensures the existence of a unique solution x to (4.1). Assumptions (A2) and (A3) simplify the technical presentation in this thesis since they guarantee smoothness of the state trajectory x and the measurements y; see [35, Lemma 2.5] for further details. The degree of smoothness in assumption (A3) depends on the index of (E, A), see [48, Theorem 2.42], and continuity of u is sufficient for the index-one examples presented in Chapter 3. In Section 4.2.5 we discuss the case when assumptions (A2) and (A3) are dropped.

4.1.2 Model of static, dynamic, and active monitors

A *monitor* is a deterministic pair (Φ, γ), where $\Phi : \Lambda \to \Psi$ is an algorithm, and $\gamma : \mathbb{R}_{\geq 0} \to \mathbb{R}^{n+p}$ is an input signal. In particular, Λ is the algorithm input, which we specify later, $\Psi = \{\psi_1, \psi_2\}$, with $\psi_1 \in \{\text{True}, \text{False}\}$ and $\psi_2 \subseteq \{1, \ldots, n+p\}$, is the algorithm output, and γ is an auxiliary input injected by the monitor into the system (4.1). We consider the following classes of monitors.

Definition 1 ***(Static monitor)*** *A* static monitor *is a monitor with* $\gamma(t) = 0\ \forall t \in \mathbb{R}_{\geq 0}$*, and* $\Lambda = \{C, y(t)\ \forall t \in \mathbb{N}\}$.

Note that static monitors do not exploit relations among measurements taken at different time instants. An example of static monitor is the *bad data detector* [1].

Definition 2 ***(Dynamic monitor)*** *A* dynamic monitor *is a monitor with* $\gamma(t) = 0\ \forall t \in \mathbb{R}_{\geq 0}$*, and* $\Lambda = \{E, A, C, y(t)\ \forall t \in \mathbb{R}_{\geq 0}\}$.

Differently from static monitors, dynamic monitors have knowledge of the system dynamics generating y and may exploit temporal relations among different measurements. The filters defined in [80] are examples of dynamic monitors.

Definition 3 ***(Active monitor)*** *An* active monitor *is a monitor with* $\gamma(t) \neq 0$ *for some* $t \in \mathbb{R}_{\geq 0}$*, and* $\Lambda = \{E, A, C, y(t)\ \forall t \in \mathbb{R}_{\geq 0}\}$.

Active monitors are dynamic monitors with the ability of modifying the system dynamics through an input. An example of active monitor is presented in [69] to detect replay attacks.

Since we only consider deterministic cyber-physical systems, we assume monitors to be *consistent*, that is,

(i) $\psi_1 =$ True *only if* the attack set K is nonempty ($\psi_1 =$ False, otherwise),

(ii) $\psi_2 = \emptyset$ *if and only if* $\psi_1 =$ False, and

(iii) $\psi_2 = K$ *only if* K is the (unique) smallest subset $S \subseteq \{1, \dots, n+p\}$ satisfying $y(t) = y(x_1, u_S, t)$ for some initial state x_1 and at all times ($\psi_2 = \{1, \dots, n+p\}$, otherwise).

Due to consistency, the above monitors do not trigger false-alarms.

The objective of a monitor is twofold:

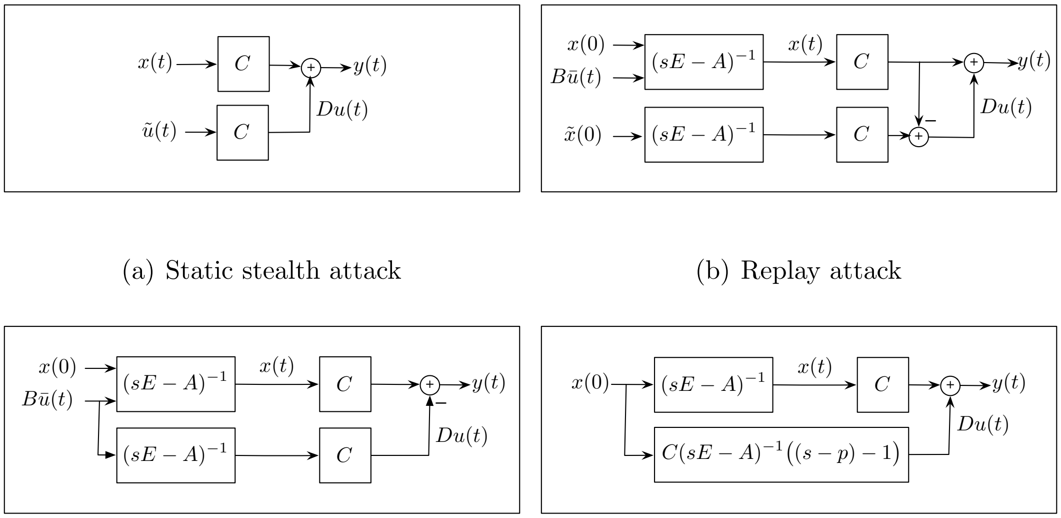
Definition 4 ***(Attack detection)*** *A nonzero attack* $(B_K u_K, D_K u_K)$ *is detected by a monitor if* $\psi_1 = \text{True}$.

Definition 5 ***(Attack identification)*** *A nonzero attack* $(B_K u_K, D_K u_K)$ *is identified by a monitor if* $\psi_2 = K$.

An attack is called *undetectable* (respectively *unidentifiable*) by a monitor if it fails to be detected (respectively identified) by every monitor in the same class. Of course, an undetectable attack is also unidentifiable, since it cannot be distinguished from the zero attack. By extension, an attack set K is undetectable (respectively unidentifiable) if there exists an undetectable (respectively unidentifiable) attack $(B_K u_K, D_K u_K)$. Notice that no assumptions are made on the algorithm Φ and the input signal γ. Hence, the monitoring limitations we will discuss are fundamental, and they apply to any monitoring algorithm.

4.1.3 Model of attacks

In this work we consider colluding omniscient attackers with the ability of altering the cyber-physical dynamics through exogenous inputs. In particular we let the attack (Bu, Du) in (4.1) be designed based on knowledge of the system structure and parameters E, A, C, and the full state x at all times. Additionally, attackers have unlimited computation capabilities, and their objective is to disrupt



(a) Static stealth attack

(b) Replay attack

(c) Covert attack

(d) Dynamic false data injection

Figure 4.1: *A block diagram illustration of prototypical attacks is here reported. In Fig. 4.1(a) the attacker corrupts the measurements y with the signal $D_K u_K \in \mathrm{Im}(C)$. Notice that in this attack the dynamics of the system are not considered. In Fig. 4.1(b) the attacker affects the output so that $y(t) = y(x(0), [\bar{u}_K^\mathsf{T}\ u_K^\mathsf{T}]^\mathsf{T}, t) = y(\tilde{x}(0), 0, t)$. The covert attack in Fig. 4.1(c) is a feedback version of the replay attack, and it can be explained analogously. In Fig. 4.1(d) the attack is such that the unstable pole p is made unobservable.*

the physical state or the measurements while avoiding detection. Clearly, specific attacks may require weaker attack capabilities to be cast.

Remark 1 ***(Existing attack strategies as subcases)*** *The following prototypical attacks can be modeled and analyzed through our theoretical framework:*

(i) stealth attacks *defined in [25] correspond to output attacks compatible with the measurements equation;*

(ii) replay attacks *defined in [69] are state and output attacks which affect the system dynamics and reset the measurements;*

(iii) covert attacks *defined in [102] are closed-loop replay attacks, where the output attack is chosen to cancel out the effect on the measurements of the state attack; and*

(iv) (dynamic) false-data injection attacks *defined in [68] are output attacks rendering an unstable mode (if any) of the system unobservable.*

A possible implementation of the above attacks is illustrated in Fig. 4.1. □

To conclude this section we remark that the examples presented in Chapter 3 are captured in our framework. In particular, classical power networks failures modeled by additive inputs include sudden change in the mechanical power input to generators, lines outage, and sensors failure; see [80] for a detailed discussion. Analogously, for a water network, faults modeled by additive inputs include leakages, variation in demand, and failures of pumps and sensors. Possible cyber-physical attacks in both power and water networks include comprising measurements [3, 56, 110] and attacks on the control architecture or the physical state itself [26, 70, 101, 103]. Similar situations are envisioned for consensus networks.

4.2 Limitations of Static, Dynamic and Active Monitors

The objective of this section is to highlight fundamental detection and identification limitations of static, dynamic, and active monitors. In particular, we show that the performance of widely used static monitors can be improved by exploiting the system dynamics. On the other hand, the possibility of injecting monitoring signals does not improve the detection capabilities of a (passive) dynamic monitor.

Observe that a cyber-physical attack is undetectable if there exists a normal operating condition of the system under which the output would be the same as under the perturbation due to the attacker. Let $y(x_0, u, t)$ be the output sequence generated from the initial state x_0 under the attack signal u.

Lemma 4.2.1 ***(Undetectable attack)*** *For the linear descriptor system* (4.1)*, the attack* $(B_K u_K, D_K u_K)$ *is undetectable by a static monitor if and only if*

$$y(x_1, u_K, t) = y(x_2, 0, t),$$

for some initial condition $x_1, x_2 \in \mathbb{R}^n$ *and for all* $t \in \mathbb{N}_0$*. If the same holds for* $t \in \mathbb{R}_{\geq 0}$*, then the attack is also undetectable by a dynamic monitor.*

Lemma 7 follows from the fact that our monitors are deterministic, so that $y(x_1, u_K, t)$ and $y(x_2, 0, t)$ lead to the same output ψ_1. A more general concern than detectability is identifiability of attackers, that is, the possibility to distinguish from measurements between the action of two distinct attacks. We quantify the strength of an attack through the cardinality of the attack set. Since an attacker can independently compromise any state variable or measurement, every subset of the states and measurements of fixed cardinality is a possible attack set.

Lemma 4.2.2 ***(Unidentifiable attack)*** *For the linear descriptor system* (4.1)*, the attack* $(B_K u_K, D_K u_K)$ *is unidentifiable by a static monitor if and only if*

$$y(x_1, u_K, t) = y(x_2, u_R, t),$$

for some initial condition $x_1, x_2 \in \mathbb{R}^n$*, attack* $(B_R u_R, D_R u_R)$ *with* $|R| \leq |K|$ *and* $R \neq K$*, and for all* $t \in \mathbb{N}_0$*. If the same holds for* $t \in \mathbb{R}_{\geq 0}$*, then the attack is also unidentifiable by a dynamic monitor.*

Lemma 8 follows analogously to Lemma 7. We now elaborate on the above results to derive fundamental monitoring limitations for the considered monitors.

4.2.1 Fundamental limitations of static monitors

Following Lemma 7, an attack is undetectable by a static monitor if and only if, for all $t \in \mathbb{N}_0$, there exists a vector $\xi(t)$ such that $y(t) = C\xi(t)$. Notice that this condition is compatible with [56], where an attack is detected if and only if the residual $r(t) = y(t) - C\hat{x}(t)$ is nonzero for some $t \in \mathbb{N}_0$, where $\hat{x}(t) = C^{\dagger}y(t)$. Let $\|v\|_0$ denote the number of nonzero components of the vector v.

Theorem 4.2.3 ***(Static detectability of cyber-physical attacks)*** *For the cyber-physical descriptor system* (4.1) *and an attack set K, the following statements are equivalent:*

(i) the attack set K is undetectable by a static monitor;

(ii) there exists an attack mode u_K satisfying, for some $x(t)$ and at every $t \in \mathbb{N}_0$,

$$Cx(t) + D_K u_K(t) = 0. \tag{4.2}$$

Moreover, there exists an attack set K, with $|K| = k \in \mathbb{N}_0$, undetectable by a static monitor if and only if there exist $x \in \mathbb{R}^n$ such that $\|Cx\|_0 = k$.

Before presenting a proof of the above theorem, we highlight that a necessary and sufficient condition for the equation (4.2) to be satisfied is that $D_K u_K(t) = u_{y,K}(t) \in \operatorname{Im}(C)$ at all times $t \in \mathbb{N}_0$, where $u_{y,K}(t)$ is the vector of the last p components of $u_K(t)$. Hence, statement (ii) in Theorem 4.2.3 implies that *no* state attack can be detected by a static detection procedure, and that an undetectable output attack exists if and only if $\operatorname{Im}(D_K) \cap \operatorname{Im}(C) \neq \{0\}$.

Proof of Theorem 4.2.3: As previously discussed, the attack K is undetectable by a static monitor if and only if for each $t \in \mathbb{N}$ there exists $x(t)$, and $u_K(t)$ such that

$$r(t) = y(t) - CC^{\dagger}y(t) = (I - CC^{\dagger})\,(Cx(t) + D_K u_K(t))$$

vanishes. Consequently, $r(t) = (I - CC^{\dagger})D_K u_K(t)$, and the attack set K is undetectable if and only if $D_K u_K(t) \in \operatorname{Im}(C)$, which is equivalent to statement (ii). The last necessary and sufficient condition in the theorem follows from (ii), and the fact that every output variable can be attacked independently of each other since $D = \begin{bmatrix}0, I\end{bmatrix}$. □

We now focus on the static identification problem. Following Lemma 8, the following result is stated.

Theorem 4.2.4 ***(Static identification of cyber-physical attacks)*** *For the cyber-physical descriptor system* (4.1) *and an attack set K, the following statements are equivalent:*

(i) the attack set K is unidentifiable by a static monitor;

(ii) there exists an attack set R, with $|R| \leq |K|$ and $R \neq K$, and attack modes u_K, u_R satisfying, for some $x(t)$ and at every $t \in \mathbb{N}_0$,

$$Cx(t) + D_K\,(u_K(t) + u_R(t)) = 0.$$

Moreover, there exists an attack set K, with $|K| = k \in \mathbb{N}_0$, unidentifiable by a static monitor if and only if there exists an attack set $\bar{K}$, with $|\bar{K}| \leq 2k$, which is undetectable by a static monitor.

Similar to the fundamental limitations of static detectability in Theorem 4.2.3, Theorem 4.2.4 implies that, for instance, state attacks cannot be identified and that an undetectable output attack of cardinality k exists if and only if $\mathrm{Im}(D_{\bar{K}}) \cap \mathrm{Im}(C) \neq \{0\}$, for some attack set $\bar{K}$ with $|\bar{K}| \leq 2k$.

Proof of Theorem 4.2.4: Due to linearity of the system (4.1), the unidentifiability condition in Lemma 8 is equivalent to $y(x_K - x_R, u_K - u_R, t) = 0$, for some initial conditions x_K, x_R, and attack modes u_K, u_R. The equivalence between statements (i) and (ii) follows. The last statement follows from Theorem 4.2.3. □

4.2.2 Fundamental limitations of dynamic monitors

As opposed to a static monitor, a dynamic monitor checks for the presence of attacks at every time $t \in \mathbb{R}_{\geq 0}$. Intuitively, a dynamic monitor is harder to mislead than a static monitor. The following theorem formalizes this expected result.

Theorem 4.2.5 ***(Dynamic detectability of cyber-physical attacks)*** *For the cyber-physical descriptor system* (4.1) *and an attack set K, the following statements are equivalent:*

(i) the attack set K is undetectable by a dynamic monitor;

(ii) there exists an attack mode u_K satisfying, for some $x(0)$ and for every $t \in \mathbb{R}_{\geq 0}$,

$$\begin{aligned} E\dot{x}(t) &= Ax(t) + B_K u_K(t)\,, \\ 0 &= Cx(t) + D_K u_K(t)\,; \end{aligned}$$

(iii) there exist $s \in \mathbb{C}$, $g \in \mathbb{R}^{|K|}$, and $x \in \mathbb{R}^n$, with $x \neq 0$, such that $(sE - A)x - B_K g = 0$ and $Cx + D_K g = 0$.

Moreover, there exists an attack set K, with $|K| = k$, undetectable by a dynamic monitor if and only if there exist $s \in \mathbb{C}$ and $x \in \mathbb{R}^n$ such that $\|(sE - A)x\|_0 + \|Cx\|_0 = k$.

Before proving Theorem 4.2.5, some comments are in order. First, differently from the static case, state attacks *can be detected* in the dynamic case. Second, in order to mislead a dynamic monitor an attacker needs to inject a signal which is consistent with the system dynamics at every instant of time. Hence, as opposed to the static case, the condition $D_K u_K(t) = u_{y,K}(t) \in \operatorname{Im}(C)$ needs to be satisfied for every $t \in \mathbb{R}_{\geq 0}$, and it is only necessary for the undetectability of an output attack. Indeed, for instance, state attacks can be detected even though they automatically satisfy the condition $D_K u_K = 0 \in \operatorname{Im}(C)$. Third and finally, according to the last statement of Theorem 4.2.5, the existence of invariant zeros[2] for the system (E, A, B_K, C, D_K) is equivalent to the existence of undetectable attacks. As a consequence, a dynamic monitor performs better than a static monitor, while requiring, possibly, fewer measurements. We refer to Section 4.4.2 for an illustrative example of this last statement.

Proof of Theorem 4.2.5: By Lemma 7 and linearity of the system (4.5), the attack mode u_K is undetectable by a dynamic monitor if and only if there exists x_0 such that $y(x_0, u_K, t) = 0$ for all $t \in \mathbb{R}_{\geq 0}$, that is, if and only if the system (4.1) features zero dynamics. Hence, statements (i) and (ii) are equivalent. For a linear descriptor system with smooth input and consistent initial condition, the existence of zero dynamics is equivalent to the existence of invariant zeros [35, Theorem 3.2 and Proposition 3.4]. The equivalence of statements (ii) and (iii) follows. The last statement follows from (iii), and the fact that $B = \begin{bmatrix} I, 0 \end{bmatrix}$ and $D = \begin{bmatrix} 0, I \end{bmatrix}$. □

We now consider the identification problem.

Theorem 4.2.6 ***(Dynamic identifiability of cyber-physical attacks)*** *For the cyber-physical descriptor system* (4.1) *and an attack set K, the following statements are equivalent:*

(i) the attack set K is unidentifiable by a dynamic monitor;

[2] For the system (E, A, B_K, C, D_K), the value $s \in \mathbb{C}$ is an invariant zero if there exists $x \in \mathbb{R}^n$, with $x \neq 0$, $g \in \mathbb{R}^{|K|}$, such that $(sE - A)x - B_K g = 0$ and $Cx + D_K g = 0$.

(ii) there exists an attack set R, with $|R| \leq |K|$ and $R \neq K$, and attack modes u_K, u_R satisfying, for some $x(0)$ and for every $t \in \mathbb{R}_{\geq 0}$,

$$\begin{aligned} E\dot{x}(t) &= Ax(t) + B_K u_K(t) + B_R u_R(t)\,, \\ 0 &= Cx(t) + D_K u_K(t) + D_R u_R(t)\,; \end{aligned}$$

(iii) there exists an attack set R, with $|R| \leq |K|$ and $R \neq K$, $s \in \mathbb{C}$, $g_K \in \mathbb{R}^{|K|}$, $g_R \in \mathbb{R}^{|R|}$, and $x \in \mathbb{R}^n$, with $x \neq 0$, such that $(sE - A)x - B_K g_K - B_R g_R = 0$ and $Cx + D_K g_K + D_R g_R = 0$.

Moreover, there exists an attack set K, with $|K| = k \in \mathbb{N}_0$, unidentifiable by a dynamic monitor if and only if there exists an attack set $\bar{K}$, with $|\bar{K}| \leq 2k$, which is undetectable by a dynamic monitor.

Proof: Notice that, because of the linearity of the system (4.1), the unidentifiability condition in Lemma 8 is equivalent to the condition $y(x_K - x_R, u_K - u_R, t) = 0$, for some initial conditions x_K, x_R, and attack modes u_K, u_R. The equivalence between statements (i) and (ii) follows. Finally, the last two statements follow from Theorem 4.2.5, and the fact that $B = \begin{bmatrix} I, 0 \end{bmatrix}$ and $D = \begin{bmatrix} 0, I \end{bmatrix}$. □

In other words, the existence of an unidentifiable attack set K of cardinality k is equivalent to the existence of invariant zeros for the system $(E, A, B_{\bar{K}}, C, D_{\bar{K}})$, for some attack set $\bar{K}$ with $|\bar{K}| \leq 2k$. We conclude this section with the following remarks. The existence condition in Theorem 3.4 is hard to verify because of its combinatorial complexity: in order to check if there exists an unidentifiable attack set K, with $|K| = k$, one needs to certify the absence of invariant zeros for all possible $2k$-dimensional attack sets. Thus, a conservative verification scheme requires $\binom{n+p}{2k}$ tests. In Section 4.3 we present intuitive graph-theoretic conditions for the existence of undetectable and unidentifiable attack sets for a given sparsity pattern of the system matrices and generic system parameters. Finally, Theorem 4.2.6 includes as a special case Proposition 4 in [40], which considers exclusively output attacks.

4.2.3 Fundamental limitations of active monitors

An active monitor uses a control signal (unknown to the attacker) to reveal the presence of attacks; see [69] for the case of replay attacks. In the presence of an active monitor with input signal $\gamma = [\gamma_x^\mathsf{T}\ \gamma_y^\mathsf{T}]^\mathsf{T}$, the system (4.1) reads as

$$E\dot{x}(t) = Ax(t) + B_K u_K(t) + \gamma_x(t),$$

$$y(t) = Cx(t) + D_K u_K(t) + \gamma_y(t).$$

Although the attacker is unaware of the signal γ, active and dynamic monitors share the same limitations.

Theorem 4.2.7 ***(Limitations of active monitors)*** *For the cyber-physical descriptor system* (4.1)*, let γ be an additive signal injected by an active monitor.*

The existence of undetectable (respectively unidentifiable) attacks does not depend upon the signal γ. Moreover, undetectable (respectively unidentifiable) attacks can be designed independently of γ.

Proof: For the system (4.1), let u be the attack mode, and let γ be the monitoring input. Let $y(x, u, \gamma, t)$ denotes the output generated by the inputs u and γ with initial condition $x = x_1 + x_2$. Observe that, because of the linearity of (4.1), we have $y(x, u, \gamma, t) = y(x_1, u, 0, t) + y(x_2, 0, \gamma, t)$, with consistent initial conditions x_1 and x_2. Then, an attack u is undetectable if and only if $y(x, u, \gamma, t) = y(\bar{x}, 0, \gamma, t)$, or equivalently $y(x_1, u, 0, t) + y(x_2, 0, \gamma, t) = y(\bar{x}_1, 0, 0, t) + y(x_2, 0, \gamma, t)$, for some initial conditions x and $\bar{x} = \bar{x}_1 + x_2$. The statement follows, since, from the equality above, the detectability of u does not depend upon w. □

As a consequence of Theorem 4.2.7, the existence of undetectable attacks is independent of the presence of known control signals. Therefore, in a worst-case scenario, active monitors are as powerful as dynamic monitors. Since replay attacks are detectable by an active monitor [69], Theorem 4.2.7 shows that replay attacks are not worst-case attacks.

Remark 2 ***(Undetectable attacks in the presence of state and measurements noise)*** *The input γ in Theorem 4.2.7 may represent sensors and actuators noise. In this case, Theorem 4.2.7 states that the existence of undetectable attacks* *for a noise-free system implies the existence of undetectable attacks for the same system driven by noise. The converse does not hold, since attackers may remain undetected by injecting a signal compatible with the noise statistics.* □

4.2.4 Specific results for index-one singular systems

For many interesting real-world descriptor systems, including the examples in Section 3.1 and Section 3.2, the algebraic system equations can be solved explicitly, and the descriptor system (4.1) can be reduced to a nonsingular state space system. For this reason, this section presents specific results for the case of *index-one* systems [53]. In this case, without loss of generality, we assume the system (4.1) to be written in the canonical form

$$\begin{aligned}\begin{bmatrix} E_{11} & 0 \\ 0 & 0 \end{bmatrix}\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} &= \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} B_1 \\ B_2 \end{bmatrix} u_K(t), \\ y(t) &= \begin{bmatrix} C_1 & C_2 \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + D_K u_K(t),\end{aligned} \tag{4.3}$$

where E_{11} is nonsingular and A_{22} is nonsingular. Consequently, the state x_1 and x_2 are referred to as *dynamic state* and *algebraic state*, respectively. The algebraic state can be expressed via the dynamic state and the attack mode as

$$x_2(t) = -A_{22}^{-1}A_{21}x_1(t) - A_{22}^{-1}B_2u_K(t). \tag{4.4}$$

The elimination of the algebraic state x_2 in the descriptor system (4.3) leads to the nonsingular state space system

$$\begin{aligned} \dot{x}_1 &= \underbrace{E_{11}^{-1}\left(A_{11} - A_{12}A_{22}^{-1}A_{21}\right)}_{\tilde{A}} x_1(t) + \underbrace{E_{11}^{-1}\left(B_1 - A_{12}A_{22}^{-1}B_2\right)}_{\tilde{B}_K} u_K(t), \\ y(t) &= \underbrace{\left(C_1 - C_2A_{22}^{-1}A_{21}\right)}_{\tilde{C}} x_1(t) + \underbrace{\left(D_K - C_2A_{22}^{-1}B_2\right)}_{\tilde{D}_K} u_K(t). \end{aligned} \tag{4.5}$$

This reduction of the algebraic states is known as Kron reduction in the literature on power networks and circuit theory [31]. Hence, we refer to (4.5) as the *Kron-reduced system*.

Clearly, for any state trajectory x_1 of the Kron-reduced system (4.5), the corresponding state trajectory $[x_1^\mathsf{T}\ x_2^\mathsf{T}]^\mathsf{T}$ of the (non-reduced) cyber-physical descriptor system (4.1) can be recovered by identity (4.4) and given knowledge of the input u_K. The following subtle issues are easily visible in the Kron-reduced system (4.4). First, a state attack affects directly the output y, provided that $C_2A_{22}^{-1}B_2u_K \neq 0$. Second, since the matrix A_{22}^{-1} is generally fully populated, an attack on a single algebraic component can affect not only the locally attacked state or its vicinity but larger parts of the system.

According to the transformations in (4.5), for each attack set K, the attack signature (B_K, D_K) is mapped to the corresponding signature $(\tilde{B}_K, \tilde{D}_K)$ in the Kron-reduced system. As an apparent disadvantage, the sparsity pattern of the original (non-reduced) cyber-physical descriptor system (4.1) is lost in the Kron-reduced representation (4.5), and so is, possibly, the physical interpretation of the state and the direct representation of system components. However, as we show in the following lemma, the notions of detectability and identifiability of an attack set K defined for the original descriptor system (4.1) are equivalent for the Kron-reduced system (4.5). This property renders the low-dimensional and nonsingular Kron-reduced system (4.5) attractive from a computational point of view to design attack detection and identification monitors; see Chapter 6.

Lemma 4.2.8 ***(Preservation of detectability and identifiability under Kron reduction)*** *For the cyber-physical descriptor system* (4.3)*, the attack set K is detectable (respectively identifiable) if and only if it is detectable (respectively identifiable) for the associated Kron-reduced system* (4.5)*.*

Proof: The lemma follows from the fact that the input and initial condition to output map for the system (4.1) coincides with the corresponding map for the Kron-reduced system (4.5) and equation (4.4). Indeed, according to Theorem 4.2.5, the attack set K is undetectable if and only if there exist $s \in \mathbb{C}$, $g \in \mathbb{R}^{|K|}$, and $x = [x_1^\mathsf{T}\ x_2^\mathsf{T}]^\mathsf{T} \in \mathbb{R}^n$, with $x \neq 0$, such that

$$(sE - A)x - B_K g = 0 \text{ and } Cx + D_K g = 0\,.$$

Equivalently, by eliminating the algebraic constraints as in (4.4), the attack set K is undetectable if and only if the conditons

$$(sI - \tilde{A})x_1 - \tilde{B}_K g = 0 \text{ and } \tilde{C}x_1 + \tilde{D}_K g = 0$$

are satisfied together with $x_2 = -A_{22}^{-1}A_{21}x_1 - A_{22}^{-1}B_2 g$. Notice that the latter equation is always satisfied due to the consistency assumption (A2), and the equivalence of detectability of the attack set K follows. The equivalence of attack identifiability follows by analogous arguments. □

4.2.5 The case of inconsistent initial states and impulsive inputs

We now discuss the case of non-smooth attack signal and inconsistent initial condition. If the consistency assumption (A2) is dropped, then discontinuities in the state $x(t \downarrow 0)$ may affect the measurements $y(t \downarrow 0)$. For instance for index-one systems, an inconsistent initial condition leads to an initial jump for the algebraic variable $x_2(t \downarrow 0)$ to obey equation (4.4). Consequently, the inconsistent initial value $[0^\mathsf{T}\ x_2(0)^\mathsf{T}]^\mathsf{T} \in \operatorname{Ker}(E)$ cannot be recovered through measurements.

Assumption (A4) requires the attack signal to be sufficiently smooth such that x and y are at least continuous. Suppose that assumption (A3) is dropped and the input u belongs to the class of impulsive smooth distributions $\mathcal{C}_{\text{imp}} =$

$\mathcal{C}_{\text{smooth}} \cup \mathcal{C}_{\text{p-imp}}$, that is, loosely speaking, the class of functions given by the linear combination of a smooth function on $\mathbb{R}_{\geq 0}$ (denoted by $\mathcal{C}_{\text{smooth}}$) and Dirac impulses and their derivatives at $t = 0$ (denoted by $\mathcal{C}_{\text{p-imp}}$), see [35], [48, Section 2.4]. In this case, an attacker commanding an impulsive input $u(0) \in \mathcal{C}_{\text{imp}}$ can reset the initial state $x(0)$ and, possibly, evade detection.

The discussion in the previous two paragraphs can be formalized as follows. Let $\mathcal{V}_c$ be the subspace of points $x_0 \in \mathbb{R}^n$ of consistent initial conditions for which there exists an input $u \in \mathcal{C}^m_{\text{smooth}}$ and a state trajectory $x \in \mathcal{C}^n_{\text{smooth}}$ to the descriptor system (4.1) such that $y(t) = 0$ for all $t \in \mathbb{R}_{\geq 0}$. Let $\mathcal{V}_d$ (respectively $\mathcal{W}$) be the subspace of points $x_0 \in \mathbb{R}^n$ for which there exists an input $u \in \mathcal{C}^{n+p}_{\text{imp}}$ (respectively $u \in \mathcal{C}^{n+p}_{\text{p-imp}}$) and a state trajectory $x \in \mathcal{C}^n_{\text{imp}}$ (respectively $x \in \mathcal{C}^n_{\text{p-imp}}$) to the descriptor system (4.1) such that $y(t) = 0$ for all $t \in \mathbb{R}_{\geq 0}$. The output-nulling subspace $\mathcal{V}_d$ can be decomposed as follows:

Lemma 4.2.9 ***(Decomposition of output-nulling space [35, Theorem 3.2 and Proposition 3.4]))*** $\mathcal{V}_d = \mathcal{V}_c + \mathcal{W} + \operatorname{Ker}(E)$.

In words, from an initial condition $x(0) \in \mathcal{V}_d$ the output can be nullified by a smooth input or by an impulsive input (with consistent or inconsistent initial conditions in $\operatorname{Ker}(E)$).

In this work we focus on the smooth output-nulling subspace $\mathcal{V}_c$, which is exactly space of zero dynamics identified in Theorems 4.2.5 and 4.2.6. Hence, by Lemma 4.2.9, for inconsistent initial conditions, the results presented in this section are valid only for strictly positive times $t > 0$. On the other hand, if an attacker is capable of injecting impulsive signals, then it can avoid detection for initial conditions $x(0) \in \mathcal{W}$.

4.3 Graph Theoretic Detectability Conditions

In this section we characterize undetectable attacks against cyber-physical systems from a structural perspective. In particular, we derive detectability conditions based upon a connectivity property of a graph associated with the system. For the ease of notation, we now drop the subscript K from B_K, D_K, and u_K.

4.3.1 Preliminary notions

We start by recalling some useful facts about structured systems and structural properties [87, 117]. Let a *structure matrix* $[M]$ be a matrix in which each entry is either a fixed zero or an indeterminate parameter. The system

$$\begin{aligned} [E]\dot{x}(t) &= [A]x(t) + [B]u(t), \\ y(t) &= [C]x(t) + [D]u(t). \end{aligned} \tag{4.6}$$

is called *structured system*, and it is sometimes referred to with the tuple $([E], [A], [B], [C], [D])$ of structure matrices. A system (E, A, B, C, D) is an admissible realization of $([E], [A], [B], [C], [D])$ if it can be obtained from the latter by fixing the indeterminate entries at some particular value. Two systems are structurally equivalent if they are both an admissible realization of the same structured system. Let d be the number of indeterminate entries of a structured system altogether. By collecting the indeterminate parameters into a vector, an admissible realization is mapped to a point in the Euclidean space $\mathbb{R}^d$. A property which can be asserted on a dynamical system is called *structural* if, informally, it holds for *almost all* admissible realizations. To be more precise, we say that a property is structural if and only if the set of admissible realizations satisfying such property forms a dense subset of the parameters space.[3] For instance, left-invertibility of a nonsingular system is a structural property with respect to $\mathbb{R}^d$ [29].

Consider the structured cyber-physical system (4.6). It is often the case that, for the tuple (E, A, B, C, D) to be an admissible realization of (4.6), the numerical entries need to satisfy certain algebraic relations. For instance, for (E, A, B, C, D) to be an admissible power network realization, the matrices E and A need to be of the form (3.4). Let $\mathbb{S} \subseteq \mathbb{R}^d$ be the admissible parameter space. We make the following assumption:

(A4) the admissible parameters space $\mathbb{S}$ is a polytope of $\mathbb{R}^d$, that is, $\mathbb{S} = \{x \in \mathbb{R}^d : Mx \geq 0\}$ for some matrix M.

[3] A subset $S \subseteq P \subseteq \mathbb{R}^d$ is dense in P if, for each $r \in P$ and every $\varepsilon > 0$, there exists $s \in S$ such that the Euclidean distance $\|s - r\| \leq \varepsilon$.

It should be noticed that assumption (A4) is automatically verified for the case of power networks [77, Lemma 3.1]. Unfortunately, if the admissible parameters space is a subset of $\mathbb{R}^d$, then classical structural system-theoretic results are, in general, not valid [87, Section 15].

We now define a mapping between dynamical systems in descriptor form and digraphs. Let ($[E]$,$[A]$,$[B]$,$[C]$,$[D]$) be a structured cyber-physical system under attack. We associate a directed graph $G = (\mathcal{V}, \mathcal{E})$ with the tuple ($[E]$,$[A]$,$[B]$,$[C]$,$[D]$). The vertex set is $\mathcal{V} = \mathcal{U} \cup \mathcal{X} \cup \mathcal{Y}$, where $\mathcal{U} = \{u_1, \dots, u_m\}$ is the set of input vertices, $\mathcal{X} = \{x_1, \dots, x_n\}$ is the set of state vertices, and $\mathcal{Y} = \{y_1, \dots, y_p\}$ is the set of output vertices. If (i, j) denotes the edge from the vertex i to the vertex j, then the edge set $\mathcal{E}$ is $\mathcal{E}_{[E]} \cup \mathcal{E}_{[A]} \cup \mathcal{E}_{[B]} \cup \mathcal{E}_{[C]} \cup \mathcal{E}_{[D]}$, with $\mathcal{E}_{[E]} = \{(x_j, x_i) : [E]_{ij} \neq 0\}$, $\mathcal{E}_{[A]} = \{(x_j, x_i) : [A]_{ij} \neq 0\}$, $\mathcal{E}_{[B]} = \{(u_j, x_i) : [B]_{ij} \neq 0\}$, $\mathcal{E}_{[C]} = \{(x_j, y_i) : [C]_{ij} \neq 0\}$, and $\mathcal{E}_{[D]} = \{(u_j, y_i) : [D]_{ij} \neq 0\}$. In the latter, for instance, the expression $[E]_{ij} \neq 0$ means that the (i, j)-th entry of $[E]$ is a free parameter.

Example 1 ***(Power network structural analysis)*** *Consider the power network illustrated in Fig. 4.2, where, being e_i the i-th canonical vector, we take $[E] = \operatorname{blkdiag}(1, 1, 1, M_1, M_2, M_3, 0, 0, 0, 0, 0, 0)$, $[B] = [e_8\ \ e_9]$, $[C] = [e_1\ \ e_4]^\mathsf{T}$,*

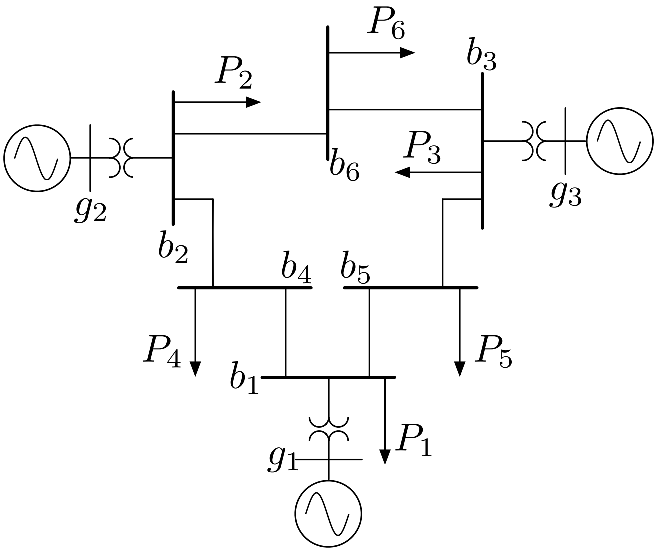


Figure 4.2: *WSSC power network with* 3 *generators and* 6 *buses. The numerical value of the network parameters can be found in [97].*

$[D] = 0$, *and* $[A]$ *equal to*

$$\begin{bmatrix}
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
a_{4,1} & 0 & 0 & a_{4,4} & 0 & 0 & a_{4,7} & 0 & 0 & 0 & 0 & 0 \\
0 & a_{5,2} & 0 & 0 & a_{5,5} & 0 & 0 & a_{5,8} & 0 & 0 & 0 & 0 \\
0 & 0 & a_{6,3} & 0 & 0 & a_{6,6} & 0 & 0 & a_{6,9} & 0 & 0 & 0 \\
a_{7,1} & 0 & 0 & 0 & 0 & 0 & a_{7,7} & 0 & 0 & a_{7,10} & a_{7,11} & 0 \\
0 & a_{8,2} & 0 & 0 & 0 & 0 & 0 & a_{8,8} & 0 & a_{8,10} & 0 & a_{8,12} \\
0 & 0 & a_{9,3} & 0 & 0 & 0 & 0 & 0 & a_{9,9} & 0 & a_{9,11} & a_{9,12} \\
0 & 0 & 0 & 0 & 0 & 0 & a_{10,7} & a_{10,8} & 0 & a_{10,10} & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & a_{11,7} & 0 & a_{11,9} & 0 & a_{11,11} & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & a_{12,8} & a_{12,9} & 0 & 0 & a_{12,12}
\end{bmatrix}$$

The digraph associated with $([E], [A], [B], [C], [D])$ *is shown in Fig. 4.3.* □

4.3.2 Network vulnerability with known initial state

We derive graph-theoretic detectability conditions for two different scenarios. Recall from Lemma 7 that an attack u is undetectable if $y(x_1, u, t) = y(x_2, 0, t)$ for some initial states x_1 and x_2. In this section, we assume that the system

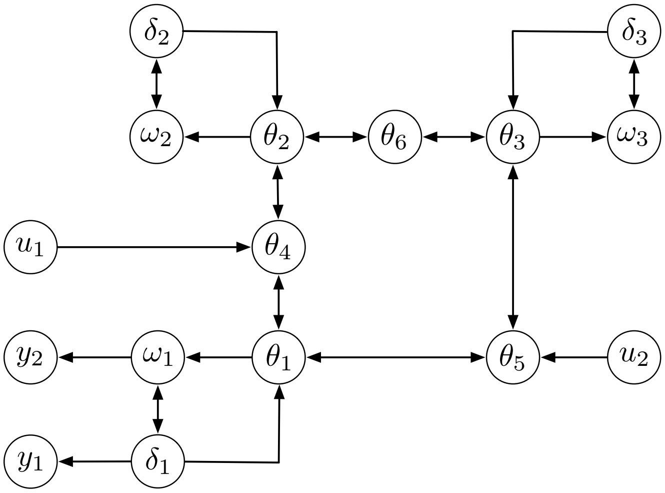


Figure 4.3: *The digraph associated with the network in Fig. 4.2. The self-loops of the vertices* $\{\delta_1, \delta_2, \delta_3\}$*,* $\{\omega_1, \omega_2, \omega_3\}$*, and* $\{\theta_1, \dots, \theta_6\}$ *are not drawn. The inputs* u_1 *and* u_2 *affect respectively the bus* b_4 *and the bus* b_5*. The measured variables are the rotor angle and frequency of the first generator.*

state is known at the failure initial time,[4] so that an attack u is undetectable if $y(x_0, u, t) = y(x_0, 0, t)$ for some system initial state x_0. The complementary case of unknown initial state is studied in Section 4.3.3.

Consider the cyber-physical system described by the matrices (E, A, B, C, D), and notice that, if the initial state is known, then the attack undetectability condition $y(x_0, u, t) = y(x_0, 0, t)$ coincides with the system being not left-invertible.[5] Recall that a subset $S \subseteq \mathbb{R}^d$ is an *algebraic variety* if it coincides with the locus of common zeros of a finite number of polynomials [117].

[4] The failure initial state can be estimated through a state observer [97].

[5] A regular descriptor system is left-invertible if and only if its transfer matrix $G(s)$ is of full column rank for all almost all $s \in \mathbb{C}$, or if and only if $\left[\begin{smallmatrix} sE-A & -B \\ C & D \end{smallmatrix}\right]$ has full column rank for almost all $s \in \mathbb{C}$ [35, Theorem 4.2].

Lemma 4.3.1 ***(Polytopes and algebraic varieties)*** *Let $S \subseteq \mathbb{R}^d$ be a polytope, and let $T \subseteq \mathbb{R}^d$ be an algebraic variety. Then, either $S \subseteq T$, or $S \setminus (S \cap T)$ is dense in S.*

Proof: Let $T \subseteq \mathbb{R}^d$ be the algebraic variety described by the locus of common zeros of the polynomials $\{\phi_1(x), \dots, \phi_t(x)\}$, with $t \in \mathbb{N}$, $t < \infty$. Let $P \subseteq \mathbb{R}^d$ be the smallest vector subspace containing the polytope S. Then $P \subseteq T$ if and only if every polynomial ϕ_i vanishes identically on P. Suppose that the polynomial ϕ_i does not vanish identically on P. Then, the set $T \cap P$ is contained in the algebraic variety $\{x \in P : \phi_i(x) = 0\}$, and, therefore [117], the complement $P \setminus (P \cap T)$ is dense in P. By definition of a dense set, the set $S \setminus (S \cap T)$ is also dense in S. □

In Lemma 4.3.1 interpret the polytope S as the admissible parameters space of a structured cyber-physical system. Then we have shown that left-invertibility of a cyber-physical system is a structural property even when the admissible parameters space is a polytope of the whole parameters space. Consequently, given a structured cyber-physical system, either every admissible realization admits an undetectable attack, or there is no undetectable attack in almost all admissible realizations. Moreover, in order to show that almost all realizations have no undetectable attacks, it is sufficient to prove that this is the case for some specific admissible realizations. Before presenting our main result, we recall the following result. Let $\bar{E}$ and $\bar{A}$ be N-dimensional square matrices, and let $G(s\bar{E} - \bar{A})$ be the graph associated with the matrix $s\bar{E} - \bar{A}$ that consists of N vertices, and an edge from vertex j to i if $\bar{A}_{ij} \neq 0$ or $\bar{E}_{ij} \neq 0$. The matrix $s[\bar{E}] - [\bar{A}]$ is said to be structurally degenerate if, for any admissible realization $\bar{E}$ (respectively $\bar{A}$) of $[\bar{E}]$ (respectively $[\bar{A}]$), the determinant $|s\bar{E} - \bar{A}|$ vanishes for all $s \in \mathbb{C}$. Recall the following definitions from [29]. For a given graph G, a path is a sequence of vertices where each vertex is connected to the following one in the sequence. A path is simple if every vertex on the path (except possibly the first and the last vertex) occurs only once. Two paths are disjoint if they consist of disjoint sets of vertices. A set of l mutually disjoint and simple paths between two sets of vertices S_1 and S_2 is called a *linking* of size l from S_1 to S_2. A simple path in which the first and the last vertex coincide is called cycle; a *cycle family* of size l is a set of l mutually disjoint cycles. The length of a cycle family equals the total number of edges in the family.

Theorem 4.3.2 ***(Structural rank of a square matrix [88])*** *The structure N-dimensional matrix $s[\bar{E}] - [\bar{A}]$ is structurally degenerate if and only if there exists no cycle family of length N in $G(s[\bar{E}] - [\bar{A}])$.*

We are now able to state our main result on structural detectability.

Theorem 4.3.3 ***(Structurally undetectable attack)*** *Let the parameters space of the structured cyber-physical system* $([E],[A],[B],[C],[D])$ *define a polytope in* $\mathbb{R}^d$ *for some* $d \in \mathbb{N}_0$. *Assume that* $s[E]-[A]$ *is structurally non-degenerate. The system* $([E],[A],[B],[C],[D])$ *is structurally left-invertible if and only if there exists a linking of size* $|\mathcal{U}|$ *from* $\mathcal{U}$ *to* $\mathcal{Y}$.

Theorem 4.3.3 can be interpreted in the context of cyber-physical systems. Indeed, since $|sE-A| \neq 0$ by assumption (A1), and because of assumption (A4), Theorem 4.3.3 states that there exists a structural undetectable attack if and only if there is no linking of size $|\mathcal{U}|$ from $\mathcal{U}$ to $\mathcal{Y}$, provided that the network state at the failure time is known.

Proof: Because of Lemma 4.3.1, we need to show that, if there are $|\mathcal{U}|$ disjoint paths from $\mathcal{U}$ to $\mathcal{Y}$, then there exists admissible left-invertible realizations. Conversely, if there are at most $|\mathcal{U}|-1$ disjoint paths from $\mathcal{U}$ to $\mathcal{Y}$, then every admissible realization is not left-invertible.

(If) Let (E,A,B,C,D), with $|sE-A| \neq 0$, be an admissible realization, and suppose there exists a linking of size $|\mathcal{U}|$ from $\mathcal{U}$ to $\mathcal{Y}$. Without affecting generality, assume $|\mathcal{Y}| = |\mathcal{U}|$. For the left-invertibility property we need

$$\left|\begin{bmatrix} sE-A & -B \\ C & D \end{bmatrix}\right| = |sE-A| \left|D + C(sE-A)^{-1}B\right| \neq 0,$$

and hence we need $|D + C(sE - A)^{-1}B| \neq 0$. Notice that $D + C(sE - A)^{-1}B$ corresponds to the transfer matrix of the cyber-physical system. Since there are $|\mathcal{U}|$ independent paths from $\mathcal{U}$ to $\mathcal{Y}$, the matrix $D + C(sE - A)^{-1}B$ can be made nonsingular and diagonal by removing some connection lines from the network. In particular, for a given linking of size $|\mathcal{U}|$ from $\mathcal{U}$ to $\mathcal{Y}$, a nonsingular and diagonal transfer matrix is obtained by setting to zero the entries of E and A corresponding to the edges not in the linking. Then there exist admissible left-invertible realizations, and thus the system $([E], [A], [D], [C], [D])$ is structurally left-invertible.

(Only if) Take any subset of $|\mathcal{U}|$ output vertices, and let $|\mathcal{U}| - 1$ be the maximum size of a linking from $\mathcal{U}$ to $\mathcal{Y}$. Let $[\bar{E}]$ and $[\bar{A}]$ be such that $s[\bar{E}] - [\bar{A}] = \left[\begin{smallmatrix} s[E]-[A] & [B] \\ [C] & [D] \end{smallmatrix}\right]$. Consider the previously defined graph $G(s[\bar{E}] - [\bar{A}])$, and notice that a path from $\mathcal{U}$ to $\mathcal{Y}$ in the digraph associated with the structured system corresponds, possibly after relabeling the output variables, to a cycle in involving input/output vertices in $G(s[\bar{E}] - [\bar{A}])$. Observe that there are only $|\mathcal{U}| - 1$ such (disjoint) cycles. Hence, there is no cycle family of length N, being N the size of $[\bar{A}]$, and the statement follows from Theorem 4.3.2. □

To conclude this section, note that Theorem 4.3.3 extends [115] to regular descriptor systems with constraints on parameters.

4.3.3 Network vulnerability with unknown initial state

If the failure initial state is unknown, then a vulnerability is identified by the existence of a pair of initial conditions x_1 and x_2, and an attack u such that $y(x_1, 0, t) = y(x_2, u, t)$, or, equivalently, by the existence of invariant zeros for the given cyber-physical system. We will now show that, provided that a cyber-physical system is left-invertible, its invariant zeros can be computed by simply looking at an associated nonsingular state space system. Let the state vector x of the descriptor system (4.1) be partitioned as $[x_1^\mathsf{T} \; x_2^\mathsf{T}]^\mathsf{T}$, where x_1 corresponds to the dynamic variables. Let the network matrices E, A, B, C, and D be partitioned accordingly, and assume, without loss of generality, that E is given as $E = \text{blkdiag}(E_{11}, 0)$, where E_{11} is nonsingular. In this case, the descriptor model (4.1) reads as

$$\begin{aligned} E_{11}\dot{x}_1(t) &= A_{11}x_1(t) + B_1u(t) + A_{12}x_2(t)\,, \\ 0 &= A_{21}x_1(t) + A_{22}x_2(t) + B_2u(t)\,, \\ y(t) &= C_1x_1(t) + C_2x_2(t) + Du(t)\,. \end{aligned} \tag{4.7}$$

Consider now the associated nonsingular state space system which is obtained by regarding x_2 as an external input to the descriptor system (6.30) and the algebraic constraint as output:

$$\begin{aligned}\dot{x}_1(t) &= E_{11}^{-1}A_{11}x_1(t) + E_{11}^{-1}B_1u(t) + E_{11}^{-1}A_{12}x_2(t),\\ \tilde{y}(t) &= \begin{bmatrix} A_{21} \\ C_1 \end{bmatrix} x_1(t) + \begin{bmatrix} A_{22} & B_2 \\ C_2 & D \end{bmatrix} \begin{bmatrix} x_2(t) \\ u(t) \end{bmatrix}.\end{aligned} \tag{4.8}$$

Theorem 4.3.4 ***(Equivalence of invariant zeros)*** *Consider the descriptor system* (4.1) *partitioned as in* (6.30)*. Assume that, for the corresponding structured system* $([E],[A],[B],[C],[D])$*, there exists a linking of size* $|\mathcal{U}|$ *from* $\mathcal{U}$ *to* $\mathcal{Y}$*. Then, in almost all admissible realizations, the invariant zeros of the descriptor system* (6.30) *coincide with those of the associated nonsingular system* (6.31)*.*

Proof: From Theorem 4.3.3, the structured descriptor system $([E],[A],[B],[C],[D])$ is structurally left-invertible. Let (E,A,B,C,D) be a left-invertible realization. The proof now follows a procedure similar to [111, Proposition 8.4]. Let $s \in \mathbb{C}$ be an invariant zero for the nonsingular system (6.31) with state-zero direction $x_1 \neq 0$ and input-zero direction u, that is

$$\begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} = \underbrace{\left[\begin{array}{c|cc} sI - E_{11}^{-1}A_{11} & -E_{11}A_{12} & -E_{11}^{-1}B_1 \\ \hline A_{21} & A_{22} & B_2 \\ C_1 & C_2 & D \end{array}\right]}_{P_{\text{nonsingular}}(s)} \begin{bmatrix} x_1 \\ x_2 \\ u \end{bmatrix}.$$

A multiplication of the above equation by blkdiag$(E_{11}, -I, I)$ and a re-partioning of the resulting matrix yields

$$\begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} = \underbrace{\left[\begin{array}{cc|c} sE_{11} - A_{11} & -A_{12} & -B_1 \\ -A_{21} & -A_{22} & -B_2 \\ \hline C_1 & C_2 & D \end{array}\right]}_{P_{\text{singular}}(s)} \begin{bmatrix} x_1 \\ x_2 \\ u \end{bmatrix}. \qquad (4.9)$$

Since $x_1 \neq 0$, we also have $x = [x_1^\mathsf{T} \; x_2^\mathsf{T}]^\mathsf{T} \neq 0$. Then, equation (4.9) implies that $s \in \mathbb{C}$ is an invariant zero of the descriptor system (6.30) with state-zero direction $x \neq 0$ and input-zero direction u. We conclude that the invariant zeros of the nonsingular system (6.31) are a subset of the zeros of the descriptor system (6.30). In order to continue, suppose that there is $s \in \mathbb{C}$ which is an invariant zero of the descriptor system (6.30) but not of the nonsingular system (6.31). Let $x = [x_1^\mathsf{T} \; x_2^\mathsf{T}]^\mathsf{T} \neq 0$ and u be the associated state-zero and input-zero direction, respectively. Since $\text{Ker}(P_{\text{singular}}(s)) = \text{Ker}(P_{\text{nonsingular}}(s))$ and s is not a zero of the nonsingular system (6.31), it follows that $x_1 = 0$ and $x_2 \neq 0$. Accordingly, we have that

$$\text{Ker}\left(\begin{bmatrix} -A_{12} & -B_1 \\ -A_{22} & -B_2 \\ C_2 & D \end{bmatrix} \right) \neq \{\emptyset\}.$$

It follows that the vector $[0^\mathsf{T} \; x_2^\mathsf{T} \; u^\mathsf{T}]^\mathsf{T}$ lies in the nullspace of $P_{\text{singular}}(s)$ for each $s \in \mathbb{C}$, and thus the descriptor system (6.30) is not left-invertible. In conclusion, if the descriptor system (6.30) is left-invertible, then its invariant zeros coincide with those of the nonsingular system (6.31). □

It should be noticed that, because of Theorem 4.3.4, under the assumption of left-invertibility, classical linear systems results can be used to investigate the presence of structural undetectable attacks in a cyber-physical system; see [29] for a survey of results on generic properties of linear systems.

4.4 Illustrative Examples

4.4.1 A state attack against a power network

Consider the power network model analyzed in Example 1 and illustrated in Fig. 4.2, and let the variables θ_4 and θ_5 be affected, respectively, by the unknown and unmeasurable signals u_1 and u_2. Suppose that a monitoring unit is allowed to measure directly the state variables of the first generator, that is, $y_1 = \delta_1$ and $y_2 = \omega_1$. Notice from Fig. 4.4 that the maximum size of a linking from the failure to the output vertices is 1, so that, by Theorem 4.3.3, there exists a structural vulnerability. In other words, for every choice of the network matrices, there exist nonzero u_1 and u_2 that are not detectable via the measurements.[6]

[6] When these ouput-nulling inputs u_1, u_2 are regarded as additional loads, then they are entirely sustained by the second and third generator.

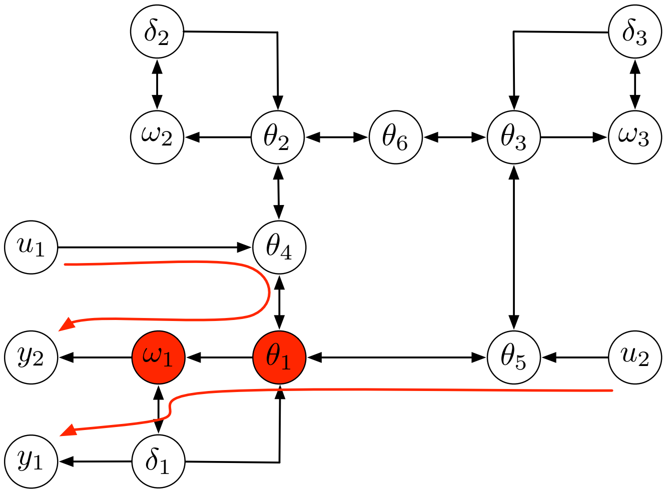


Figure 4.4: In the above network, there is no linking of size 2 from the input to the output vertices. Indeed, the vertices θ_1 and ω_1 belong to every path from $\{u_1, u_2\}$ to $\{y_1, y_2\}$. Two input to output paths are depicted in red.

We now consider a numerical realization of this system. Let the input matrices be $B = [e_8 \; e_9]$ and $D = [0 \; 0]^\mathsf{T}$, the measurement matrix be $C = [e_1 \; e_4]^\mathsf{T}$, and the system matrix A be as in equation (3.4) with $M_g = \text{blkdiag}(.125, .034, .016)$, $D_g = \text{blkdiag}(.125, .068, .048)$, and

$$\mathcal{L} = \begin{bmatrix} .058 & 0 & 0 & -.058 & 0 & 0 & 0 & 0 & 0 \\ 0 & .063 & 0 & 0 & -.063 & 0 & 0 & 0 & 0 \\ 0 & 0 & .059 & 0 & 0 & -.059 & 0 & 0 & 0 \\ -.058 & 0 & 0 & .235 & 0 & 0 & -.085 & -.092 & 0 \\ 0 & -.063 & 0 & 0 & .296 & 0 & -.161 & 0 & -.072 \\ 0 & 0 & -.059 & 0 & 0 & .330 & 0 & -.170 & -.101 \\ 0 & 0 & 0 & -.085 & -.161 & 0 & .246 & 0 & 0 \\ 0 & 0 & 0 & -.092 & 0 & -.170 & 0 & .262 & 0 \\ 0 & 0 & 0 & 0 & -.072 & -.101 & 0 & 0 & .173 \end{bmatrix}.$$

Let $U_1(s)$ and $U_2(s)$ be the Laplace transform of the attack signals u_1 and u_2, and let

$$\begin{bmatrix} U_1(s) \\ U_2(s) \end{bmatrix} = \underbrace{\begin{bmatrix} \frac{-1.024s^4 - 5.121s^3 - 10.34s^2 - 9.584s - 3.531}{s^4 + 5s^3 + 9.865s^2 + 9.173s + 3.531} \\ 1 \end{bmatrix}}_{\mathcal{N}(s)} \bar{U}(s),$$

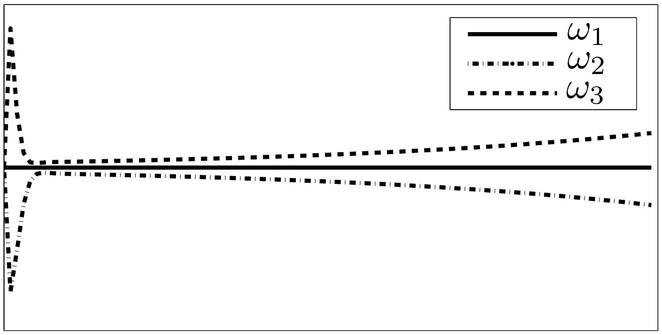


Figure 4.5: The velocities ω_2 and ω_3 are driven unstable by the signals u_1 and u_2, which are undetectable from the measurements of ω_1 and δ_1.

for *some arbitrary* nonzero signal $\bar{U}(s)$. Then it can be verified that the failure cannot be detected through the measurements y_1 and y_2. In fact, $\mathcal{N}(s)$ coincides with the null space of the input/output transfer matrix. An example is in Fig. 4.5, where the second and the third generators are driven unstable by the attack, yet the first generator does not deviate from the nominal operating condition.

Suppose now that the rotor angle of the first generator and the voltage angle at the 6-th bus are measured, that is, $C = \begin{bmatrix} e_1 \; e_{12} \end{bmatrix}^\mathsf{T}$. Then, there exists a linking of size 2 from $\mathcal{U}$ to $\mathcal{Y}$, and the system (E, A, B, C) is left-invertible. Following Theorem 4.3.4, the invariant zeros of the power network can be computed by looking at its reduced system, and they are $-1.6864 \pm 1.8070i$ and $-0.8136 \pm 0.2258i$. Consequently, if the network state is unknown at the failure time, there exists vulnerabilities that an attacker may exploit to affect the network while

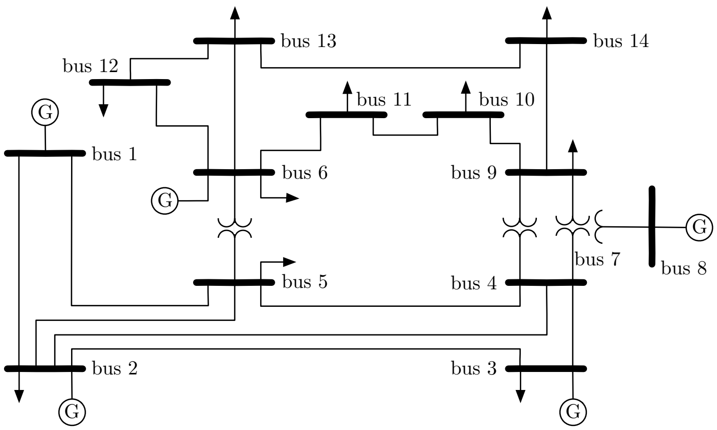


Figure 4.6: For the here represented IEEE 14 bus system, if the voltage angle of one bus is measured exactly, then a cyber attack against the measurements data is always detectable by our dynamic detection procedure. In contrary, as shown in [56], a cyber attack may remain undetected by a static procedure if it compromises as few as four measurements.

remaining undetected. Finally, we remark that such state attacks are entirely realizable by cyber attacks [70].

4.4.2 An output attack against a power network

Let the IEEE 14 bus power network (Fig. 4.6) be modeled as a descriptor system as in Section 3.1. Following [56], let the measurement matrix C consist of the real power injections at all buses, of the real power flows of all branches, and of one rotor angle (or one bus angle). We assume that an attacker can compromise all the measurements, independently of each other, except for one referring to the rotor angle.

Let $k \in \mathbb{N}_0$ be the cardinality of the attack set. It is known that an attack undetectable to a static detector exists if $k \geq 4$ [56]. In other words, due to the sparsity pattern of C, there exists a signal u_K, with (the same) four nonzero entries at all times, such that $Du_K(t) \in \operatorname{Im}(C)$ at all times. By Theorem 4.2.3 the attack set K remains undetected by a Static Detector through the attack mode u_K. On the other hand, following Theorem 4.2.5, it can be verified that, for the same output matrix C, and independent of the value of k, there exists *no* undetectable (output) attacks for a dynamic monitor.

It should be notice that this result relies on the fact that the rotor angle measurement is known to be correct, because, for instance, it is protected using sophisticated and costly security methods [64]. Since the state of the IEEE 14 bus system can be reconstructed by means of this measurement only (in a system theoretic sense, the system is observable by measuring one generator rotor angle), the output attack Du is easily identified as $Du(t) = y(t) - C\hat{x}(t)$, where $\hat{x}(t) = x(t)$ is the reconstructed system state at time t.

4.4.3 A state and output attack against a water supply network

Consider the water supply network EPANET 3 linearized at a steady state with non-zero pressure drops [91]. The water network model as well as a possible

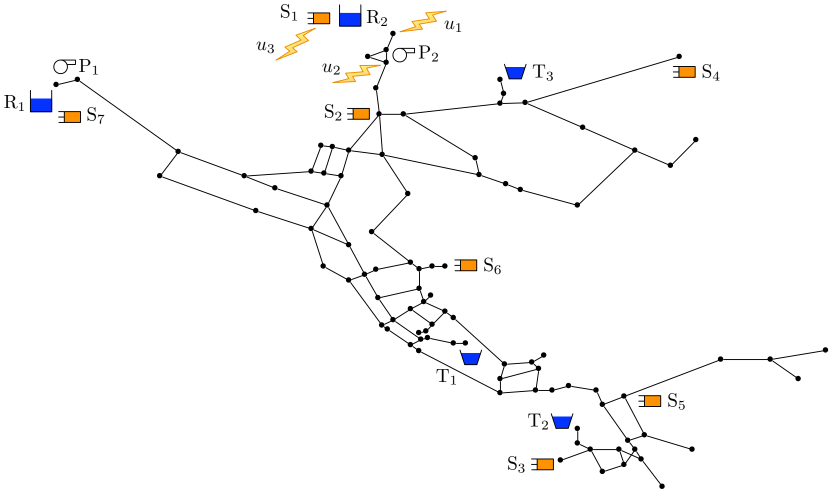


Figure 4.7: This figure shows the structure of the EPANET water supply network model # 3, which features 3 tanks (T$_1$, T$_2$, T$_3$), 2 reservoirs (R$_1$, R$_2$), 2 pumps (P$_1$, P$_2$), 96 junctions, and 119 pipes. Seven pressure sensors (S$_1, \dots,$ S$_7$) have been installed to monitor the network functionalities. A cyber-physical attack to steal water from the reservoir R$_2$ is reported. Notice that the cyber-physical attack features two state attacks (u_1, u_2) and one output attack (u_3).

cyber-physical attack are illustrated in Fig. 4.7. The considered cyber-physical attack aims at stealing water from the reservoir R$_2$ while remaining undetected from the installed pressure sensors S$_1, \dots,$ S$_7$. In order to achieve its goal, the attacker corrupts the measurements of sensor S$_1$ (output attack), it steals water from the reservoir R$_2$ (state attack), and, finally, it modifies the input of the control pump P$_2$ to restore the pressure drop due to the loss of water in R$_2$ (state attack). We now analyze this attack in more details.

Following the modeling in Section 3.2, an index-one descriptor model describing the evolution of the water network in Fig. 4.7 is computed. For notational convenience, let $x_1(t)$, $x_2(t)$, $x_3(t)$, and $x_4(t)$ denote, respectively, the pressure at time t at the reservoir R_2, at the reservoir R_1 and at the tanks T_1, T_2 and T_3, at the junction P_2, and at the remaining junctions. The index-one descriptor model reads as

$$\begin{bmatrix} \dot{x}_1(t) \\ M\dot{x}_2(t) \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & A_{22} & 0 & A_{24} \\ A_{31} & 0 & A_{33} & A_{34} \\ 0 & A_{42} & A_{43} & A_{44} \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \\ x_4(t) \end{bmatrix},$$

where the pattern of zeros is due to the network interconnection structure, and $M = \mathrm{diag}(1, A_1, A_2, A_3)$ corresponds to the dynamics of the reservoir R_1 and the tanks T_1, T_2, and T_3. With the same partitioning, the attack signature reads as $B = \begin{bmatrix} B_1 \; B_2 \; 0 \end{bmatrix}$ and $D = \begin{bmatrix} 0 \; 0 \; D_1 \end{bmatrix}$, where

$$B_1 = \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix}^{\mathsf{T}}, \; B_2 = \begin{bmatrix} 0 & 0 & 1 & 0 \end{bmatrix}^{\mathsf{T}}, \text{ and } D_1 = \begin{bmatrix} 1 & 0 & \dots & 0 \end{bmatrix}^{\mathsf{T}}.$$

Let the attack u_2 be chosen as $u_2(t) = -A_{31}x_1(t)$. Then, the state variables x_2, x_3, and x_4 are decoupled from x_1. Consequently, the attack mode u_1 does not affect the dynamics of x_2, x_3, and x_4. Let $u_1 = -1$, and notice that the pressure x_1 decreases with time (that is, water is being removed from R_2). Finally, for the attack to be undetectable, since the state variable x_1 is continuously monitored by S_1, let $u_3 = -x_1$. It can be verified that the proposed attack strategy allows an attacker to steal water from the reservoir R_2 while remaining undetected from the sensors measurements. In other words, the attack (Bu, Du), with $u = \left[u_1^\mathsf{T}\ u_2^\mathsf{T}\ u_3^\mathsf{T}\right]^\mathsf{T}$, excites only zero dynamics for the water network system in Fig. 4.7.

We conclude this section with the following remarks. First, for the implementation of the proposed attack strategy, neither the network initial state, nor the network structure besides A_{31} need to be known to the attacker. Second, the effectiveness of the proposed attack strategy is independent of the sensors measuring the variables x_3 and x_4. On the other hand, if additional sensors are used to measure the flow between the reservoir R_2 and the pump P_2, then an attacker would need to corrupt these measurements as well to remain undetected. Third and finally, due to the reliance on networks to control actuators in cyber-physical systems, the attack u_2 on the pump P_2 could be generated by a cyber attack [70].

Chapter 5

Static Monitors for State Estimation and False Data Detection

In this chapter we design a distributed static monitor for state estimation and false data detection. We start by introducing the mathematical models and the necessary notation. We focus on power networks, although our method applies to general models described by systems of linear equations.

5.1 Problem Setup

For a power network, an example of which is reported in Fig. 5.1, the state at a certain instant of time consists of the voltage angles and magnitudes at all the system buses. The (static) state estimation problem introduced in the seminal work by Schweppe [98] refers to the procedure of estimating the state of a power

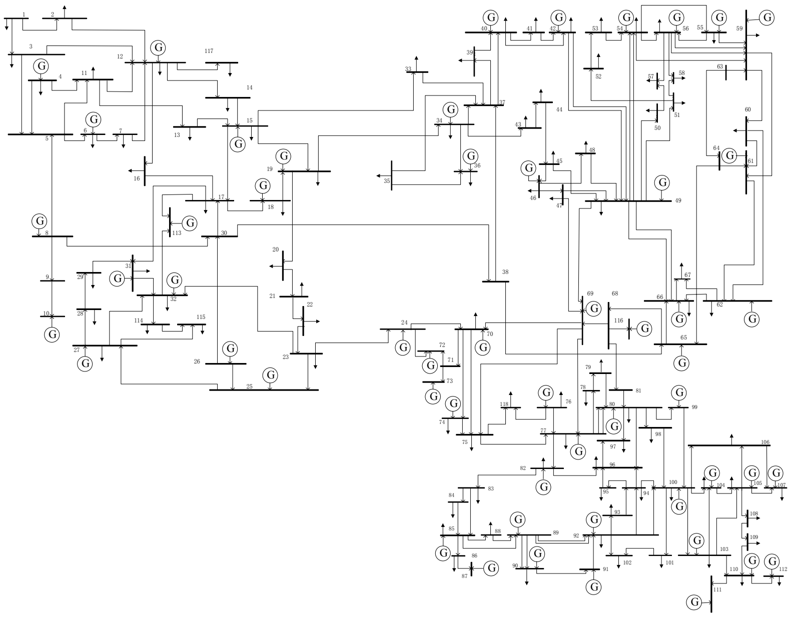


Figure 5.1: This figure shows the diagram of the IEEE 118 bus system (courtesy of the IIT Power Group). The network has 118 buses, 186 branches, 99 loads, and 54 generators.

network given a set of measurements of the network variables, such as voltages, currents, and power flows along the transmission lines. To be more precise, let $x \in \mathbb{R}^n$ and $z \in \mathbb{R}^p$ be, respectively, the state and measurements vectors. Then, the vectors x and z are related by the relation

$$z = h(x) + \eta, \tag{5.1}$$

where $h(\cdot)$ is a nonlinear measurement function, and where η, which is traditionally assumed to be a zero mean random vector satisfying $\mathbb{E}[\eta\eta^{\mathsf{T}}] = \Sigma_\eta = \Sigma_\eta^{\mathsf{T}} > 0$, is the measurements noise. An optimal estimate of the network state coincides with the *most likely* vector $\hat{x}$ that solves equation (5.1). It should be observed that, instead of by solving the above estimation problem, the network state could be obtained by measuring directly the voltage phasors by means of phasor measurement devices.[1] Such an approach, however, would be economically expensive, since it requires a phasor measurement device at each network bus, and it would be very vulnerable to communication failures [1]. In this work, we adopt the approximated estimation model presented in [99], which follows from the linearization around an operating point of equation (5.1). Specifically, we have

$$z = Hx + v, \tag{5.2}$$

where $H \in \mathbb{R}^{p \times n}$ and where v, the measurements noise, is such that $\mathbb{E}[v] = 0$ and $E[vv^\mathsf{T}] = \Sigma = \Sigma^\mathsf{T} > 0$. Observe that, because of the interconnection structure of a power network, the measurement matrix H is usually sparse. Let $\mathrm{Ker}(H)$ denote the null space of H, and assume $\mathrm{Ker}(H) = \{0\}$. Recall from [59] that the vector

$$x_{\mathrm{wls}} = (H^\mathsf{T}\Sigma^{-1}H)^{-1}H^\mathsf{T}\Sigma^{-1}z \tag{5.3}$$

minimizes the weighted variance of the estimation error, i.e., $x_{\mathrm{wls}} = \arg\min_{\hat{x}}(z - H\hat{x})^\mathsf{T}\Sigma^{-1}(z - H\hat{x})$.

[1] Phasor measurement units are devices that synchronize by using GPS signals, and that allow for a direct measurement of voltage and current phasors.

The centralized computation of x_{wls} assumes the complete knowledge of the matrices H and Σ, and it requires the inversion of the matrix $H^{\mathsf{T}}\Sigma^{-1}H$. For a large power network, such computation imposes a limitation on the dimension of the matrix H, and hence on the number of measurements that can be efficiently processed to obtain a real-time state estimate. Since the performance of network control and optimization algorithms depend upon the precision of the state estimate, a limitation on the number of network measurements constitutes a bottleneck toward the development of a more efficient power grid. A possible solution to address this complexity problem is to distribute the computation of x_{wls} among geographically deployed control centers (monitors), in a way that each monitor is responsible for a subpart of the whole network. To be more precise, let the matrices H and Σ, and the vector z be partitioned as[2]

$$H = \begin{bmatrix} H_1 \\ H_2 \\ \vdots \\ H_m \end{bmatrix}, \quad \Sigma = \begin{bmatrix} \Sigma_1 \\ \Sigma_2 \\ \vdots \\ \Sigma_m \end{bmatrix}, \quad z = \begin{bmatrix} z_1 \\ z_2 \\ \vdots \\ z_m \end{bmatrix}, \tag{5.4}$$

where, for $i \in \{1, \dots, m\}$, $m_i \in \mathbb{N}$, $H_i \in \mathbb{R}^{m_i \times n}$, $\Sigma_i \in \mathbb{R}^{m_i \times p}$, $z_i \in \mathbb{R}^{m_i}$, and $\sum_{i=1}^{m} m_i = p$. Let $G = (V, \mathcal{E})$ be the connected graph in which each vertex $i \in V = \{1, \dots, m\}$ denotes a monitor, and $\mathcal{E} \in V \times V$ denotes the set of monitors

[2] In most application the error covariance matrix is assumed to be diagonal, so that each submatrix Σ_i is very sparse. However, we do not impose any particular structure on the error covariance matrix.

interconnections. For $i \in \{1, \dots, m\}$, assume that monitor i knows the matrices H_i, Σ_i, and the vector z_i, and that two neighboring monitors are allowed to cooperate by exchanging information. Notice that, if the full matrices H and Σ are nowhere available, and if they cannot be used for the computation of x_{wls}, then, with no cooperation among the monitors, the vector x_{wls} cannot be computed by any of the monitor. Hence we consider the following problem.

Problem 1 (Distributed state estimation) *Design an algorithm for the monitors to compute the minimum variance estimate of the network state via distributed computation.*

We now introduce the second problem addressed in this work. Given the distributed nature of a power system and the increasing reliance on local area networks to transmit data to control centers, there exists the possibility for an attacker to compromise the network functionalities by corrupting the measurements vector. When a malignant agent corrupts some of the measurements, the state to measurements relation becomes

$$z = Hx + v + w,$$

where the vector $w \in \mathbb{R}^p$ is chosen by the attacker, and, consequently, it is unknown and unmeasurable by any of the monitoring stations. We refer to the vector

w to as *false* data. From the above equation, it should be observed that there exist vectors w that cannot be detected through the measurements z. For instance, if the false data vector is intentionally chosen such that $w \in \mathrm{Im}(H)$, then the attack cannot be detected through the measurements z. Indeed, denoting with † the pseudoinverse operation, the vector $x + H^\dagger w$ is a valid network state. We assume that the vector w is detectable from the measurements z, and we consider the following problem.

Problem 2 (Distributed detection) *Design an algorithm for the monitors to detect the presence of false data in the measurements via distributed computation.*

As it will be clear in the sequel, the complexity of our methods depends upon the dimension of the state, as well as the number of monitors. In particular, few monitors should be used in the absence of severe computation and communication contraints, while many monitors are preferred otherwise. We believe that a suitable choice of the number of monitors depends upon the specific scenario, and it is not further discussed in this work.

Remark 3 (Generality of our methods) *In this thesis we focus on the state estimation and the false data detection problems for power systems. The methods described in the following sections, however, are general, and they have applicability beyond the power network scenario. For instance, our procedures can be used* *for state estimation and false data detection in dynamical system, as described in [79] for the case of sensors networks.*

5.2 Distributed State Estimation and False Data Detection

The objective of this section is the design of distributed methods to compute an optimal state estimate from measurements. With respect to a centralized method, in which a powerful central processor is in charge of processing all the data, our procedures require the computing units to have access to only a subset of the measurements, and are shown to reduce significantly the computational burden. In addition to being convenient for the implementation, our methods are also optimal, in the sense that they maintain the same estimation accuracy of a centralized method.

For a distributed method to be implemented, the interaction structure among the computing units needs to be defined. Here we consider two modes of cooperations among the computing units, and, namely, the *incremental* and the *diffusive* interactions. In an incremental mode of cooperation, information flows in a sequential manner from one node to the adjacent one. This setting, which usually requires the least amount of communications [84], induces a cyclic interaction graph among the processors. In a diffusive strategy, instead, each node exchanges information with all (or a subset of) its neighbors as defined by an interaction graph. In this case, the amount of communication and computation is higher than in the incremental case, but each node possesses a good estimate before the termination of the algorithm, since it improves its estimate at each communication round. This section is divided into three parts. In Section 5.2.2, we first develop a distributed incremental method to compute the minimum norm solution to a set of linear equations, and then exploit such method to solve a minimum variance estimation problem. In Section 5.2.3 we derive a diffusive strategy which is amenable to asynchronous implementation. Finally, in Section 5.2.4 we propose a distributed algorithm for the detection of false data among the measurements. Our detection procedure requires the computation of the minimum variance state estimate, for which either the incremental or the diffusive strategy can be used.

5.2.1 Incremental solution to a set of linear equations

We start by introducing a distributed incremental procedure to compute the minimum norm solution to a set of linear equations. This procedure constitutes the key ingredient of the incremental method we later propose to solve a minimum variance estimation problem. Let $H \in \mathbb{R}^{p \times m}$ and let $z \in \operatorname{Im}(H)$, where $\operatorname{Im}(H)$ denotes the range space spanned by the matrix H. Consider the system of linear

Algorithm 1: Incremental minimum norm solution (i-th monitor)

```
Input    : Matrices $H_i$, $z_i$;
Require  : $\left[z_1^\mathsf{T} \, \dots \, z_m^\mathsf{T}\right]^\mathsf{T} \in \mathrm{Im}(\left[H_1^\mathsf{T} \, \dots \, H_m^\mathsf{T}\right]^\mathsf{T})$;

if $i = 1$ then $\hat{x}_0 := 0$, $K_0 := I_n$;
else receive $\hat{x}_{i-1}$ and $K_{i-1}$ from monitor $i-1$;
$\hat{x}_i := \hat{x}_{i-1} + K_{i-1}(H_i K_{i-1})^\dagger (z_i - H_i \hat{x}_{i-1})$;
$K_i := \mathrm{Basis}(K_{i-1}\,\mathrm{Ker}(H_i K_{i-1}))$;
if $i < m$ then transmit $\hat{x}_i$ and $K_i$ to monitor $i+1$;
else return $\hat{x}_m$;
```

equations $z = Hx$, and recall that the unique minimum norm solution to $z = Hx$ coincides with the vector $\hat{x}$ such that $z = H\hat{x}$ and $\|\hat{x}\|_2$ is minimum. It can be shown that $\|\hat{x}\|_2$ being minimum corresponds to $\hat{x}$ being orthogonal to the null space $\mathrm{Ker}(H)$ [59]. Let H and z be partitioned in m blocks as in (5.4), and let $G = (V, \mathcal{E})$ be a directed graph such that $V = \{1, \dots, m\}$ corresponds to the set of monitors, and, denoting with (i, j) the directed edge from j to i, $\mathcal{E} = \{(i+1, i) : i = 1, \dots, m-1\} \cup \{(1, m)\}$. Our incremental procedure to compute the minimum norm solution to $z = H\hat{x}$ is in Algorithm 1, where, given a subspace $\mathcal{V}$, we write $\mathrm{Basis}(\mathcal{V})$ to denote any full rank matrix whose columns span the subspace $\mathcal{V}$. We now proceed with the analysis of the convergence properties of the *Incremental minimum norm solution* algorithm.

Theorem 5.2.1 (Convergence of Algorithm 1) *Let $z = Hx$, where H and z are partitioned in m row-blocks as in* (5.4)*. In Algorithm 1, the m-th monitor returns the vector $\hat{x}$ such that $z = H\hat{x}$ and $\hat{x} \perp \operatorname{Ker}(H)$.*

Proof: See Section 5.5.1. □

It should be observed that the dimension of K_i decreases, in general, when the index i increases. In particular, $K_m = \{0\}$ and $K_1 = \operatorname{Ker}(H_1)$. To reduce the communication burden of the algorithm, monitor i could transmit the smallest among $\operatorname{Basis}(K_{i-1} \operatorname{Ker}(H_i K_{i-1}))$ and $\operatorname{Basis}(K_{i-1} \operatorname{Ker}(H_i K_{i-1})^{\perp})$, together with a packet containing the type of the transmitted basis.

Remark 4 (Computational complexity of Algorithm 1) *In Algorithm 1, the main operation to be performed by the i-th agent is a singular value decomposition (SVD).*[3] *Indeed, since the range space and the null space of a matrix can be obtained through its SVD, both the matrices $(H_i K_{i-1})^{\dagger}$ and $\operatorname{Basis}(K_{i-1} \operatorname{Ker}(H_i K_{i-1}))$ can be recovered from the SVD of $H_i K_{i-1}$. Let $H \in \mathbb{R}^{m \times n}$, $m > n$, and assume the presence of $\lceil m/k \rceil$ monitors, $1 \leq k \leq m$. Recall that, for a matrix $M \in \mathbb{R}^{k \times p}$, the singular value decomposition can be performed with complexity*

[3] *The matrix H is usually very sparse, since it reflects the network interconnection structure. If the matrices $H_i K_{i-1}$ are also sparse, then efficient SVD algorithms for very large sparse matrices can be employed (cf.* SVDPACK*). In general, if the dimension of $H_i K_{i-1}$ is too large for an SVD algorithm to be numerically reliable, then additional monitors should be used to reduce the dimension of $H_i K_{i-1}$.*

$O(\min\{kp^2, k^2p\})$ [37]. Hence, the computational complexity of computing a minimum norm solution to the system $z = Hx$ is $O(mn^2)$. In Table 5.1 we report the computational complexity of Algorithm 1 as a function of the block size k.

Table 5.1: *Computational complexity of Algorithm 1.*

Block size	*i-th complexity*	*Total complexity*
$k \leq n$	$O(k^2n)$	$O(mkn)$
$k > n$	$O(kn^2)$	$O(mn^2)$

The following observations are in order. First, if $k \leq n$, then the computational complexity sustained by the i-th monitor is much smaller than the complexity of a centralized implementation, i.e., $O(k^2n) \ll O(mn^2)$. Second, the complexity of the entire algorithm is optimal, since, in the worst case, it maintains the computational complexity of a centralized solution, i.e., $O(mkn) \leq O(mn^2)$. Third and finally, a compromise exists between the blocks size k and the number of communications needed to terminate Algorithm 1. In particular, if $k = m$, then no communication is needed, while, if $k = 1$, then $m - 1$ communication rounds are necessary to terminate the estimation algorithm.[4]

[4] *Additional $m - 1$ communication rounds are needed to transmit the estimate to all monitors.*

5.2.2 Incremental estimation via distributed computation

We now focus on the computation of the weighted least squares solution to a set of linear equations. Let v be an unknown and unmeasurable random vector, with $\mathbb{E}(v) = 0$ and $\mathbb{E}(vv^{\mathsf{T}}) = \Sigma = \Sigma^{\mathsf{T}} > 0$. Consider the system of equations

$$z = Hx + v, \tag{5.5}$$

and assume $\text{Ker}(H) = 0$. Notice that, because of the noise vector v, we generally have $z \notin \text{Im}(H)$, so that Algorithm 1 cannot be directly employed to compute the vector x_{wls} defined in (5.3). It is possible, however, to recast the above weighted least squares estimation problem to be solvable with Algorithm 1. Note that, because the matrix Σ is symmetric and positive definite, there exists[5] a full row rank matrix B such that $\Sigma = BB^{\mathsf{T}}$. Then, equation (5.5) can be rewritten as

$$z = \begin{bmatrix} H & \varepsilon B \end{bmatrix} \begin{bmatrix} x \\ \bar{v} \end{bmatrix}, \tag{5.6}$$

where $\varepsilon \in \mathbb{R}_{>0}$, $\mathbb{E}\left[\bar{v}\right] = 0$ and $\mathbb{E}\left[\bar{v}\bar{v}^{\mathsf{T}}\right] = \varepsilon^{-2} I$. Observe that, because B has full row rank, the system (5.6) is underdetermined, i.e., $z \in \text{Im}(\begin{bmatrix} H & \varepsilon B \end{bmatrix})$ and $\text{Ker}(\begin{bmatrix} H & \varepsilon B \end{bmatrix}) \neq 0$. Let

$$\begin{bmatrix} \hat{x}(\varepsilon) \\ \hat{\bar{v}} \end{bmatrix} = \begin{bmatrix} H & \varepsilon B \end{bmatrix}^{\dagger} z. \tag{5.7}$$

[5] Choose for instance $B = W\Lambda^{1/2}$, where W is a basis of eigenvectors of Σ and Λ is the corresponding diagonal matrix of the eigenvalues.

The following theorem characterizes the relation between the minimum variance estimate x_{wls} and $\hat{x}(\varepsilon)$.

Theorem 5.2.2 (Convergence with ε) *Consider the system of linear equations $z = Hx + v$. Let $\mathbb{E}(v) = 0$ and $\mathbb{E}(vv^{\mathsf{T}}) = \Sigma = BB^{\mathsf{T}} > 0$ for a full row rank matrix B. Let*

$$C = \varepsilon(I - HH^{\dagger})B, \qquad E = I - C^{\dagger}C,$$

$$D = \varepsilon E[I + \varepsilon^2 EB^{\mathsf{T}}(HH^{\mathsf{T}})^{\dagger}BE]^{-1}B^{\mathsf{T}}(HH^{\mathsf{T}})^{\dagger}(I - \varepsilon BC^{\dagger}).$$

Then

$$\begin{bmatrix} H & \varepsilon B \end{bmatrix}^{\dagger} = \begin{bmatrix} H^{\dagger} - \varepsilon H^{\dagger}B(C^{\dagger} + D) \\ C^{\dagger} + D \end{bmatrix};$$

and

$$\lim_{\varepsilon \to 0^+} H^{\dagger} - \varepsilon H^{\dagger}B(C^{\dagger} + D) = (H^{\mathsf{T}}\Sigma^{-1}H)^{-1}H^{\mathsf{T}}\Sigma^{-1}.$$

Proof: See Section 5.5.2. □

Throughout the proof, let $\hat{x}(\varepsilon)$ be the vector defined in (5.7), and notice that Theorem 5.2.2 implies that

$$x_{\mathrm{wls}} = \lim_{\varepsilon \to 0^+} \hat{x}(\varepsilon).$$

Remark 5 (Incremental state estimation) *For the system of equations $z = Hx + v$, let BB^T be the covariance matrix of the noise vector v, and let*

$$H = \begin{bmatrix} H_1 \\ H_2 \\ \vdots \\ H_m \end{bmatrix}, \quad B = \begin{bmatrix} B_1 \\ B_2 \\ \vdots \\ B_m \end{bmatrix}, \quad z = \begin{bmatrix} z_1 \\ z_2 \\ \vdots \\ z_m \end{bmatrix}, \tag{5.8}$$

where $m_i \in \mathbb{N}$, $H_i \in \mathbb{R}^{m_i \times n}$, $B_i \in \mathbb{R}^{m_i \times p}$, and $z_i \in \mathbb{R}^{m_i}$. For $\varepsilon > 0$, the estimate $\hat{x}(\varepsilon)$ of the weighted least squares solution to $z = Hx + v$ can be computed by means of Algorithm 1 with input $\begin{bmatrix} H_i \ \varepsilon B_i \end{bmatrix}$ and z_i.

Observe now that the estimate $\hat{x}(\varepsilon)$ coincides with $\hat{x}_{\text{wls}}$ only in the limit for $\varepsilon \to 0^+$. When the parameter ε is fixed, the estimate $\hat{x}(\varepsilon)$ differs from the minimum variance estimate $\hat{x}_{\text{wls}}$. We next characterize the approximation error $x_{\text{wls}} - \hat{x}(\varepsilon)$.

Corollary 5.2.1 (Approximation error) *Consider the system $z = Hx + v$, and let $\mathbb{E}\left[vv^\mathsf{T}\right] = BB^\mathsf{T}$ for a full row rank matrix B. Then*

$$x_{\text{wls}} - \hat{x}(\varepsilon) = \varepsilon H^\dagger B D z,$$

where D is as in Theorem 5.2.2.

Proof: With the same notation as in the proof of Theorem 5.2.2, for every value of $\varepsilon > 0$, the difference $x_{\text{wls}} - \hat{x}(\varepsilon)$ equals

$$\left((H^\mathsf{T}\Sigma^{-1}H)^{-1}H^\mathsf{T}\Sigma^{-1} - H^\dagger + \varepsilon H^\dagger B(C^\dagger + D)\right) z.$$

Since $(H^{\mathsf{T}}\Sigma^{-1}H)^{-1}H^{\mathsf{T}}\Sigma^{-1} - H^{\dagger} + \varepsilon H^{\dagger}BC^{\dagger} = 0$ for every $\varepsilon > 0$, it follows $x_{\text{wls}} - \hat{x}(\varepsilon) = \varepsilon H^{\dagger}BDz$. □

Therefore, for the solution of system (5.5) by means of Algorithm 1, the parameter ε is chosen according to Corollary 5.2.1 to meet a desired estimation accuracy. It should be observed that, even if the entire matrix H needs to be known for the computation of the exact parameter ε, the advantages of our estimation technique are preserved. Indeed, if the matrix H is unknown and an upper bound for $\|H^{\dagger}BDz\|$ is known, then a value for ε can still be computed that guarantees the desired estimation accuracy. On the other hand, if H is entirely known, it may be inefficient to use H to perform a centralized state estimation over time. Instead, a suitable parameter ε needs to be computed only once. To conclude this section we characterize the estimation residual $z - H\hat{x}$. This quantity plays an important role for the synthesis of a distributed false data detection algorithm.

Corollary 5.2.2 (Estimation residual) *Consider the system $z = Hx + v$, and let $\mathbb{E}\left[vv^{\mathsf{T}}\right] = \Sigma = \Sigma^{\mathsf{T}} > 0$. Then*[6]

$$\lim_{\varepsilon \to 0^+} \|z - H\hat{x}(\varepsilon)\| \leq \|(I - HW)\|\|v\|,$$

where $W = (H^{\mathsf{T}}\Sigma^{-1}H)^{-1}H^{\mathsf{T}}\Sigma^{-1}$.

[6] *Given a vector v and a matrix H, we denote by $\|v\|$ any vector norm, and by $\|H\|$ the corresponding induced matrix norm.*

Proof: By virtue of Theorem 5.2.2 we have

$$\lim_{\varepsilon \to 0^+} \hat{x}(\varepsilon) = x_{\text{wls}} = (H^\mathsf{T} \Sigma^{-1} H)^{-1} H^\mathsf{T} \Sigma^{-1} z = W z.$$

Observe that $HWH = H$, and recall that $z = Hx + v$. For any matrix norm, we have

$$\begin{aligned} \|z - Hx_{\text{wls}}\| &= \|z - HWz\| = \|(I - HW)(Hx + v)\| \\ &= \|Hx - Hx + (I - HW)v\| \\ &\leq \|(I - HW)\| \|v\|, \end{aligned}$$

and the theorem follows. □

5.2.3 Diffusive estimation via distributed computation

The implementation of the incremental state estimation algorithm described in Section 5.2.2 requires a certain degree of coordination among the control centers. For instance, an ordering of the monitors is necessary, such that the i-th monitor transmits its estimate to the $(i+1)$-th monitor. This requirement imposes a constraint on the monitors interconnection structure, which may be undesirable, and, potentially, less robust to link failures. In this section, we overcome this limitation by presenting a diffusive implementation of Algorithm 1, which only requires the monitors interconnection structure to be connected.[7] To be more

[7] An undirected graph is said to be connected if there exists a path between any two vertices [36].

precise, let $\mathcal{V} = \{1, \dots, m\}$ be the set of monitors, and let $G = (\mathcal{V}, \mathcal{E})$ be the undirected graph describing the monitors interconnection structure, where $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$, and $(i, j) \in \mathcal{E}$ if and only if the monitors i and j are connected. The neighbor set of node i is defined as $\mathcal{N}_i = \{j \in V : (i, j) \in \mathcal{E}\}$. We assume that G is connected, and we let the distance between two vertices be the minimum number of edges in a path connecting them. Finally, the diameter of a graph G, in short $\operatorname{diam}(G)$, equals the greatest distance between any pair of vertices. Our diffusive procedure is described in Algorithm 2, where the matrices H_i and εB_i are as defined in equation (5.8). During the h-th iteration of the algorithm, monitor i, with $i \in \{1, \dots, N\}$, performs the following three actions in order:

(i) transmits its current estimates $\hat{x}_i$ and K_i to all its neighbors;

(ii) receives the estimates $\hat{x}_j$ from neighbors $\mathcal{N}_i$; and

(iii) updates $\hat{x}_i$ and K_i as in the *for* loop of Algorithm 2.

We now show the convergence properties of Algorithm 2.

Theorem 5.2.3 (Convergence of Algorithm 2) *Consider the system of linear equations $z = Hx + v$, where $\mathbb{E}[v] = 0$ and $\mathbb{E}[vv^{\mathsf{T}}] = BB^{\mathsf{T}}$. Let H, B and z be partitioned as in* (5.8)*, and let $\varepsilon > 0$. Let the monitors communication graph be connected, let d be its diameter, and let the monitors execute the Diffusive state*

Algorithm 2: Diffusive state estimation (i-th monitor)

```
Input      : Matrices $H_i$, $\varepsilon B_i$, $z_i$;

$\hat{x}_i := [H_i \; \varepsilon B_i]^\dagger z_i$;
$K_i :=$ Basis(Ker($[H_i \; \varepsilon B_i]$));
while $K_i \neq 0$ do
    for $j \in \mathcal{N}_i$ do
        receive $\hat{x}_j$ and $K_j$;
        $\hat{x}_i := \hat{x}_i + [K_i \;\; 0][-K_i \;\; K_j]^\dagger(\hat{x}_i - \hat{x}_j)$;
        $K_i :=$ Basis(Im($K_i$) $\cap$ Im($K_j$));
    transmit $\hat{x}_i$ and $K_i$;
```

estimation algorithm. Then, each monitor computes the estimate $\hat{x}(\varepsilon)$ of x in d steps.

Proof: Let $\hat{x}_i$ be the estimate of the monitor i, and let K_i be such that $\begin{bmatrix}x^\mathsf{T} \; v^\mathsf{T}\end{bmatrix}^\mathsf{T} - \hat{x}_i \in \operatorname{Im}(K_i)$, where x denotes the network state, v is the measurements noise vector, and $\hat{x}_i \perp \operatorname{Im}(K_i)$. Notice that $z_i = \begin{bmatrix}H_i \;\; \varepsilon B_i\end{bmatrix}\hat{x}_i$, where z_i it the i-th measurements vector. Let i and j be two neighboring monitors. Notice that there exist vectors v_i and v_j such that $\hat{x}_i + K_i v_i = \hat{x}_j + K_j v_j$. In particular, those vectors can be chosen as

$$\begin{bmatrix} v_i \\ v_j \end{bmatrix} = [-K_i \; K_j]^\dagger(\hat{x}_i - \hat{x}_j).$$

It follows that the vector

$$\hat{x}_i^+ = \hat{x}_i + [K_i \;\; 0][-K_i \;\; K_j]^\dagger (\hat{x}_i - \hat{x}_j)$$

is such that $z_i = \begin{bmatrix} H_i & \varepsilon B_i \end{bmatrix} \hat{x}_i^+$ and $z_j = \begin{bmatrix} H_j & \varepsilon B_j \end{bmatrix} \hat{x}_i^+$. Moreover we have $\hat{x}_i^+ \perp (\mathrm{Im}(K_i) \cap \mathrm{Im}(K_j))$. Indeed, notice that

$$\begin{bmatrix} v_i \\ \\ v_j \end{bmatrix} \perp \mathrm{Ker}(\begin{bmatrix} -K_i & K_j \end{bmatrix}) \supseteq \left\{ \begin{bmatrix} w_i \\ \\ w_j \end{bmatrix} : K_i w_i = K_j w_j \right\}.$$

We now show that $K_i v_i \perp \mathrm{Im}(K_j)$. By contradiction, if $K_i v_i \not\perp \mathrm{Im}(K_j)$, then $v_i = \tilde{v}_i + \bar{v}_i$, with $K_i \tilde{v}_i \perp \mathrm{Im}(K_j)$ and $K_i \bar{v}_i \in \mathrm{Im}(K_j)$. Let $\bar{v}_j = K_j^\dagger K_i \bar{v}_i$, and $\tilde{v}_j = v_j - \bar{v}_j$. Then, $\begin{bmatrix} \bar{v}_i^\mathsf{T} & \bar{v}_j^\mathsf{T} \end{bmatrix}^\mathsf{T} \in \mathrm{Ker}(\begin{bmatrix} -K_i & K_j \end{bmatrix})$, and hence $\begin{bmatrix} v_i^\mathsf{T} & v_j^\mathsf{T} \end{bmatrix}^\mathsf{T} \not\perp \mathrm{Ker}(\begin{bmatrix} -K_i & K_j \end{bmatrix})$, which contradicts the hypothesis. We conclude that $\begin{bmatrix} K_i & 0 \end{bmatrix} \begin{bmatrix} -K_i & K_j \end{bmatrix}^\dagger (\hat{x}_i - \hat{x}_j) \perp \mathrm{Im}(K_j)$, and, since $\hat{x}_i \perp \mathrm{Im}(K_i)$, it follows $\hat{x}_i^+ \perp (\mathrm{Im}(K_i) \cap \mathrm{Im}(K_j))$. The theorem follows from the fact that after a number of steps equal to the diameter of the monitors communication graph, each vector $\hat{x}_i$ verifies all the measurements, and $\hat{x}_i \perp \mathrm{Im}(K_1) \cap \cdots \cap \mathrm{Im}(K_m)$. □

As a consequence of Theorem 5.2.2, in the limit for ε to zero, Algorithm 2 returns the minimum variance estimate of the state vector, being therefore the diffusive counterpart of Algorithm 1. A detailed comparison between incremental and diffusive methods is beyond the purpose of this work, and we refer the interested reader to [57, 58] and the references therein for a thorough discussion.

Here we only underline some key differences. While Algorithm 1 requires less operations, being therefore computationally more efficient, Algorithm 2 does not constraint the monitors communication graph. Additionally, Algorithm 2 can be implemented adopting general asynchronous communication protocols. For instance, consider the *Asynchronous (diffusive) state estimation* algorithm, where, at any given instant of time, at most one monitor, say j, sends its current estimates to its neighbors, and where, for $i \in N_j$, monitor i performs the following operations:

(i) $\hat{x}_i := \hat{x}_i + \begin{bmatrix} K_i & 0 \end{bmatrix} \begin{bmatrix} -K_i & K_j \end{bmatrix}^\dagger (\hat{x}_i - \hat{x}_j)$,

(ii) $K_i := \text{Basis}(\text{Im}(K_i) \cap \text{Im}(K_j))$.

Corollary 5.2.3 (Asynchronous estimation) *Consider the system of linear equations $z = Hx + v$, where $\mathbb{E}\left[v\right] = 0$ and $\mathbb{E}\left[vv^\mathsf{T}\right] = BB^\mathsf{T}$. Let H, B and z be partitioned as in* (5.8)*, and let $\varepsilon > 0$. Let the monitors communication graph be connected, let d be its diameter, and let the monitors execute the Asynchronous (diffusive) state estimation algorithm. Assume that there exists a duration $T \in \mathbb{R}$ such that, within each time interval of duration T, each monitor transmits its current estimates to its neighbors. Then, each monitor computes the estimate $\hat{x}(\varepsilon)$ of x within time dT.*

Proof: The proof follows from the following two facts. First, the intersection of subspaces is a commutative operation. Second, since each monitor performs a data transmission within any time interval of length T, it follows that, at time dT, the information related to one monitor has propagated through the network to all monitors. □

5.2.4 Detection of false data via distributed computation

In the previous sections we have shown how to compute an optimal state estimate via distributed computation. A rather straightforward application of the proposed state estimation technique is the detection of false data among the measurements. When the measurements are corrupted, the state to measurements relation becomes

$$z = Hx + v + w,$$

where w is the false data vector. As a consequence of Corollary 5.2.2, the vector w is detectable if it affects significantly the estimation residual, i.e., if $\lim_{\varepsilon \to 0} \|z - H\hat{x}(\varepsilon)\| > \Gamma$, where the threshold Γ depends upon the magnitude of the noise v. Notice that, because false data can be injected at any time by a malignant agent, the detection algorithm needs to be executed over time by the control centers. Let $z(t) = Hx(t) + v(t) + w(t)$ be the measurements vector at a given

Algorithm 3: *False data detection (i-th monitor)*

```
Input      : Matrices $H_i$, $\varepsilon B_i$, $\Gamma$;
while True do
    collect measurements $z_i(t)$;
    estimate network state $\hat{x}(t)$ via Algorithm 1 or 2;
    if $\|z_i(t) - H_i\hat{x}(t)\|_\infty > \Gamma$ then
        return false data detected;
```

time instant t. Based on these considerations, our distributed detection procedure is in Algorithm 3, where the matrices H_i and εB_i are as defined in equation (5.8), and Γ is a predefined threshold.

In Algorithm 3, the value of the threshold Γ determines the false alarm and the missed detection rates. Clearly, if $\Gamma \geq \|(I - HW)\|\|v(t)\|$ at all times t, and ε is sufficiently small, then no false alarm is triggered, at the expenses of the missed detection rate. By decreasing the value of Γ the sensitivity to failures increases together with the false alarm rate. Notice that, if the magnitude of the noise signals is upper bounded by γ, then a reasonable choice of the threshold is $\Gamma = \gamma\|(I - HW)\|_\infty$, where the use of the infinity norm in Algorithm 3 is also convenient for the implementation. Indeed, since the condition $\|z(t) - H\hat{x}(t)\|_\infty > \Gamma$ is equivalent to $\|z_i(t) - H_i\hat{x}(t)\|_\infty > \Gamma$ for some monitor i, the presence of false data can be independently checked by each monitor without further computation.

Notice that an eventual alarm message needs to be propagated to all control centers.

Remark 6 (Statistical detection) *A different strategy for the detection of false data relies on statistical techniques, e.g., see [1]. In the interest of brevity, we do not consider these methods, and we only remark that, once the estimation residual has been computed by each monitor, the implementation of a (distributed) statistical procedure, such as, for instance, the (distributed) χ^2-Test, is a straightforward task.*

5.3 A Finite-memory Estimation Technique

The procedure described in Algorithm 1 allows each agent to compute an optimal estimate of the whole network state in finite time. In this section, we allow each agent to handle only local and of small dimension vectors, and we develop a procedure to recover an estimate of only a certain subnetwork. We envision that the knowledge of only a subnetwork may be sufficient to implement distributed estimation and control strategies.

We start by introducing the necessary notation. Let the measurements matrix H be partitioned into m^2 blocks, being m the number of monitors in the network, as

$$H = \begin{bmatrix} H_{11} & \cdots & H_{1m} \\ \vdots & & \vdots \\ H_{m1} & \cdots & H_{mm} \end{bmatrix}, \tag{5.9}$$

where $H_{ij} \in \mathbb{R}^{m_i \times n_i}$ for all $i, j \in \{1, \dots, m\}$. The above partitioning reflects a division of the whole network into competence regions: we let each monitor be responsible for the correct functionality of the subnetwork defined by its blocks. Additionally, we assume that the union of the different regions covers the whole network, and that different competence regions may overlap. Observe that, in most of the practical situations, the matrix H has a sparse structure, so that many blocks H_{ij} have only zero entries. We associate an undirected graph G_h with the matrix H, in a way that G_h reflects the interconnection structure of the blocks H_{ij}. To be more precise, we let $G_h = (\mathcal{V}_h, \mathcal{E}_h)$, where $\mathcal{V}_h = \{1, \dots, m\}$ denotes the set of monitors, and where, denoting by (i, j) the undirected edge from j to i, it holds $(i, j) \in \mathcal{E}_h$ if and only if $\|H_{ij}\| \neq 0$ or $\|H_{ji}\| \neq 0$. Noticed that the structure of the graph G_h, besides reflecting the sparsity pattern of the measurement matrix, describes also the monitors interconnections. By using the same partitioning as in (5.9), the Moore-Penrose pseudoinverse of H can be written as

$$H^{\dagger} = \tilde{H} = \begin{bmatrix} \tilde{H}_{11} & \cdots & \tilde{H}_{1m} \\ \vdots & & \vdots \\ \tilde{H}_{m1} & \cdots & \tilde{H}_{mm} \end{bmatrix}, \tag{5.10}$$

where $\tilde{H}_{ij} \in \mathbb{R}^{n_i \times m_i}$. Assume that H has full row rank,[8] and observe that $H^{\dagger} = H^{\mathsf{T}}(HH^{\mathsf{T}})^{-1}$. Consider the equation $z = Hx$, and let $H^{\dagger}z = \hat{x} = \begin{bmatrix}\hat{x}_1^{\mathsf{T}} \dots \hat{x}_m^{\mathsf{T}}\end{bmatrix}^{\mathsf{T}}$, where, for all $i \in \{1, \dots, m\}$, $\hat{x}_i \in \mathbb{R}^{n_i}$. We employ Algorithm 2 for the computation of the vector $\hat{x}$, and we let

$$\hat{x}^{(i,h)} = \begin{bmatrix} \hat{x}_1^{(i,h)} \\ \vdots \\ \hat{x}_m^{(i,h)} \end{bmatrix}$$

be the estimate vector of the i-th monitor after h iterations of Algorithm 2, i.e., after h executions of the *while* loop in Algorithm 2. In what follows, we will show that, for a sufficiently sparse matrix H, the error $\|\hat{x}_i - \hat{x}_i^{(i,h)}\|$ has an exponential decay when h increases, so that it becomes negligible before the termination of Algorithm 2, i.e., when $h < \operatorname{diam}(G_h)$. The main result of this section is next stated.

Theorem 5.3.1 (Local estimation) *Let the full-row rank matrix H be partitioned as in (5.9). Let $\big[a, b\big]$, with $a < b$, be the smallest interval containing the*

[8] The case of a full-column rank matrix is treated analogously.

spectrum of HH^{T}. Then, for $i \in \{1, \dots, m\}$ and $h \in \mathbb{N}$, there exist $C \in \mathbb{R}_{>0}$ and $q \in (0, 1)$ such that

$$\|\hat{x}_i - \hat{x}_i^{(i,h)}\| \leq Cq^{\frac{h}{2}+1}.$$

For the readers convenience, before proving the above result, we recall the following definitions and results. Given an invertible matrix M of dimension n, let us define the *support sets*

$$S_h(M) = \bigcup_{k=0}^{h} \{(i, j) : M^k(i, j) \neq 0\},$$

being $M^k(i, j)$ the (i, j)-th entry of M^k, and the *decay sets*

$$D_h(M) = (\{1, \dots, n\} \times \{1, \dots, n\}) \setminus S_h(M).$$

Theorem 5.3.2 (Decay rate [27]) *Let M be of full row rank, and let $\big[a, b\big]$, with $a < b$, be the smallest interval containing the spectrum of M. There exist $C \in \mathbb{R}_{>0}$ and $q \in (0, 1)$ such that*

$$\sup\{|M^{\dagger}(i, j)| : (i, j) \in D_h(MM^{\mathsf{T}})\} \leq Cq^{h+1}.$$

For a graph G_h and two nodes i and j, let $\mathrm{dist}(i, j)$ denote the smallest number of edges in a path from j to i in G_h. The following result will be used to prove Theorem 5.3.1. Recall that, for a matrix M, we have $\|M\|_{\max} = \max\{|M(i, j)|\}$.

Lemma 5.3.3 (Decay sets and local neighborhood) *Let the matrix H be partitioned as in* (5.9)*, and let G_h be the graph associated with H. For $i, j \in \{1, \dots, m\}$, if* $\mathrm{dist}(i,j) = h$*, then*

$$\|H_{ij}^{\dagger}\|_{\max} \leq C q^{\frac{h}{2}+1}.$$

Proof: A proof of this result follows from simple inspection and it is omitted here. □

Lemma 5.3.3 establishes a relationship between the decay sets of an invertible matrix and the distance among the vertices of a graph associated with the same matrix. By using this result, we are now ready to prove Theorem 5.3.1.

Proof of Theorem 5.3.1: Notice that, after h iterations of Algorithm 2, the i-th monitor has received data from the monitors within distance h from i, i.e., from the monitors T such that, for each $j \in T$, there exists a path of length up to h from j to i in the graph associated with H. Reorder the rows of H such that the i-th block come first and the T-th blocks second. Let $H = \begin{bmatrix} H_1^{\mathsf{T}} \; H_2^{\mathsf{T}} \; H_3^{\mathsf{T}} \end{bmatrix}^{\mathsf{T}}$ be the resulting matrix. Accordingly, let $z = \begin{bmatrix} z_1^{\mathsf{T}} \; z_2^{\mathsf{T}} \; z_3^{\mathsf{T}} \end{bmatrix}^{\mathsf{T}}$, and let $x = \begin{bmatrix} x_1^{\mathsf{T}} \; x_2^{\mathsf{T}} \; x_3^{\mathsf{T}} \end{bmatrix}^{\mathsf{T}}$, where $z = Hx$.

Because H has full row rank, we have

$$\begin{bmatrix} H_1 \\ H_2 \\ H_3 \end{bmatrix} \begin{bmatrix} P_{11} & P_{12} & P_{13} \\ P_{21} & P_{22} & P_{23} \\ P_{31} & P_{32} & P_{33} \end{bmatrix} = \begin{bmatrix} I_1 & 0 & 0 \\ 0 & I_2 & 0 \\ 0 & 0 & I_3 \end{bmatrix},$$

where I_1, I_2, and I_3 are identity matrices of appropriate dimension, and

$$H^\dagger = \begin{bmatrix} P_{11} & P_{12} & P_{13} \\ P_{21} & P_{22} & P_{23} \\ P_{31} & P_{32} & P_{33} \end{bmatrix}.$$

For a matrix M, let $\text{col}(M)$ denote the number of columns of M. Let $T_1 = \{1, \dots, \text{col}(P_{11})\}$, $T_2 = \{1 + \text{col}(P_{11}), \dots, \text{col}(\begin{bmatrix} P_{11} \, P_{12} \end{bmatrix})\}$, and

$$T_3 = \{1 + \text{col}(\begin{bmatrix} P_{11} \, P_{12} \end{bmatrix}), \dots, \text{col}(\begin{bmatrix} P_{11} \, P_{12} \, P_{13} \end{bmatrix})\}.$$

Let T_1, T_2, and T_3, be, respectively, the indices of the columns of P_{11}, P_{12}, and P_{13}. Notice that, by construction, if $i \in T_1$ and $j \in T_3$, then $\text{dist}(i, j) > h$. Then, by virtue of Lemma 5.3.3 and Theorem 5.3.2, the magnitude of each entry of P_{13} is bounded by $\bar{C}\bar{q}^{\lfloor \frac{h}{2} \rfloor + 1}$, for $\bar{C}, \bar{q} \in \mathbb{R}$.

Because H has full row rank, from Theorem 5.2.1 we have that

$$\hat{x} = H^\dagger z = \hat{\bar{x}} + K_1 (H_3 K_1)^\dagger (z_3 - H_3 \hat{x}^1), \tag{5.11}$$

where

$$\hat{\bar{x}} = \begin{bmatrix} H_1^\mathsf{T} \; H_2^\mathsf{T} \end{bmatrix}^{\mathsf{T}\dagger} \begin{bmatrix} z_1^\mathsf{T} \; z_2^\mathsf{T} \end{bmatrix}^\mathsf{T} \text{ and } K_1 = \text{Basis}(\text{Ker}(\begin{bmatrix} H_1^\mathsf{T} \; H_2^\mathsf{T} \end{bmatrix}^\mathsf{T})).$$

With the same partitioning as before, let $\hat{x} = \left[\hat{x}_1^\mathsf{T}\ \hat{x}_2^\mathsf{T}\ \hat{x}_3^\mathsf{T}\right]^\mathsf{T}$. In order to prove the theorem, we need to show that there exists $C \in \mathbb{R}_{>0}$ and $q \in (0,1)$ such that

$$\|\hat{x}_1 - \hat{\hat{x}}_1\| \leq Cq^{\lfloor \frac{h}{2} \rfloor + 1}.$$

Notice that, for (5.11) to hold, the matrix K_1 can be any basis of $\mathrm{Ker}([H_1^\mathsf{T}\ H_2^\mathsf{T}]^\mathsf{T})$. Hence, let $K_1 = \left[P_{13}^\mathsf{T}\ P_{23}^\mathsf{T}\ P_{33}^\mathsf{T}\right]$. Because every entry of P_{13} decays exponentially, the theorem follows. □

In Section 5.4.2 we provide an example to clarify the exponential decay described in Theorem 5.3.1.

5.4 Illustrative Examples

The effectiveness of the methods developed in the previous sections is now shown through some examples.

5.4.1 Estimation and detection for the IEEE 118 system

The IEEE 118 bus system represents a portion of the American Electric Power System as of December, 1962. This test case system, whose diagram is reported in Fig. 5.1, is composed of 118 buses, 186 branches, 54 generators, and 99 loads. The voltage angles θ_{bus} and the power injections P_{bus} at the network buses are

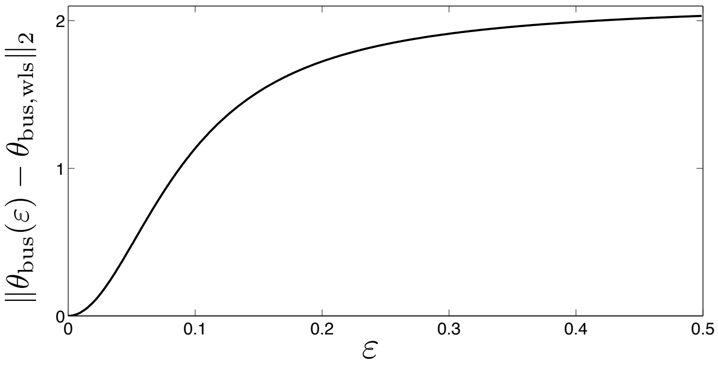


Figure 5.2: In this figure the normalized euclidean norm of the error vector $\theta_{\mathrm{bus}}(\varepsilon) - \theta_{\mathrm{bus,wls}}$ is plotted as a function of the parameter ε. Here, $\theta_{\mathrm{bus}}(\varepsilon)$ is the estimation vector computed according to Theorem 5.2.2, and $\theta_{\mathrm{bus,wls}}$ is the minimum variance estimate of θ_{bus}. As ε decreases, the vector $\theta_{\mathrm{bus}}(\varepsilon)$ converges to the minimum variance estimate $\theta_{\mathrm{bus,wls}}$.

assumed to be related through the linear relation

$$P_{\mathrm{bus}} = H_{\mathrm{bus}}\theta_{\mathrm{bus}},$$

where the matrix H_{bus} depends upon the network interconnection structure and the network admittance matrix. For the network in Fig. 5.1, let $z = P_{\mathrm{bus}} - v$ be the measurements vector, where $\mathbb{E}\left[v\right] = 0$ and $\mathbb{E}\left[vv^{\mathsf{T}}\right] = \sigma^2 I$, $\sigma \in \mathbb{R}$. Then, following the notation in Theorem 5.2.2, the minimum variance estimate of θ_{bus} can be recovered as

$$\lim_{\varepsilon \to 0^+} \left[H_{\mathrm{bus}}\ \varepsilon\sigma I\right]^{\dagger} z.$$

In Fig. 5.2 we show that, as ε decreases, the estimation vector computed according to Theorem 5.2.2 converges to the minimum variance estimate of θ_{bus}.

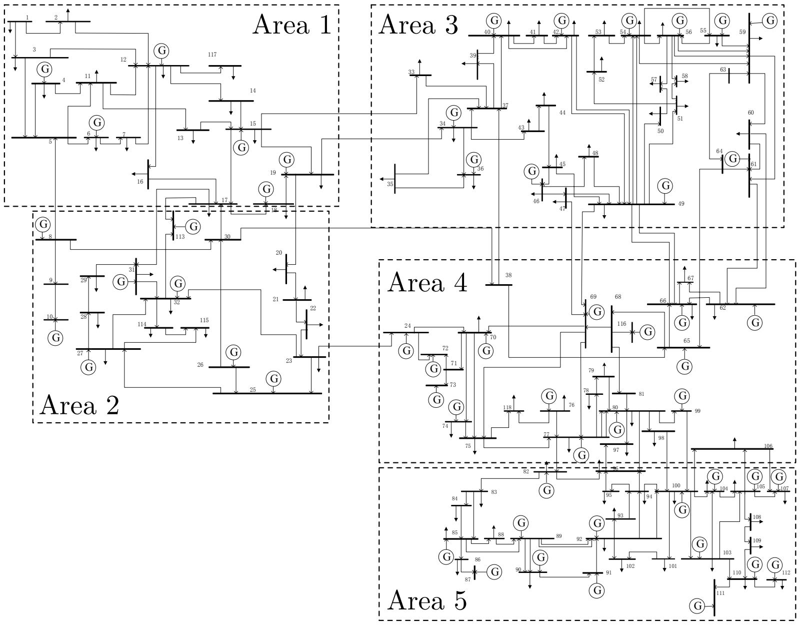


Figure 5.3: In this figure the IEEE 118 bus system has been divided into 5 areas. Each area is monitored and operated by a control center. The control centers cooperate to estimate the state and to assess the functionality of the whole network.

In order to demonstrate the advantage of our decentralized estimation algorithm, we assume the presence of 5 control centers in the network of Fig. 5.1, each one responsible for a subpart of the entire network. The situation is depicted in Fig. 5.3. Assume that each control center measures the real power injected at the buses in its area, and let $z_i = P_{\text{bus,i}} - v_i$, with $\mathbb{E}\left[v_i\right] = 0$ and $\mathbb{E}\left[v_i v_i^{\mathsf{T}}\right] = \sigma_i^2 I$, be the measurements vector of the i-th area. Finally, assume that the i-th control center knows the matrix $H_{\text{bus,i}}$ such that $z_i = H_{\text{bus,i}}\theta_{\text{bus}} + v_i$. Then, as discussed in Section 5.2.2, the control centers can compute an optimal estimate of θ_{bus} by

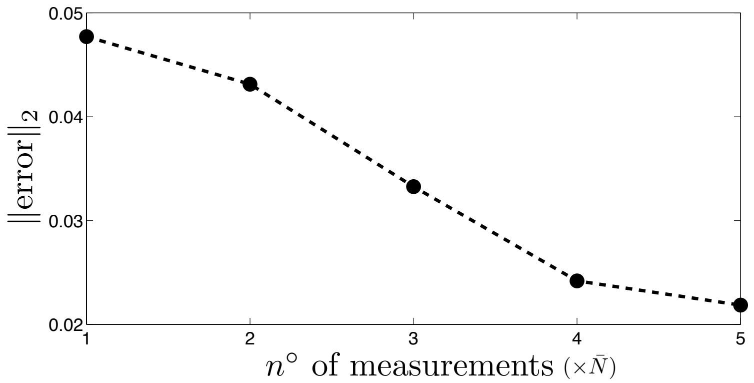


Figure 5.4: For a fixed value of ε, Fig. 5.4 shows the average (over 100 tests) of the norm of the error (with respect to the network state) of the estimate obtained by means of Algorithm 1. The estimation error decreases with the number of measurements. Because of the presence of several control centers, our algorithm processes more measurements (up to $5\bar{N}$) while maintaining the same (or smaller) computational complexity of a centralized estimation with $\bar{N}$ measurements.

means of Algorithm 1 or 2. Let n_i be the number of measurements of the i-th area, and let $N = \sum_{i=1}^{5} n_i$. Notice that, with respect to a centralized computation of the minimum variance estimate of the state vector, our estimation procedure obtains the same estimation accuracy while requiring a smaller computational burden and memory requirement. Indeed, the i-th monitor uses n_i measurements instead of N. Let $\bar{N}$ be the maximum number of measurements that, due to hardware or numerical contraints, a control center can efficiently handle for the state estimation problem. In Fig. 5.4, we increase the number of measurements taken by a control center, so that $n_i \leq \bar{N}$, and we show how the accuracy of the state estimate increases with respect to a single control center with $\bar{N}$ measurements.

To conclude this section, we consider a security application, in which the control centers aim at detecting the presence of false data among the network measurements via distributed computation. For this example, we assume that each control center measures the real power injection as well the current magnitude at some of the buses of its area. By doing so, a sufficient redundancy in the measurements is obtained for the detection to be feasible [1]. Suppose that the measurements of the power injection at the first bus of the first area is corrupted by a malignant agent. To be more precise, let the measurements vector of the first area be $\bar{z}_i = z_i + e_1 w_i$, where e_1 is the first canonical vector, and w_i is a random variable. For the simulation we choose w_i to be uniformly distributed in the interval $[0, w_{\max}]$, where $w_{\max}$ corresponds approximately to the 10% of the nominal real injection value. In order to detect the presence of false data among the measurements, the control centers implement Algorithm 3, where, being H the measurements matrix, and σ, Σ the noise standard deviation and covariance matrix, the threshold value Γ is chosen as $2\sigma \| I - H(H^{\mathsf{T}}\Sigma^{-1}H)^{-1}H^{\mathsf{T}}\Sigma^{-1}\|_{\infty}$.[9] The residual functions $\|z_i - H\hat{x}\|_{\infty}$ are reported in Fig. 5.5. Observe that, since the first residual is greater than the threshold Γ, the control centers successfully detect the false data. Regarding the identification of the corrupted measurements, we remark that a regional identification may be possible by simply analyzing the

[9] For a Gaussian distribution with mean μ and variance σ^2, about 95% of the realizations are contained in $[\mu - 2\sigma, \mu + 2\sigma]$.

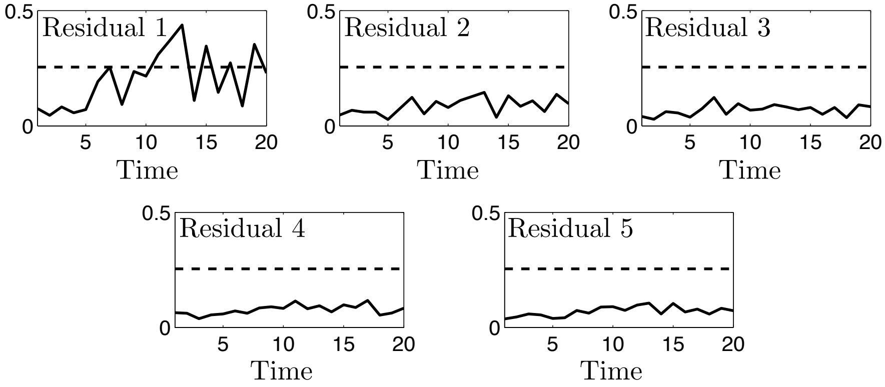


Figure 5.5: Fig. 5.5 shows the residual functions computed by the 5 control centers. Since the first residual is greater than the threshold value, the presence of false data is correctly detected by the first control center. A form of regional identification is possible by simple identifying the residuals above the security threshold.

residual functions. In this example, for instance, since the residuals 2 to 5 are below the threshold value, the corrupted data is likely to be among the measurements of the first area. This important aspect is left as the subject of future research.

5.4.2 Scalability property of finite-memory estimation

Consider an electrical network with $(ab)^2$ buses, where $a, b \in \mathbb{N}$. Let the buses interconnection structure be a two dimensional lattice, and let G be the graph whose vertices are the $(ab)^2$ buses, and whose edges are the network branches. Let G be partitioned into b^2 identical blocks containing a^2 vertices each, and assume the presence of b^2 control centers, each one responsible for a different network

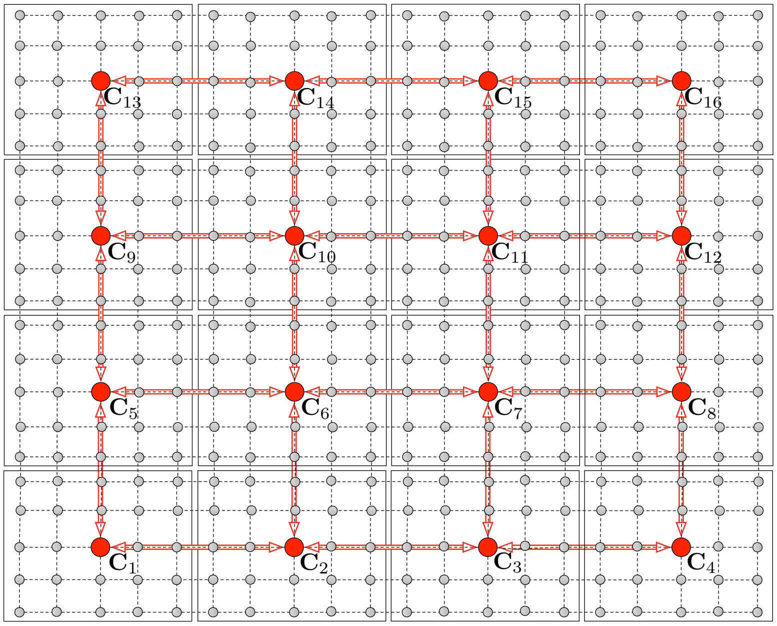


Figure 5.6: In Fig. 5.6, a two dimensional power grid with 400 buses. The network is operated by 16 control centers, each one responsible for a different subnetwork. Control centers cooperate through the red communication graph.

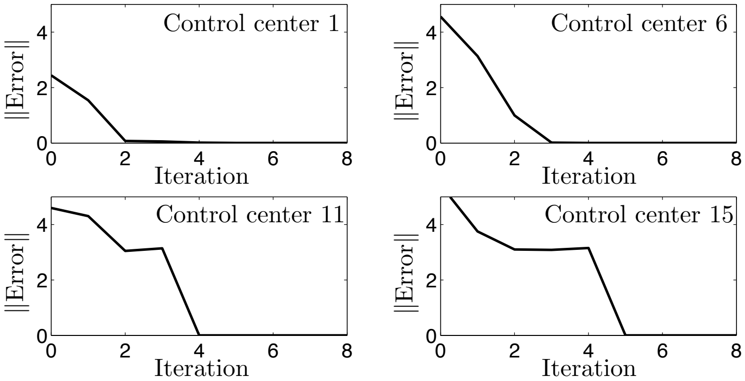


Figure 5.7: Fig. 5.7 shows the norm of the estimation error of the local subnetwork as a function of the number of iterations of Algorithm 2. The considered monitors are C_1 ,C_6, C_{11}, and C_{15}. As predicted by Theorem 5.3.1, the local estimation error becomes negligible before the termination of the algorithm.

part. We assume the control centers to be interconnected through an undirected graph. In particular, being V_i the set of buses assigned to the control center C_i, we let the control centers C_i and C_j be connected if there exists a network branch linking a bus in V_i to a bus in V_j. An example with $b = 4$ and $a = 5$ is in Fig. 5.6. In order to show the effectiveness of our approximation procedure, suppose that each control center C_i aims at estimating the vector of the voltage angles at the buses in its region. We assume also that the control centers cooperate, and that each of them receives the measurements of the real power injected at only the buses in its region. Algorithm 2 is implemented by the control centers to solve the estimation problem. In Fig. 5.7 we report the estimation error during the iterations of the algorithm. Notice that, as predicted by Theorem 5.3.1, each leader has a good estimate of the state of its region before the termination of the algorithm.

5.5 Proofs of Main Results

5.5.1 Proof of Theorem 5.2.1

Proof: Let $H^i = \begin{bmatrix} H_1^\mathsf{T} & \cdots & H_i^\mathsf{T} \end{bmatrix}^\mathsf{T}$, $z^i = \begin{bmatrix} z_1^\mathsf{T} & \cdots & z_i^\mathsf{T} \end{bmatrix}^\mathsf{T}$. We show by induction that $z^i = H^i \hat{x}_i$, $\quad K_i = \text{Basis}(\text{Ker}(H^i))$, and $\hat{x}_i \perp \text{Ker}(H^i)$. Note that the statements are trivially verified for $i = 1$. Suppose that they are verified up to i, then we need to show that $K_{i+1} = \text{Basis}(\text{Ker}(H^{i+1}))$, $\hat{x}_{i+1} \perp \text{Ker}(H^{i+1})$, and $z^{i+1} = H^{i+1}\hat{x}_{i+1}$.

We start by proving that $K_{i+1} = \text{Basis}(\text{Ker}(H^{i+1}))$. Observe that $\text{Ker}(K_i) = 0$ for all i, and that

$$\text{Ker}(H_{i+1}K_i) = \{v : K_i v \in \text{Ker}(H_{i+1})\}. \tag{5.12}$$

Hence,

$$\begin{aligned}\text{Im}(K_{i+1}) &= \text{Im}(K_i \,\text{Ker}(H_{i+1}K_i)) \\ &= \text{Im}(K_i) \cap \text{Ker}(H_{i+1}) \\ &= \text{Ker}(H^i) \cap \text{Ker}(H_{i+1}) = \text{Ker}(H^{i+1}).\end{aligned}$$

We now show that $x_{i+1} \perp \text{Ker}(H^{i+1})$, which is equivalent to

$$(\hat{x}_i + K_i(H_{i+1}K_i)^\dagger(z_{i+1} - H_{i+1}\hat{x}_i)) \in \text{Ker}(H^{i+1})^\perp.$$

Note that

$$\text{Ker}(H^{i+1}) \subseteq \text{Ker}(H^i) \;\;\Leftrightarrow\;\; \text{Ker}(H^{i+1})^\perp \supseteq \text{Ker}(H^i)^\perp.$$

By the induction hypothesis we have $\hat{x}_i \in \text{Ker}(H^i)^\perp$, and hence $\hat{x}_i \in \text{Ker}(H^{i+1})^\perp$. Therefore, we need to show that

$$K_i(H_{i+1}K_i)^\dagger(z_{i+1} - H_{i+1}\hat{x}_i) \in \text{Ker}(H^{i+1})^\perp.$$

Let $w = (H_{i+1}K_i)^\dagger(z_{i+1} - H_{i+1}\hat{x}_i)$, and notice that $w \in \text{Ker}(H_{i+1}K_i)^\perp$ due to the properties of the pseudoinverse operation. Suppose that $K_i w \notin \text{Ker}(H_{i+1})^\perp$. Since $\text{Ker}(K_i) = \{0\}$, the vector w can be written as $w = w_1 + w_2$, where $K_i w_1 \in \text{Ker}(H_{i+1})^\perp$ and $K_i w_2 = K_i w - K_i w_1 \neq 0$, $K_i w_2 \in \text{Ker}(H_{i+1})$. Then, it holds $H_{i+1}K_i w_2 = 0$, and hence $w_2 \in \text{Ker}(H_{i+1}K_i)$, which contradicts the hypothesis $w \in \text{Ker}(H_{i+1}K_i)^\perp$. Finally $K_i w \in \text{Ker}(H_{i+1})^\perp \subseteq \text{Ker}(H^{i+1})^\perp$.

We now show that $z^{i+1} = H^{i+1}\hat{x}_{i+1}$. Because of the consistency of the system of linear equations, and because $z^i = H^i\hat{x}_i$ by the induction hypothesis, there exists a vector $v_i \in \text{Ker}(H^i) = \text{Im}(K_i)$ such that $z^{i+1} = H^{i+1}(\hat{x}_i + v_i)$, and hence that $z_{i+1} = H_{i+1}(\hat{x}_i + v_i)$. We conclude that $(z_{i+1} - H_{i+1}\hat{x}_i) \in \text{Im}(H_{i+1}K_i)$, and finally that $z^{i+1} = H^{i+1}\hat{x}_{i+1}$. □

5.5.2 Proof of Theorem 5.2.2

Before proceeding with the proof of the above theorem, we recall the following fact in linear algebra.

Lemma 5.5.1 *Let $H \in \mathbb{R}^{n\times m}$. Then* $\text{Ker}((H^\dagger)^\mathsf{T}) = \text{Ker}(H)$.

Proof: We first show that $\text{Ker}((H^\dagger)^\mathsf{T}) \subseteq \text{Ker}(H)$. Recall from [9] that $H = HH^\mathsf{T}(H^\dagger)^\mathsf{T}$. Let x be such that $(H^\dagger)^\mathsf{T}x = 0$, then $Hx = HH^\mathsf{T}(H^\dagger)^\mathsf{T}x = 0$, so that $\text{Ker}((H^\dagger)^\mathsf{T}) \subseteq \text{Ker}(H)$. We now show that $\text{Ker}(H) \subseteq \text{Ker}((H^\dagger)^\mathsf{T})$. Recall that $(H^\dagger)^\mathsf{T} = (H^\mathsf{T})^\dagger = (HH^\mathsf{T})^\dagger H$. Let x be such that $Hx = 0$, then $(H^\dagger)^\mathsf{T} x = (HH^\mathsf{T})^\dagger Hx = 0$, so that $\mathrm{Ker}(H) \subseteq \mathrm{Ker}((H^\dagger)^\mathsf{T})$, which concludes the proof. □

We are now ready to prove Theorem 5.2.2.

Proof: The first property follows directly from [9] (cfr. page 427). To show the second property, observe that $C^\dagger = \frac{1}{\varepsilon}((I - HH^\dagger)B)^\dagger$, so that

$$\lim_{\varepsilon \to 0^+} \varepsilon D = 0.$$

For the theorem to hold, we need to verify that

$$H^\dagger - H^\dagger B((I - HH^\dagger)B)^\dagger = (H^\mathsf{T}\Sigma^{-1}H)^{-1}H^\mathsf{T}\Sigma^{-1},$$

or, equivalently, that

$$\left(H^\dagger - H^\dagger B((I - HH^\dagger)B)^\dagger\right) HH^\dagger = (H^\mathsf{T}\Sigma^{-1}H)^{-1}H^\mathsf{T}\Sigma^{-1}HH^\dagger, \tag{5.13}$$

and

$$\left(H^\dagger - H^\dagger B((I - HH^\dagger)B)^\dagger\right) (I - HH^\dagger) = (H^\mathsf{T}\Sigma^{-1}H)^{-1}H^\mathsf{T}\Sigma^{-1}(I - HH^\dagger). \tag{5.14}$$

Consider equation (5.13). After simple manipulation, we have

$$H^\dagger - H^\dagger B((I - HH^\dagger)B)^\dagger HH^\dagger = H^\dagger,$$

so that we need to show only that

$$H^\dagger B((I - HH^\dagger)B)^\dagger HH^\dagger = 0.$$

Recall that for a matrix W it holds $W^\dagger = (W^\mathsf{T} W)^\dagger W^\mathsf{T}$. Then the term $((I - HH^\dagger)B)^\dagger HH^\dagger$ equals

$$\left(((I-HH^\dagger)B)^\mathsf{T}((I-HH^\dagger)B)\right)^\dagger B^\mathsf{T}(I-HH^\dagger)HH^\dagger = 0,$$

because $(I - HH^\dagger)HH^\dagger = 0$. We conclude that equation (5.13) holds. Consider now equation (5.14). Observe that $HH^\dagger(I - HH^\dagger) = 0$. Because B has full row rank, and $\Sigma = BB^\mathsf{T}$, simple manipulation yields

$$-H^\mathsf{T}(BB^\mathsf{T})^{-1}HH^\dagger B\left[(I - HH^\dagger)B\right]^\dagger (I - HH^\dagger)B = H^\mathsf{T}(BB^\mathsf{T})^{-1}(I - HH^\dagger)B,$$

and hence

$$H^\mathsf{T}(BB^\mathsf{T})^{-1}\left\{I + HH^\dagger B\left[(I-HH^\dagger)B\right]^\dagger\right\}(I-HH^\dagger)B = 0.$$

Since $HH^\dagger = I - (I - HH^\dagger)$, we obtain

$$H^\mathsf{T}(BB^\mathsf{T})^{-1}B\left[(I - HH^\dagger)B\right]^\dagger (I - HH^\dagger)B = 0.$$

A sufficient condition for the above equation to be true is

$$\left(\left[(I - HH^\dagger)B\right]^\dagger\right)^\mathsf{T} B^\mathsf{T}(BB^\mathsf{T})^{-1}H = 0.$$

From Lemma 5.5.1 we have.

$$\mathrm{Ker}\left(\left(\left[(I - AA^\dagger)B\right]^\dagger\right)^\mathsf{T}\right) = \mathrm{Ker}((I - AA^\dagger)B).$$

Since

$$(I - HH^{\dagger})BB^{\mathsf{T}}(BB^{\mathsf{T}})^{-1}H = (I - HH^{\dagger})H = 0,$$

we have that

$$H^{\mathsf{T}}(BB^{\mathsf{T}})^{-1}B\left[(I - HH^{\dagger})B\right]^{\dagger}(I - HH^{\dagger})B = 0,$$

and that equation (5.14) holds. This concludes the proof. □

Chapter 6

Dynamic Monitors for Attack Detection and Identification

In this chapter we design centralized and distributed monitors for attack detection and identification. Following the discussion in Chapter 4, we do not design active monitors. In particular, the algorithms presented in this chapter constitute dynamic monitors. We start with the design of dynamic detection monitors.

6.1 Monitors for Attack Detection

6.1.1 A centralized attack detection monitor

In the following we present a centralized attack detection filter based on a modified Luenberger observer.

Theorem 6.1.1 ***(Centralized attack detection filter)*** *Consider the descriptor system* (4.1) *and assume that the attack set K is detectable, and that the* *network initial state $x(0)$ is known. Consider the* centralized attack detection filter

$$
\begin{aligned}
E\dot{w}(t) &= (A + GC)w(t) - Gy(t), \\
r(t) &= Cw(t) - y(t),
\end{aligned}
\tag{6.1}
$$

where $w(0) = x(0)$ and the output injection $G \in \mathbb{R}^{n\times p}$ is such that the pair $(E, A+GC)$ is regular and Hurwitz. Then $r(t) = 0$ at all times $t \in \mathbb{R}_{\geq 0}$ if and only if $u_K(t) = 0$ at all times $t \in \mathbb{R}_{\geq 0}$. Moreover, in the absence of attacks, the filter error $w - x$ is exponentially stable.

Proof: Consider the error $e = w - x$ between the dynamic states of the filter (6.1) and the descriptor system (4.1). The error dynamics with output r are given by

$$
\begin{aligned}
E\dot{e}(t) &= (A + GC)e(t) - (B_K + GD_K)u_K(t), \\
r(t) &= Ce(t) - D_K u_K(t),
\end{aligned}
\tag{6.2}
$$

where $e(0) = 0$. To prove the theorem we show that the error system (6.2) has no invariant zeros, that is, $r(t) = 0$ for all $t \in \mathbb{R}_{\geq 0}$ if and only if $u_K(t) = 0$ for all $t \in \mathbb{R}_{\geq 0}$. Since the initial condition $x(0)$ and the input u_K are assumed to be consistent (A2) and non-impulsive (A3), the error system (6.2) has no invariant zeros if and only if [35, Proposition 3.4] there exists no triple $(s, \bar{w}, g_K) \in \mathbb{C}\times\mathbb{R}^n\times\mathbb{R}^p$ satisfying

$$
\begin{bmatrix} sE - (A + GC) & B_K + GD_K \\ C & -D_K \end{bmatrix} \begin{bmatrix} \bar{w} \\ g_K \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}. \tag{6.3}
$$

The second equation of (6.3) yields $C\bar{w} = D_K g_K$. Thus, by substituting $C\bar{w}$ by $D_K g_K$ in the first equation of (6.3), the set of equations (6.3) can be equivalently written as

$$\begin{bmatrix} sE - A & B_K \\ C & -D_K \end{bmatrix} \begin{bmatrix} \bar{w} \\ g_K \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}. \tag{6.4}$$

Finally, note that a solution $(s, -\bar{w}, g_K)$ to above set of equations would yield an invariant zero, zero state, and zero input for the descriptor system (4.1). By the detectability assumption,[1] the descriptor model (4.1) has no zero dynamics and the matrix pencil in (6.4) necessarily has full rank. It follows that the triple (E, A, C) is observable, so that G can be chosen to make the pair $(E, A + GC)$ Hurwitz [24, Theorem 4.1.1], and the error system (6.2) is stable and with no zero dynamics. □

Remark 7 ***(Detection and identification filters for unknown initial condition and noisy dynamics)*** *If the network initial state is not available, then, since $(E, A + GC)$ is Hurwitz, an arbitrary initial state $w(0) \in \mathbb{R}^n$ can be chosen. Consequently, the filter converges asymptotically, and some attacks may remain undetected or unidentified. For instance, if the eigenvalues of the detection filter matrix have real part smaller than $c < 0$, with $c \in \mathbb{R}$, then, in the absence of*

[1] Due to linearity of the descriptor system (4.1), the detectability assumption reads as "the attack $(Bu, Du,)$ is detectable if there exist no initial condition $x_0 \in \mathbb{R}^n$, such that $y(x_0, u, t) = 0$ for all $t \in \mathbb{R}_{\geq 0}$."

attacks, the residual r *exponentially converges to zero with rate less than* c*. Hence, only inputs* u *that vanish faster or equal than* e^{-ct} *may remain undetected by the filter* (6.1)*. Alternatively, the detection filter can be modified so as to converge in a predefined finite time, see [63, 85]. In this case, every attack signal is detectable after a finite transient.*

If the dynamics and the measurements of (4.1) *are affected by modeling uncertainties and noise with known statistics, then the output injection matrix* G *in* (6.1) *should be chosen as to optimize the sensitivity of the residual* r *to attacks versus the effect of noise. Standard robust filtering or model matching techniques can be adopted for this task [100]. Statistical hypothesis techniques can subsequently be used to analyze the residual* r *[7]. Finally, attacks aligned with the noise statistics turn out to be undetectable.* □

Observe that the design of the filter (6.1) is independent of the particular attack signature (B_K, D_K) and its performance is optimal in the sense that any detectable attack set K can be detected. We remark that for index-one descriptor systems such as power system models, the filter (6.1) can analogously be designed for the corresponding Kron-reduced model; see also [80]. In this case, the resulting attack detection filter is low-dimensional and non-singular but also non-sparse, see [80]. In comparison, the presented filter (6.1), although inherently centralized, features the *sparse* matrices (E, A, C). This sparsity will be key to develop a distributed attack detection filter.

6.1.2 A decentralized attack detection monitor

Let $G_\mathrm{t} = (\mathcal{V}, \mathcal{E})$ be the directed graph associated with the pair (E, A), where the vertex set $\mathcal{V} = \{1, \dots, n\}$ corresponds to the system state, and the set of directed edges $\mathcal{E} = \{(x_j, x_i) : e_{ij} \neq 0 \text{ or } a_{ij} \neq 0\}$ is induced by the sparsity pattern of E and A; see also Section 4.3. Assume that $\mathcal{V}$ has been partitioned into N disjoint subsets as $\mathcal{V} = \mathcal{V}_1 \cup \dots \cup \mathcal{V}_N$, with $|\mathcal{V}_i| = n_i$, and let $G_\mathrm{t}^i = (\mathcal{V}_i, \mathcal{E}_i)$ be the i-th subgraph of G_r with vertices $\mathcal{V}_i$ and edges $\mathcal{E}_i = \mathcal{E} \cap (\mathcal{V}_i \times \mathcal{V}_i)$. According to this partition, and possibly after relabeling the states, the system matrix A in (4.1) can be written as

$$A = \begin{bmatrix} A_1 & \cdots & A_{1N} \\ \vdots & \vdots & \vdots \\ A_{N1} & \cdots & A_N \end{bmatrix} = A_D + A_C,$$

where $A_i \in \mathbb{R}^{n_i \times n_i}$, $A_{ij} \in \mathbb{R}^{n_i \times n_j}$, A_D is block-diagonal, and $A_C = A - A_D$. Notice that, if $A_D = \operatorname{blkdiag}(A_1, \dots, A_N)$, then A_D represents the isolated subsystems and A_C describes the interconnection structure among the subsystems. Additionally, if the original system is sparse, then several blocks in A_C vanish. We make the following assumptions:

(A4) the matrices E, C are block-diagonal, that is $E = \mathrm{blkdiag}(E_1, \dots, E_N)$, $C = \mathrm{blkdiag}(C_1, \dots, C_N)$, where $E_i \in \mathbb{R}^{n_i \times n_i}$ and $C_i \in \mathbb{R}^{p_i \times n_i}$,

(A5) each pair (E_i, A_i) is regular, and each triple (E_i, A_i, C_i) is observable.

Given the above structure and in the absence of attacks, the descriptor system (4.1) can be written as the interconnection of N subsystems of the form

$$\begin{aligned} E_i \dot{x}_i(t) &= A_i x_i(t) + \sum_{j \in \mathcal{N}_i^{\mathrm{in}}} A_{ij} x_j(t), \\ y_i(t) &= C_i x_i(t), \;\; i \in \{1, \dots, N\}, \end{aligned} \tag{6.5}$$

where x_i and y_i are the state and output of the i-th subsystem and $\mathcal{N}_i^{\mathrm{in}} = \{j \in \{1, \dots, N\} \setminus i \; : \; \|A_{ij}\| \neq 0\}$ are the in-neighbors of subsystem i. We also define the set of out-neighbors as $\mathcal{N}_i^{\mathrm{out}} = \{j \in \{1, \dots, N\} \setminus i \; : \; \|A_{ji}\| \neq 0\}$. We assume the presence of a *control center* in each subnetwork G_{t}^i with the following capabilities:

(A6) the i-th control center knows the matrices E_i, A_i, C_i, as well as the neighboring matrices A_{ij}, $j \in \mathcal{N}_i^{\mathrm{in}}$; and

(A7) the i-th control center can transmit an estimate of its state to the j-th control center if $j \in \mathcal{N}_i^{\mathrm{out}}$.

Before deriving a fully-distributed attack detection filter, we explore the question of *decentralized stabilization* of the error dynamics of the filter (6.1). For each subsystem (6.5), consider the local residual generator

$$\begin{aligned} E_i\dot{w}_i(t) &= (A_i + G_iC_i)w_i(t) + \sum_{j\in\mathcal{N}_i^{\text{in}}} A_{ij}x_j(t) - G_iy_i(t), \\ r_i(t) &= y_i(t) - C_iw_i(t), \quad i \in \{1, \dots, N\}, \end{aligned} \tag{6.6}$$

where w_i is the i-th estimate of x_i and $G_i \in \mathbb{R}^{n_i\times p_i}$. In order to derive a compact formulation, let $w = [w_1^\mathsf{T} \cdots w_N^\mathsf{T}]^\mathsf{T}$, $r = [r_1^\mathsf{T} \cdots r_N^\mathsf{T}]^\mathsf{T}$, and $G = \text{blkdiag}(G_1, \dots, G_N)$. Then, the overall filter dynamics (6.6) are

$$\begin{aligned} E\dot{w}(t) &= (A_D + GC)w(t) + A_Cw(t) - Gy(t)\,, \\ r(t) &= y(t) - Cw(t)\,. \end{aligned} \tag{6.7}$$

Due to the observability assumption (A5) an output injection matrix G_i can be chosen such that each pair $(E_i, A_i - G_iC_i)$ is Hurwitz [24, Theorem 4.1.1]. Notice that, if each pair $(E_i, A_i + G_iC_i)$ is regular and Hurwitz, then $(E, A_D + GC)$ is also regular and Hurwitz since the matrices E and $A_D + GC$ are block-diagonal. We are now ready to state a condition for the decentralized stabilization of the filter (6.7).

Lemma 6.1.2 ***(Decentralized stabilization of the attack detection filter)*** *Consider the descriptor system* (4.1)*, and assume that the attack set K is detectable and that the network initial state $x(0)$ is known. Consider the attack detection filter* (6.7)*, where $w(0) = x(0)$ and $G = \text{blkdiag}(G_1, \dots, G_N)$ is such* *that $(E, A_D + GC)$ is regular and Hurwitz. Assume that*

$$\rho\left((j\omega E - A_D - GC)^{-1} A_C\right) < 1 \text{ for all } \omega \in \mathbb{R}\,, \tag{6.8}$$

where $\rho(\cdot)$ denotes the spectral radius operator. Then $r(t) = 0$ at all times $t \in \mathbb{R}_{\geq 0}$ if and only if $u_K(t) = 0$ at all times $t \in \mathbb{R}_{\geq 0}$. Moreover, in the absence of attacks, the filter error $w - x$ is exponentially stable.

Proof: The error $e = w - x$ obeys the dynamics

$$\begin{aligned} E\dot{e}(t) &= (A_D + A_C + GC)e(t) - (B_K + GD_K)u_K(t), \\ r(t) &= Ce(t) - D_K u_K(t)\,. \end{aligned} \tag{6.9}$$

A reasoning analogous to that in the proof of Theorem 6.1.1 shows the absence of zero dynamics. Hence, for $r(t) = 0$ at all times $t \in \mathbb{R}_{\geq 0}$ if and only if $u_K(t) = 0$ at all times $t \in \mathbb{R}_{\geq 0}$.

To show stability of the error dynamics in the absence of attacks, we employ the small-gain approach to large-scale interconnected systems [116] and rewrite the error dynamics (6.9) as the closed-loop interconnection of the two subsystems

$$\begin{aligned} \Gamma_1 : &\quad E\dot{e}(t) = (A_D + GC)e(t) + v(t)\,, \\ \Gamma_2 : &\quad v(t) = A_C e(t)\,. \end{aligned}$$

Since both subsystems Γ_1 and Γ_2 are causal and internally Hurwitz stable, the overall error dynamics (6.9) are stable if the loop transfer function $\Gamma_1(j\omega) \cdot \Gamma_2$

satisfies the spectral radius condition $\rho(\Gamma_1(j\omega)\cdot\Gamma_2) < 1$ for all $\omega \in \mathbb{R}$ [100, Theorem 4.11]. The latter condition is equivalent to (6.8). □

Observe that, although control centers can compute the output injection matrix independently of each other, an implementation of the decentralized attack detection filter (6.7) requires control centers to continuously exchange their local estimation vectors. Thus, this scheme has high communication cost, and it may not be broadly applicable. A solution to this problem is presented in the next section.

6.1.3 A distributed attack detection monitor

In this subsection we exploit the classical waveform relaxation method to develop a fully distributed variation of the decentralized attack detection filter (6.7). We refer the reader to [23, 52] for a comprehensive discussion of waveform relaxation methods. The Gauss-Jacobi waveform relaxation method applied to the system (6.7) yields the *waveform relaxation iteration*

$$E\dot{w}^{(k)}(t) = A_D w^{(k)}(t) + A_C w^{(k-1)}(t) - Gy(t) , \tag{6.10}$$

where $k \in \mathbb{N}$ denotes the iteration index, $t \in [0, T]$ is the integration interval for some uniform time horizon $T > 0$, and $w^{(k)} : [0, T] \to \mathbb{R}^n$ is a trajectory with the initial condition $w^{(k)}(0) = w_0$ for each $k \in \mathbb{N}$. Notice that (6.10) is a descriptor system in the variable $w^{(k)}$ and the vector $A_C w^{(k-1)}$ is a known input, since the value of w at iteration $k-1$ is used. The iteration (6.10) is said to be (uniformly) *convergent* if

$$\lim_{k\to\infty} \max_{t\in[0,T]} \left\| w^{(k)}(t) - w(t) \right\|_\infty = 0\,,$$

where w is the solution of the non-iterative dynamics (6.7). In order to obtain a low-complexity distributed detection scheme, we use the waveform relaxation iteration (6.10) to iteratively approximate the decentralized filter (6.7).

We start by presenting a convergence condition for the iteration (6.7). Recall that a function $f : \mathbb{R}_{\geq 0} \to \mathbb{R}^p$ is said to be of *exponential order* β if there exists $\beta \in \mathbb{R}$ such that the exponentially scaled function $t \to f(t)e^{-\beta t}$ and all its derivatives exist and are bounded. An elegant analysis of the waveform relaxation iteration (6.10) can be carried out in the Laplace domain [5], where the operator mapping $w^{(k-1)}$ to $w^{(k)}$ is $(sE - A_D - GC)^{-1}A_C$. Similar to the regular Gauss-Jacobi iteration, convergence conditions of the waveform relaxation iteration (6.10) rely on the contractivity of the iteration operator.

Lemma 6.1.3 ***(Convergence of the waveform relaxation [5, Theorem 5.2])*** *Consider the waveform relaxation iteration* (6.10). *Let the pair* $(E, A_D + GC)$ *be regular, and the initial condition* w_0 *be consistent. Let* $y : [0,T] \to \mathbb{R}^p$ *be of exponential order* β. *Let* α *be the least upper bound on the real part of the* *spectrum of* (E, A)*, and define* $\sigma = \max\{\alpha, \beta\}$*. The waveform relaxation method* (6.10) *is convergent if*

$$\rho\left(((\sigma + j\omega)E - A_D - GC)^{-1}A_C\right) < 1 \text{ for all } \omega \in \mathbb{R}. \tag{6.11}$$

In the reasonable case of bounded (integrable) measurements $y(t)$, $t \in [0, T]$, and stable filter dynamics, we have that $\sigma \leq 0$, and the convergence condition (6.11) for the waveform relaxation iteration (6.10) equals the condition (6.8) for decentralized stabilization of the filer dynamics. We now propose our distributed attack detection filter.

Theorem 6.1.4 ***(Distributed attack detection filter)*** *Consider the descriptor system* (4.1) *and assume that the attack set* K *is detectable, and that the network initial state* $x(0)$ *is known. Let assumptions (A1) through (A7) be satisfied and consider the* distributed attack detection filter

$$\begin{aligned} E\dot{w}^{(k)}(t) &= \big(A_D + GC\big)w^{(k)}(t) + A_C w^{(k-1)}(t) - Gy(t)\,, \\ r(t) &= y(t) - Cw^{(k)}(t)\,, \end{aligned} \tag{6.12}$$

where $k \in \mathbb{N}$*,* $t \in [0, T]$ *for some* $T > 0$*,* $w^{(k)}(0) = x(0)$ *for all* $k \in \mathbb{N}$*, and* $G = \operatorname{blkdiag}(G_1, \dots, G_N)$ *is such that the pair* $(E, A_D + GC)$ *is regular, Hurwitz, and*

$$\rho\left((j\omega E - A_D - GC)^{-1}A_C\right) < 1 \text{ for all } \omega \in \mathbb{R}\,. \tag{6.13}$$

Then $\lim_{k\to\infty} r^{(k)}(t) = 0$ *at all times* $t \in [0,T]$ *if and only if* $u_K(t) = 0$ *at all times* $t \in [0,T]$*. Moreover, in the absence of attacks, the asymptotic filter error* $\lim_{k\to\infty}(w^{(k)}(t) - x(t))$ *is exponentially stable for* $t \in [0,T]$*.*

Proof: Since $w^{(k)}(0) = x(0)$, it follows from Lemma 6.1.3 that the solution $w^{(k)}$ of the iteration (6.12) converges, as $k \to \infty$, to the solution w of the non-iterative filter dynamics (6.7) if condition (6.11) is satisfied with $\sigma = 0$ (due to integrability of $y(t)$, $t \in [0,T]$, and since the pair $(E, A_D + GC)$ is Hurwitz). The latter condition is equivalent to condition (6.13).

Under condition (6.13) and due to the Hurwitz assumption, it follows from Lemma 6.1.2 that the error $e = w - x$ between the state w of the decentralized filter dynamics (6.7) and the state x of the descriptor model (4.1) is asymptotically stable in the absence of attacks. Due to the detectability assumption and by reasoning analogous to the proof of Theorem 6.1.1, it follows that the error dynamics e have no invariant zeros. This concludes the proof of Theorem 6.1.4.

□

Remark 8 ***(Distributed attack detection)*** *The waveform relaxation iteration* (6.10) *can be implemented in the following distributed fashion. Assume that each* *control center i is able to numerically integrate the descriptor system*

$$E_i\dot{w}_i^{(k)}(t) =(A_i + G_iC_i)w_i^{(k)}(t) + \sum_{j\in\mathcal{N}_i^{\text{in}}} A_{ij}w_j^{(k-1)}(t) - G_iy_i(t)\,, \tag{6.14}$$

over a time interval $t \in [0,T]$, with initial condition $w_i^{(k)}(0) = w_{i,0}$, measurements y_i, and the neighboring filter states $w_j^{(k-1)}$ as external inputs. Let $w_j^{(0)}$ be an initial guess of the signal w_j. Each control center $i \in \{1,\dots,N\}$ performs the following operations assuming $k = 0$ at start:

(1) set $k := k+1$, and compute the signal $w_i^{(k)}$ by integrating the local filter equation (6.14)*,*

(2) transmit $w_i^{(k)}$ to the j-th control center if $j \in \mathcal{N}_i^{\text{out}}$

(3) update the input $w_j^{(k)}$ with the signal received from the j-th control center, with $j \in \mathcal{N}_i^{\text{in}}$, and iterate.

If the waveform relaxation is convergent, then, for k sufficiently large, the residuals $r_i^{(k)} = y_i - C_iw_i^{(k)}$ can be used to detect attacks; see Theorem 6.1.4. In summary, our distributed attack detection scheme requires integration capabilities at each control center, knowledge of the measurements $y_i(t)$, $t \in [0,T]$, as well as synchronous discrete-time communication between neighboring control centers. □

Remark 9 ***(Distributed filter design)*** *As discussed in Remark 8, the filter* (6.12) *can be implemented in a distributed fashion. In fact, it is also possible to* *design the filter* (6.12)*, that is, the output injections G_i, in an entirely distributed way. Since $\rho(A) \leq \|A\|_p$ for any matrix A and any induced p-norm, condition* (6.13) *can be relaxed by the small gain criterion to*

$$\left\|(j\omega E - A_D - GC)^{-1} A_C\right\|_p < 1 \text{ for all } \omega \in \mathbb{R}\,. \tag{6.15}$$

With $p = \infty$, in order to satisfy condition (6.15)*, it is sufficient for each control center i to verify the following* quasi-block diagonal dominance *condition [72] for each $\omega \in \mathbb{R}$:*

$$\left\|(j\omega E_i - A_i - G_i C_i)^{-1} \sum\nolimits_{j=1, j\neq i}^{n} A_{ij}\right\|_\infty < 1. \tag{6.16}$$

Note that condition (6.16) *can be checked with local information, and it is a conservative relaxation of condition* (6.13). □

6.2 Monitors for Attack Identification

6.2.1 Complexity of the attack identification problem

In this section we study the problem of attack identification, that is, the problem of identifying from measurements the state and output variables corrupted by the attacker. We start our discussion by showing that this problem is generally *NP-hard*. For a vector $x \in \mathbb{R}^n$, let $\text{supp}(x) = \{i \in \{1, \dots, n\} : x_i \neq 0\}$, let $\|x\|_{\ell_0} = |\text{supp}(x)|$ denote the number of non-zero entries, and for a vector-valued signal $v : \mathbb{R}_{\geq 0} \to \mathbb{R}^n$, let $\|v\|_{\mathcal{L}_0} = |\cup_{t \in \mathbb{R}_{\geq 0}} \operatorname{supp}(v(t))|$. We consider the following cardinality minimization problem: given a descriptor system with dynamic matrices $E, A \in \mathbb{R}^{n \times n}$, measurement matrix $C \in \mathbb{R}^{p \times n}$, and measurement signal $y : \mathbb{R}_{\geq 0} \to \mathbb{R}^p$, find the minimum cardinality input signals $v_x : \mathbb{R}_{\geq 0} \to \mathbb{R}^n$ and $v_y : \mathbb{R}_{\geq 0} \to \mathbb{R}^p$ and an arbitrary initial condition $\xi_0 \in \mathbb{R}^n$ that explain the data y, that is,

$$\begin{aligned} \min_{v_x, v_y, \xi_0} \quad & \|v_x\|_{\mathcal{L}_0} + \|v_y\|_{\mathcal{L}_0} \\ \text{subject to} \quad & E\dot{\xi}(t) = A\xi(t) + v_x(t), \\ & y(t) = C\xi(t) + v_y(t), \\ & \xi(0) = \xi_0 \in \mathbb{R}^n . \end{aligned} \tag{6.17}$$

Lemma 6.2.1 ***(Problem equivalence)*** *Consider the system* (4.1) *with identifiable attack set K. The optimization problem* (6.17) *coincides with the problem of identifying the attack set K given the system matrices E, A, C, and the measurements y, where $K = supp([v_x^\mathsf{T} \; v_y^\mathsf{T}])$.*

Proof: Due to the identifiability of K, the attack identification problem consists of finding the smallest attack set capable of injecting an attack $(B_K u_K, D_K u_K)$ that generates the given measurements y for the given dynamics E, A, C, and some initial condition; see Definition 8. The statement follows since $B = [I, 0]$ and $D = [0, I]$ in (4.1), so that $(B_K u_K, D_K u_K) = (v_x, v_y)$. $\square$

As it turns out, the optimization problem (6.17), or equivalently our identification problem, is generally *NP-hard* [34].

Corollary 6.2.1 ***(Complexity of the attack identification problem)*** *Consider the system* (4.1) *with identifiable attack set K. The attack identification problem given the system matrices E, A, C, and the measurements y is NP-hard.*

Proof: Consider the NP-hard [14] sparse recovery problem $\min_{\bar{\xi}\in\mathbb{R}^n} \|\bar{y} - \bar{C}\bar{\xi}\|_{\ell_0}$, where $\bar{C} \in \mathbb{R}^{p\times n}$ and $\bar{y} \in \mathbb{R}^p$ are given and constant. In order to prove the claimed statement, we show that every instance of the sparse recovery problem can be cast as an instance of (6.17). Let $E = I$, $A = 0$, $C = \bar{C}$, and $y(t) = \bar{y}$ at all times. Notice that $v_y(t) = \bar{y} - C\xi(t)$ and $\xi(t) = \xi(0) + \int_0^t v_x(\tau)d\tau$. The problem (6.17) can be written as

$$\min_{v_x,\,\xi} \|v_x\|_{\mathcal{L}_0} + \|\bar{y} - \bar{C}\xi(t)\|_{\mathcal{L}_0} = \min_{v_x(t),\,\bar{\xi}} \|v_x(t)\|_{\mathcal{L}_0} + \|\bar{y} - \bar{C}\bar{\xi} - \bar{C}\int_0^t v_x(\tau)d\tau\|_{\mathcal{L}_0}, \tag{6.18}$$

where $\bar{\xi} = \xi(0)$. Notice that there exists a minimizer to problem (6.18) with $v_x(t) = 0$ for all t. Indeed, since $\|\bar{y} - \bar{C}\bar{\xi} - \bar{C}\int_0^t v_x(\tau)d\tau\|_{\mathcal{L}_0} = |\cup_{t\in\mathbb{R}_{\geq 0}} \operatorname{supp}(\bar{y} - \bar{C}\bar{\xi} - \bar{C}\int_0^t v_x(\tau)d\tau)| \geq |\operatorname{supp}(\bar{y} - \bar{C}\bar{\xi} - \bar{C}\int_0^0 v_x(\tau)d\tau)| = \|\bar{y} - \bar{C}\bar{\xi}\|_{\ell_0}$, problem (6.18) can be equivalently written as $\min_{\bar{\xi}} \|\bar{y} - \bar{C}\bar{\xi}\|_{\ell_0}$. □

By Corollary 6.2.1 the general attack identification problem is combinatorial in nature, and its general solution will require substantial computational effort.

In the next sections we propose an optimal algorithm with high computational complexity, and a sub-optimal algorithm with low computational complexity. We conclude this section with an example.

Example 2 ***(Attack identification via ℓ_1 regularization)*** *A classical procedure to handle cardinality minimization problems of the form* $\min_{v\in\mathbb{R}^n} \|y - Av\|_{\ell_0}$ *is to use the ℓ_1 regularization* $\min_{v\in\mathbb{R}^n} \|y - Av\|_{\ell_1}$ *[14]. This procedure can be adapted to the optimization problem* (6.17) *after converting it into an algebraic optimization problem, for instance by taking subsequent derivatives of the output y, or by discretizing the continuous-time system* (4.1) *and recording several measurements. As shown in [40], for discrete-time systems the ℓ_1 regularization performs reasonably well in the presence of output attacks. However, in the presence of state attacks such an ℓ_1 relaxation performs generally poorly. In what follows, we develop an intuition when and why this approach fails.*

Consider a consensus system with underlying network graph (sparsity pattern of A) illustrated in Fig. 6.1. The dynamics are described by the nonsingular matrix $E = I$ and the state matrix A depending on the small parameter $0 < \varepsilon \ll 1$ as

$$A = \begin{bmatrix} -0.8 & 0.1 & 0 & 0.2 & 0.5 & 0 & 0 & 0 \\ 0.1 & -0.4-\varepsilon & \varepsilon & 0 & 0 & 0.3 & 0 & 0 \\ 0 & 3\varepsilon & -9\varepsilon & 0 & 0 & 0 & 6\varepsilon & 0 \\ 0.1 & 0 & \varepsilon & -0.5-\varepsilon & 0 & 0 & 0 & 0.4 \\ 0.1 & 0 & 0 & 0 & -0.6 & 0.2 & 0 & 0.3 \\ 0 & 0.4 & 0 & 0 & 0.1 & -0.6 & 0.1 & 0 \\ 0 & 0 & 3\varepsilon & 0 & 0 & 0.4 & -0.6-3\varepsilon & 0.2 \\ 0 & 0 & 0 & 0.3 & 0.2 & 0 & 0.2 & -0.7 \end{bmatrix}.$$

The measurement matrix C and the attack signature B_K are

$$C = \begin{bmatrix} 0&1&0&0&0&0&0&0 \\ 0&0&0&1&0&0&0&0 \\ 0&0&0&0&0&0&1&0 \end{bmatrix}, \; B_K^{\mathsf{T}} = \begin{bmatrix} 0&0&1&0&0&0&0&0 \end{bmatrix},$$

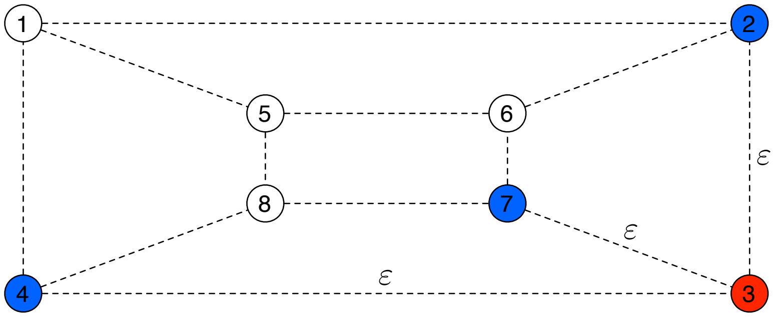


Figure 6.1: *A regular consensus system* (A, B, C)*, where the state variable* 3 *is corrupted by the attacker, and the state variables* 2*,* 4*, and* 7 *are directly measured. Due to the sparsity pattern of* (A, B, C) *any attack of cardinality one is* generically *detectable and identifiable, see [78, 81] for further details.*

and we let $G(s) = C(sI - A)^{-1}B_K$*. It can be verified that the state attack* $K = \{3\}$ *is detectable and identifiable.*

Consider also the state attack $\bar{K} = \{2, 4, 7\}$ *with signature*

$$B_{\bar{K}}^{\mathsf{T}} = \left[\begin{smallmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{smallmatrix}\right],$$

and let $\bar{G}(s) = C(sI - A)^{-1}B_{\bar{K}}$*. We now adopt the shorthands* $u(t) = u_K(t)$ *and* $\bar{u}(t) = u_{\bar{K}}(t)$*, and denote their Laplace transforms by* $U(s)$ *and* $\bar{U}(s)$*, respectively. Notice that* $\bar{G}(s)$ *is right-invertible [6]. Thus,* $Y(s) = G(s)U(s) = \bar{G}(s)\left(\bar{G}^{-1}(s)G(s)U(s)\right)$*. In other words, the measurements* $Y(s)$ *generated by the attack signal* $U(s)$ *can equivalently be generated by the signal* $\bar{U}(s) = \bar{G}^{-1}(s)G(s)U(s)$*. Obviously, we have that* $\|\bar{u}\|_{\mathcal{L}_0} = 3 > \|u\|_{\mathcal{L}_0} = 1$*, that is, the attack set* K *achieves a lower cost than* $\bar{K}$ *in the optimization problem* (6.17)*.*

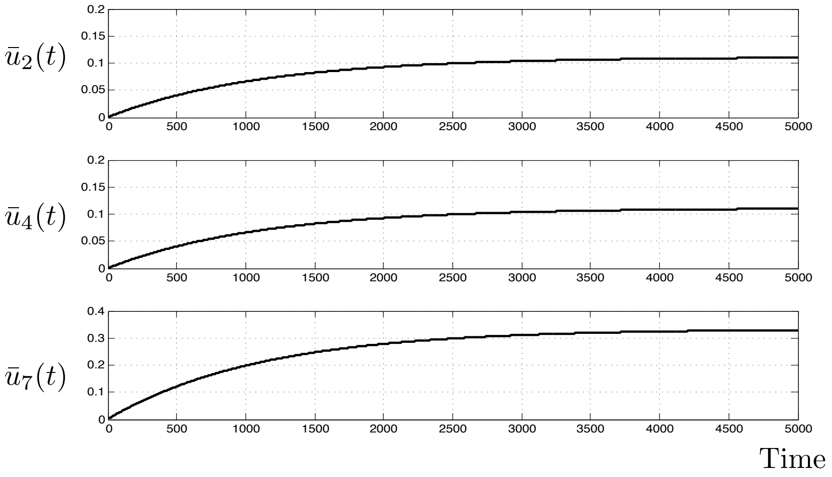


Figure 6.2: *Plot of the attack mode $\bar{u}(t)$ for the attack set $\bar{K} = \{2, 4, 7\}$ to generate the same output as the attack set $K = \{3\}$ with attack mode $u(t) = 1$. Although $|\bar{K}| > |K|$, we have that $|\bar{u}_i(t)| < |u(t)|/3$ for $i \in \{1, 2, 3\}$.*

Consider now the numerical realization $\varepsilon = 0.0001$, $x(0) = 0$, and $u(t) = 1$ for all $t \in \mathbb{R}_{\geq 0}$. The corresponding attack mode $\bar{u}$ is shown in Fig. 6.2. Since $|\bar{u}_i(t)| < 1/3$ for $i \in \{1, 2, 3\}$ and $t \in \mathbb{R}_{\geq 0}$, it follows that $\|u(t)\|_{\ell_p} > \|\bar{u}(t)\|_{\ell_p}$ point-wise in time and $\|u(t)\|_{\mathcal{L}_q/\ell_p} > \|\bar{u}(t)\|_{\mathcal{L}_p/\ell_q}$, where $p, q \geq 1$ and $\|u(t)\|_{\mathcal{L}_q/\ell_p} = \left(\int_0^\infty (\sum_{i=1}^{n+p} |u_i(\tau)|^p)^{q/p} d\tau\right)^{1/q}$ is the $\mathcal{L}_q/\ell_p$-norm. Hence, the attack set $\bar{K}$ achieves a lower cost than K for any algebraic version of the optimization problem (6.17) *penalizing a ℓ_p cost point-wise in time or a $\mathcal{L}_q/\ell_p$ cost over a time interval. Since $\|\bar{u}\|_{\mathcal{L}_0} > \|u\|_{\mathcal{L}_0}$, we conclude that, in general, the identification problem cannot be solved by a point-wise ℓ_p or $\mathcal{L}_q/\ell_p$ regularization for any $p, q \geq 1$.*

Notice that, for any choice of network parameters, a value of ε can be found such that a point-wise ℓ_p or a $\mathcal{L}_q/\ell_p$ regularization procedure fails at identifying *the attack set. Moreover, large-scale stable systems often exhibit this behavior independently of the system parameters. This can be easily seen in discrete-time systems, where a state attack with attack set K affects the output via the matrix $CA^{r-1}B_K$, where r is the relative degree of (A, B_K, C). Hence, if A is Schur stable and thus $\lim_{k\to\infty} A^k = 0$, then $CA^{r-1}B_K$ converges to the zero matrix for increasing relative degree. In this case, an attack closer to the sensors may achieve a lower $\mathcal{L}_q/\ell_p$ cost than an attack far from sensors independently of the cardinality of the attack set. In short, the ϵ-connections in Fig. 6.1 can be thought of as the effect of a large relative degree in a stable system.* □

6.2.2 A centralized attack identification monitor

As previously shown, unlike the detection case, the identification of the attack set K requires a combinatorial procedure, since, a priori, K is one of the $\binom{n+p}{|K|}$ possible attack sets. The following centralized attack identification procedure consists of designing a residual filter to determine whether a predefined set coincides with the attack set. The design of this residual filter consists of three steps – an input output transformation (see Lemma 6.2.2), a state transformation to a suitable conditioned-invariant subspace (see Lemma 6.2.3), and an output injection and definition of a proper residual (see Theorem 6.2.4).

As a first design step, we show that the identification problem can be carried out for a modified system without corrupted measurements, that is, without the feedthrough matrix D.

Lemma 6.2.2 ***(Attack identification with safe measurements)*** *Consider the descriptor system* (4.1) *with attack set K. The attack set K is identifiable for the descriptor system* (4.1) *if and only if it is identifiable for the following descriptor system:*

$$\begin{aligned} E\dot{x}(t) &= (A - B_K D_K^{\dagger} C)x(t) + B_K(I - D_K^{\dagger} D_K)u_K(t), \\ \tilde{y}(t) &= (I - D_K D_K^{\dagger})Cx(t). \end{aligned} \tag{6.19}$$

Proof: Due to the identifiability hypothesis, there exists no attack set R with $|R| \leq |K|$ and $R \neq K$, $s \in \mathbb{C}$, $g_K \in \mathbb{R}^{|K|}$, $g_R \in \mathbb{R}^{|R|}$, and $x \in \mathbb{R}^n \setminus \{0\}$ such that

$$\left[\begin{array}{c|c|c} sE - A & -B_K & -B_R \\ \hline C & D_K & D_R \\ \hline C & D_K & D_R \end{array}\right] \begin{bmatrix} x \\ g_K \\ g_R \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \tag{6.20}$$

where we added an additional (redundant) output equation; see Chapter 4. A multiplication of equation (6.20) from the left by the projectors

$\text{blkdiag}\big(I\,,\ D_K D_K^\dagger\,,\ (I - D_K D_K^\dagger)\big)$ yields

$$\left[\begin{array}{c|c|c} sE - A & -B_K & -B_R \\ \hline D_K D_K^\dagger C & D_K & D_K D_K^\dagger D_R \\ \hline (I - D_K D_K^\dagger)C & 0 & (I - D_K D_K^\dagger) D_R \end{array}\right] \begin{bmatrix} x \\ g_K \\ g_R \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}.$$

The variable g_K can be eliminated in the first redundant (corrupted) output equation according to

$$g_K = -D_K^\dagger C x - D_K^\dagger D_R g_R + (I - D_K^\dagger D_K) g_K.$$

Thus, $P(s)[x^\mathsf{T}\ g_K^\mathsf{T}\ g_R^\mathsf{T}]^\mathsf{T} = 0$ has no solution, where $P(s)$ is

$$\left[\begin{array}{c|c|c} sE - A + B_K D_K^\dagger C & -B_K(I - D_K^\dagger D_K) & -B_R + B_K D_K^\dagger D_R \\ \hline (I - D_K D_K^\dagger)C & 0 & (I - D_K D_K^\dagger) D_R \end{array}\right]$$

The statement follows. □

The second design step of our attack identification monitor relies on the concept of *conditioned invariant subspace.* We refer to [6, 35, 54] for a comprehensive discussion of conditioned invariant subspaces. Let $\mathcal{S}^*$ be the conditioned invariant subspace associated with the system (E, A, B, C, D), that is, the smallest subspace of the state space satisfying

$$\mathcal{S}^* = \begin{bmatrix} A & B \end{bmatrix} \left(\begin{bmatrix} E^{-1}\mathcal{S}^* \\ \mathbb{R}^m \end{bmatrix} \cap \operatorname{Ker} \begin{bmatrix} C & D \end{bmatrix} \right), \tag{6.21}$$

and let L be an output injection matrix satisfying

$$\begin{bmatrix} A + LC & B + LD \end{bmatrix} \begin{bmatrix} E^{-1}\mathcal{S}^* \\ \mathbb{R}^m \end{bmatrix} \subseteq \mathcal{S}^*. \tag{6.22}$$

We transform the descriptor system (6.19) into a set of canonical coordinates representing $\mathcal{S}^*$ and its orthogonal complement. For a nonsingular system ($E = I$) such an equivalent state representation can be achieved by a nonsingular transformation of the form $Q^{-1}(sI - A)Q$. However, for a singular system different transformations need to be applied in the domain and codomain such as $P^{\mathsf{T}}(sE - A)Q$ for nonsingular P and Q.

Lemma 6.2.3 ***(Input decoupled system representation)*** *For the system* (6.19), *let* $\mathcal{S}^*$ *and* L *be as in* (6.21) *and* (6.22), *respectively. Define the unitary matrices* $P = \begin{bmatrix} \text{Basis}(\mathcal{S}^*), & \text{Basis}((\mathcal{S}^*)^{\perp}) \end{bmatrix}$ *and* $Q = \begin{bmatrix} \text{Basis}(E^{-1}\mathcal{S}^*), & \text{Basis}((E^{-1}\mathcal{S}^*)^{\perp}) \end{bmatrix}$. *Then*

$$P^{\mathsf{T}}EQ = \begin{bmatrix} \tilde{E}_{11} & \tilde{E}_{12} \\ 0 & \tilde{E}_{22} \end{bmatrix}, P^{\mathsf{T}}(A - B_K D_K^{\dagger} C + LC)Q = \begin{bmatrix} \tilde{A}_{11} & \tilde{A}_{12} \\ 0 & \tilde{A}_{22} \end{bmatrix},$$

$$P^{\mathsf{T}}B_K(I - D_K^{\dagger}D_K) = \begin{bmatrix} \tilde{B}_K(t) \\ 0 \end{bmatrix}, (I - D_K D_K^{\dagger})C)Q = \begin{bmatrix} \tilde{C}_1 & \tilde{C}_2 \end{bmatrix}.$$

The attack set K is identifiable for the descriptor system (4.1) *if and only if it is identifiable for the descriptor system*

$$\begin{bmatrix} \tilde{E}_{11} & \tilde{E}_{12} \\ 0 & \tilde{E}_{22} \end{bmatrix} \begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \end{bmatrix} = \begin{bmatrix} \tilde{A}_{11} & \tilde{A}_{12} \\ 0 & \tilde{A}_{22} \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} + \begin{bmatrix} \tilde{B}_K(t) \\ 0 \end{bmatrix},$$

$$y(t) = \begin{bmatrix} \tilde{C}_1 & \tilde{C}_2 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix}. \tag{6.23}$$

Proof: Let $\mathcal{L} = E^{-1}\mathcal{S}^*$ and $\mathcal{M} = \mathcal{S}^*$. Notice that $(A + LC)E^{-1}\mathcal{S}^* \subseteq \mathcal{S}^*$ by the invariance property of $\mathcal{S}^*$ [35, 54]. It follows that $\mathcal{L}$ and $\mathcal{M}$ are a pair of *right deflating subspaces* for the matrix pair $(A+LC, E)$ [41], that is, $\mathcal{M} = A\mathcal{L}+E\mathcal{L}$ and $\dim(\mathcal{M}) \leq \dim(\mathcal{L})$. The sparsity pattern in the descriptor and dynamic matrices $\tilde{E}$ and $\tilde{A}$ of (6.23) arises by construction of the right deflating subspaces P and Q [41, Eq. (2.17)], and the sparsity pattern in the input matrix arises due to the invariance properties of $\mathcal{S}^*$ containing $\mathrm{Im}(B_K)$. The statement follows because the output injection L, the coordinate change $x \mapsto Q^{-1}x$, and the left-multiplication of the dynamics by P^{T} does not affect the existence of zero dynamics. □

We call system (6.23) the *conditioned system* associated with (4.1). For the ease of notation and without affecting generality, the third and final design step of our attack identification filter is presented for the conditioned system (6.23).

Theorem 6.2.4 ***(Attack identification filter for attack set*** K***)*** *Consider the* conditioned system (6.23) *associated with the descriptor system* (4.1). *Assume that the attack set is identifiable, the network initial state* $x(0)$ *is known, and the assumptions (A1) through (A3) are satisfied. Consider the* attack identification filter for the attack signature (B_K, D_K)

$$\begin{aligned}\tilde{E}_{22}\dot{w}_2(t) &= (\tilde{A}_{22} + \tilde{G}(I - \tilde{C}_1\tilde{C}_1^{\dagger})\tilde{C}_2)w_2(t) - \tilde{G}\bar{y}(t),\\ r_K(t) &= (I - \tilde{C}_1\tilde{C}_1^{\dagger})\tilde{C}_2 w_2(t) - \bar{y}(t), \quad \textit{with}\\ \bar{y}(t) &= (I - \tilde{C}_1\tilde{C}_1^{\dagger})y(t),\end{aligned} \tag{6.24}$$

where $w_2(0) = x_2(0)$*, and* $\tilde{G}$ *is such that* $(\tilde{E}_{22}, \tilde{A}_{22} + \tilde{G}(I - \tilde{C}_1\tilde{C}_1^{\dagger})\tilde{C}_2)$ *is Hurwitz. Then* $r_K(t) = 0$ *for all times* $t \in \mathbb{R}_{\geq 0}$ *if and only if* K *coincides with the attack set.*

Proof: Let $w = [w_1^{\mathsf{T}}\ w_2^{\mathsf{T}}]^{\mathsf{T}}$, where w_1 obeys

$$\tilde{E}_{11}\dot{w}_1(t) + \tilde{E}_{12}\dot{w}_2(t) = \tilde{A}_{11}w_1(t) + \tilde{A}_{12}w_2(t).$$

Consider the filter error $e = w - x$, and notice that

$$\begin{bmatrix}\tilde{E}_{11} & \tilde{E}_{12}\\ 0 & E_{22}\end{bmatrix}\begin{bmatrix}\dot{e}_1(t)\\ \dot{e}_2(t)\end{bmatrix} = \begin{bmatrix}\tilde{A}_{11} & \tilde{A}_{12}\\ 0 & \bar{A}_{22}\end{bmatrix}\begin{bmatrix}e_1(t)\\ e_2(t)\end{bmatrix} - \begin{bmatrix}\tilde{B}_K\\ 0\end{bmatrix}u_K(t),$$

$$r_K(t) = (I - \tilde{C}_1\tilde{C}_1^{\dagger})\tilde{C}_2 e_2(t),$$

where $\bar{A}_{22} = \tilde{A}_{22} + \tilde{G}(I - \tilde{C}_1\tilde{C}_1^{\dagger})\tilde{C}_2$. Notice that r_K is not affected by the input u_K, so that, since $e_2(0) = 0$ due to $w_2(0) = x_2(0)$, the residual r_K is identically zero when K is the attack set. In order to prove the theorem we are left to show that for every set R, with $|R| \leq |K|$ and $R \cap K = \emptyset$, every attack mode u_R results in a nonzero residual r_K. From the discussion in Chapter 4, and the identifiability hypothesis, for any $R \neq K$, there exists no solution to

$$\left[\begin{array}{cc|c|c} s\tilde{E}_{11} - \tilde{A}_{11} & s\tilde{E}_{12} - \tilde{A}_{12} & \tilde{B}_K & -B_{R1} \\ 0 & s\tilde{E}_{22} - \bar{A}_{22} & 0 & -B_{R2} \\ \hline \tilde{C}_1 & \tilde{C}_2 & 0 & D_R \end{array}\right] \begin{bmatrix} x_1 \\ x_2 \\ g_K \\ g_R \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}.$$

A projection of the equation $0 = \tilde{C}_1 x_1 + \tilde{C}_2 x_2 + D_R g_R$ onto the image of $\tilde{C}_1$ and its orthogonal complement yields

$$\left[\begin{array}{cc|c|c} s\tilde{E}_{11} - \tilde{A}_{11} & s\tilde{E}_{12} - \tilde{A}_{12} & B_K & -B_{R1} \\ 0 & s\tilde{E}_{22} - \bar{A}_{22} & 0 & -B_{R2} \\ \hline \tilde{C}_1 & \tilde{C}_1 \tilde{C}_1^\dagger \tilde{C}_2 & 0 & \tilde{C}_1 \tilde{C}_1^\dagger D_R \\ 0 & (I - \tilde{C}_1 \tilde{C}_1^\dagger)\tilde{C}_2 & 0 & (I - \tilde{C}_1 \tilde{C}_1^\dagger) D_R \end{array}\right] \begin{bmatrix} x_1 \\ x_2 \\ g_K \\ g_R \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}. \qquad (6.25)$$

Due to the identifiability hypothesis the set of equations (6.25) features no solution $[x_1^\mathsf{T} \; x_2^\mathsf{T} \; g_K^\mathsf{T} \; g_R^\mathsf{T}]^\mathsf{T}$ with $[x_1^\mathsf{T} \; x_2^\mathsf{T}]^\mathsf{T} = 0$.

Observe that, for every x_2 and g_R, there exists $x_1 \in \operatorname{Ker}(\tilde{C}_1)^\perp$ such that the third equation of (6.25) is satisfied. Furthermore, for every x_2 and g_R, there exist $x_1 \in \operatorname{Ker}(\tilde{C}_1)$ and g_K such that the first equation of (6.25) is satisfied. Indeed, since $QE^{-1}\mathcal{S}^* = \begin{bmatrix} \operatorname{Im}(I) \; 0 \end{bmatrix}^\mathsf{T}$ and $P^\mathsf{T}\mathcal{S}^* = \begin{bmatrix} \operatorname{Im}(I) \; 0 \end{bmatrix}^\mathsf{T}$, the invariance of $\mathcal{S}^*$ implies

Algorithm 4: *Identification Monitor for* (B_K, D_K)

```
Input      : Matrices $E$, $A$, $B_K$, and $D_K$,;
Require    : Identifiability of attack set $K$;
From system (4.1) define the system (6.19);
Compute $\mathcal{S}^*$ and $L$ for system (6.19) as in (6.21) and (6.22);
Apply $L$, $P$, and $Q$ as in Lemma 6.2.3 leading to system (6.23);
For (6.23), define $r_K$ and apply the output injection $\bar{G}$ as in (6.24).
```

that $\mathcal{S}^* = A(E^{-1}\mathcal{S}^* \cap \operatorname{Ker}(C)) + \operatorname{Im}(B_K)$, or equivalently in new coordinates, $\operatorname{Im}(I) = \tilde{A}_{11} \operatorname{Ker}(\tilde{C}_1) + \operatorname{Im}(\tilde{B}_K)$. Finally note that $\left[(s\tilde{E}_{11} - \tilde{A}_{11}) \operatorname{Ker}(\tilde{C}_1) \; \tilde{B}_K\right]$ is of full row rank due to the controllability of the subspace $\mathcal{S}^*$ [35]. We conclude that there exist no vectors x_2 and g_R such that $(s\tilde{E}_{22} - \bar{A}_{22})x_2 - B_{R2}g_R = 0$ and $(I - \tilde{C}_1\tilde{C}_1^\dagger)(\tilde{C}_2x_2 + D_Rg_R) = 0$ and the statement follows. □

Our identification procedure is summarized in Algorithm 4. Observe that the proposed attack identification filter extends classical results concerning the design of unknown-input fault detection filters. In particular, our filter generalizes the construction of [62] to descriptor systems with direct feedthrough matrix. Additionally, we guarantee the absence of invariant zeros in the residual dynamics. By doing so, our attack identification filter is sensitive to *every* attack mode. Notice that classical fault detection filters, for instance those presented in [62], are guaranteed to detect and isolate signals that do not excite exclusively zero dynamics. Finally, an attack identification filter for the case of state space or index-one systems is presented in our previous work [80].

Remark 10 ***(Complexity of centralized identification)*** *Our centralized identification procedure assumes the knowledge of the cardinality k of the attack set, and it achieves identification of the attack set by constructing a residual generator for $\binom{n+p}{k}$ possible attack sets. Thus, for each finite value of k, our procedure constructs $O(n^k)$ filters. If only an upper bound $\bar{k}$ on the cardinality of the attack set is available, identification can be achieved by constructing $\binom{n+p}{\bar{k}}$ filters, and by intersecting the attack sets generating zero residuals.* □

Remark 11 ***(Attack identification filter in the presence of noise)*** *Let the dynamics and the measurements of the system* (4.1) *be affected, respectively, by the additive white noise signals η, with $\mathbb{E}[\eta(t)\eta^{\mathsf{T}}(\tau)] = R_\eta\delta(t-\tau)$, and $\zeta(t)$, with $\mathbb{E}[\zeta(t)\zeta^{\mathsf{T}}(\tau)] = R_\zeta\delta(t-\tau)$. Let the state and output noise be independent of each other. Then, simple calculations show that the dynamics and the output of the attack identification filter* (6.24) *are affected, respectively, by the noise signals*

$$\hat{\eta}(t) = P^{\mathsf{T}}\eta(t) + P^{\mathsf{T}}(L(I - D_K D_K^{\dagger}) - B_K D_K^{\dagger})\zeta(t),$$

$$\hat{\zeta}(t) = -\Big(I - \Big[(I - D_K D_K^{\dagger})CQ_1\Big]\Big[(I - D_K D_K^{\dagger})CQ_1\Big]^{\dagger}(I - D_K D_K^{\dagger})\Big)\zeta(t),$$

where $Q_1 = \mathrm{Basis}(E^{-1}\mathcal{S}^*)$. *Define the covariance matrix*

$$R_{\hat{\eta},\hat{\zeta}} = \mathbb{E}\left(\begin{bmatrix}\hat{\eta}(t)\\ \hat{\zeta}(t)\end{bmatrix}\begin{bmatrix}\hat{\eta}^{\mathsf{T}}(t) & \hat{\zeta}^{\mathsf{T}}(t)\end{bmatrix}\right).$$

Notice that the off-diagonal elements of $R_{\hat{\eta},\hat{\zeta}}$ *are in general nonzero, that is, the state and output noises of the attack identification filter are not independent of each other. As in the detection case, by using the covariance matrix* $R_{\hat{\eta},\hat{\zeta}}$, *the output injection matrix* $\tilde{G}$ *in* (6.24) *can be designed to optimize the robustness of the residual* r_K *against noise. A related example is in Section 4.4.* □

We conclude this section by observing that a distributed implementation of our attack identification scheme is not practical. Indeed, even if the filters parameters may be obtained via distributed computation, still $\binom{n+p}{k}$ filters would need to be implemented to identify an attack of cardinality k. Such a distributed implementation results in an enormous communication effort and does not reduce the fundamental combinatorial complexity.

6.2.3 A fully decoupled attack identification monitor

In the following sections we develop a distributed attack identification procedure. Consider the decentralized setup presented in Section 6.1.2 with assumptions (A4)-(A7). The subsystem assigned to the i-th control center is

$$\begin{aligned} E_i\dot{x}_i(t) &= A_i x_i(t) + \sum_{j\in\mathcal{N}_i^{\text{in}}} A_{ij}x_j(t) + B_{K_i}u_{K_i}(t), \\ y_i(t) &= C_i x_i(t) + D_{K_i}u_{K_i}(t), \;\; i \in \{1,\dots,N\}, \end{aligned} \tag{6.26}$$

where $K_i = (K \cap V_i) \cup K_i^{\text{p}}$ with K being the attack set and K_i^{p} being the set of corrupted measurements in the region G_{t}^i.

As a first distributed identification method we consider the fully decoupled case (no cooperation among control centers). In the spirit of [93], the neighboring states x_j affecting x_i are treated as unknown inputs (f_i) to the i-th subsystem:

$$\begin{aligned} E_i\dot{x}_i(t) &= A_i x_i(t) + B_i^{\text{b}} f_i(t) + B_{K_i}u_{K_i}(t), \\ y_i(t) &= C_i x_i(t) + D_{K_i}u_{K_i}(t), \;\; i \in \{1,\dots,N\}, \end{aligned} \tag{6.27}$$

where $B_i^{\text{b}} = [A_{i1} \cdots A_{i,i-1}\, A_{i,i+1} \cdots A_{iN}]$. We refer to (6.27) as to the *i-th decoupled system*, and we let $K_i^{\text{b}} \subseteq V_i$ be the set of *boundary nodes* of (6.27), that is, the nodes $j \in V_i$ with $A_{jk} \neq 0$ for some $k \in \{1,\dots,n\} \setminus V_i$.

If the attack identification procedure in Section 6.2.2 is designed for the i-th decoupled system (6.27) subject to unknown inputs f_i and u_{K_i}, then a total of only $\sum_{i=1}^N \binom{n_i+p_i}{|K_i|} < \binom{n+p}{|K|}$ need to be designed. Although the combinatorial complexity of the identification problem is tremendously reduced, this decoupled identification procedure has several limitations. The following fundamental limitations follow from the discussion in Chapter 4:

(L1) if $(E_i, A_i, B_{K_i}, C_i, D_{K_i})$ has invariant zeros, then K_i is not detectable by the i-th control center;

(L2) if there is an attack set R_i, with $|R_i| \leq |K_i|$, such that $(E_i, A_i, [B_{K_i}\ B_{R_i}], C_i, [D_{K_i}\ D_{R_i}])$ has invariant zeros, then K_i is not identifiable by the i-th control center;

(L3) if $K_i \not\subseteq K_i^{\mathrm{b}}$ and $(E_i, A_i, [B_i^{\mathrm{b}}\ B_{K_i}], C_i, D_{K_i})$ has no invariant zeros, then K_i is detectable by the i-th control center; and

(L4) if $K_i \not\subseteq K_i^{\mathrm{b}}$ and there is no attack set R_i, with $|R_i| \leq |K_i|$, such that $(E_i, A_i, [B_i^{\mathrm{b}}\ B_{K_i}\ B_{R_i}],\ C_i, [D_{K_i}\ D_{R_i}])$ has invariant zeros, then K_i is identifiable by the i-th control center.

Whereas limitations (L1) and (L2) also apply to any centralized attack detection and identification monitor, limitations (L3) and (L4) arise by naively treating the neighboring signals as unknown inputs. Since, in general, the i-th control center cannot distinguish between an unknown input from a safe subsystem, an unknown input from a corrupted subsystem, and a boundary attack with the same input direction, we can further state that

(L5) any (boundary) attack set $K_i \subseteq K_i^{\mathrm{b}}$ is not detectable and not identifiable by the i-th control center, and

(L6) any (external) attack set $K \setminus K_i$ is not detectable and not identifiable by the i-th control center.

We remark that, following our graph-theoretic analysis in Section 4.3, the attack K_i is generically identifiable by the i-th control center if the number of attacks $|K_i|$ on the i-th subsystem is sufficiently small, the internal connectivity of the i-th subsystem (size of linking between unknown inputs/attacks and outputs) is sufficiently high, and the number of unknown signals $|K_i^{\mathrm{b}}|$ from neighboring subsystems is sufficiently small. These criteria can ultimately be used to select an attack-resilient partitioning of a cyber-physical system.

6.2.4 A cooperative attack identification monitor

In this section we improve upon the naive fully decoupled method presented in Subsection 6.2.3 and propose an identification method based upon a divide and conquer procedure with cooperation. This method consists of the following steps.

(S1: estimation and communication) Each control center estimates the state of its own region by means of an *unknown-input observer* for the i-th subsystem subject to the unknown input $B_i^{\mathrm{b}} f_i$. For this task we build upon existing unknown-input estimation algorithms (see the Section 6.3 for a constructive procedure). Assume that the state x_i is reconstructed modulo some subspace $\mathcal{F}_i$.[2] Let $F_i =$

[2] For nonsingular systems without feedthrough matrix, $\mathcal{F}_i$ is the largest (A_i, B_i^{b})-controlled invariant subspace contained in $\mathrm{Ker}(C_i)$ [6].

Basis($\mathcal{F}_i$), and let $x_i = \tilde{x}_i + \hat{x}_i$, where $\hat{x}_i$ is the estimate computed by the i-th control center, and $\tilde{x}_i \in \mathcal{F}_i$. Assume that each control center i transmits the estimate $\hat{x}_i$ and the uncertainty subspace F_i to every neighboring control center.

(S2: residual generation) Observe that each input signal $A_{ij}x_j$ can be written as $A_{ij}x_j = A_{ij}\tilde{x}_j + A_{ij}\hat{x}_j$, where $\tilde{x}_j \in \mathcal{F}_j$. Then, after carrying out step (S1), only the inputs $A_{ij}\tilde{x}_j$ are unknown to the i-th control center, while the inputs $A_{ij}\hat{x}_j$ are known to the i-th center due to communication. Let $B_i^{\mathrm{b}} F_i = [A_{i1}F_1 \; \cdots \; A_{i,i-1}F_{i-1} \; A_{i,i+1}F_{i+1} \; \cdots \; A_{iN}F_N]$, and rewrite the signal $B_i^{\mathrm{b}}\tilde{x}$ as $B_i^{\mathrm{b}}\tilde{x} = B_i^{\mathrm{b}} F_i f_i$, for some unknown signal f_i. Then the dynamics of the i-th subsystem read as

$$E_i \dot{x}_i(t) = A_i x_i(t) + B_i^{\mathrm{b}} \hat{x}(t) + B_i^{\mathrm{b}} F_i f_i(t) + B_{K_i} u_{K_i}(t).$$

Analogously to the filter presented in Theorem 6.2.4 for the attack signature (B_K, D_K), consider now the following filter (in appropriate coordinates) for (6.27) for the signature $(B_i^{\mathrm{b}} F_i, 0)$

$$\begin{aligned} E_i \dot{w}_i(t) &= (A_i + L_i C_i) w_i(t) - L y(t) + B_i^{\mathrm{b}} \bar{x}(t), \\ r_i(t) &= M w_i(t) - H y(t), \end{aligned} \tag{6.28}$$

where L_i is the injection matrix associated with the conditioned invariant subspace generated by $B_i^{\mathrm{b}} F_i$, with $(E_i, A_i + L_i C_i)$ Hurwitz, and $\bar{x}$ is the state transmitted to i by its neighbors. Notice that, in the absence of attacks in the regions $\mathcal{N}_i^{\mathrm{in}}$, we have $B_i^{\mathrm{b}}\bar{x} = B_i^{\mathrm{b}}\hat{x}$. Finally, let the matrices M and H in (6.28) be chosen so that the input $B_i^{\mathrm{b}}F_if_i$ does not affect the residual r_i.[3] Consider the filter error $e_i = w_i - x_i$, and notice that

$$\begin{aligned} E_i\dot{e}_i(t) &= (A_i + L_iC_i)e_i(t) + B_i^{\mathrm{b}}(\bar{x}(t) - \hat{x}(t)) - B_{K_i}u_{K_i}(t) \quad - B_i^{\mathrm{b}}F_if_i(t), \quad (6.29) \\ r_i(t) &= Me_i(t), \end{aligned}$$

(S3: cooperative residual analysis) We next state a key result for our distributed identification procedure.

Lemma 6.2.5 ***(Characterization of nonzero residuals)*** *Let each control center implement the distributed identification filter* (6.28) *with* $w_i(0) = x_i(0)$*. Assume that the attack K affects only the i-th subsystem, that is $K = K_i$. Assume that $(E_i, A_i, [B_i^{\mathrm{b}}F_i \, B_{K_i}], C_i)$ and $(E_i, A_i, B_i^{\mathrm{b}}, C_i)$ have no invariant zeros. Then,*

(i) $r_i(t) \neq 0$ *at some time t, and*

(ii) either $r_j(t) = 0$ for all $j \in \mathcal{N}_i^{\mathrm{out}}$ at all times t, or $r_j(t) \neq 0$ for all $j \in \mathcal{N}_i^{\mathrm{out}}$ at some time t.

Proof: Notice that the estimation computed by a control center is correct provided that its area is not under attack. In other words, since $K = K_i$, we have that $B_i^{\mathrm{b}}\hat{x} = B_i^{\mathrm{b}}\bar{x}$ in (6.29). Since $(E_i, A_i, [B_i^{\mathrm{b}}F_i \, B_{K_i}], C_i)$ has no invariant zeros,

[3] See Section 6.2.2 for a detailed construction of this type of filter.

statement (i) follows. In order to prove statement (ii), consider the following two cases: the i-th control center provides the correct estimation $\hat{x}_i = \bar{x}_i$ or an incorrect estimation $\hat{x}_i \neq \bar{x}_i$. For instance, if $\operatorname{Im}(B_{K_i}) \subseteq \operatorname{Im}(B_i^{\mathrm{b}})$, that is, the attack set K_i lies on the boundary of the i-th area, then $\hat{x}_i = \bar{x}_i$. Notice that, if $\hat{x}_i = \bar{x}_i$, then each residual r_j, $j \neq i$, is identically zero since the associated residual dynamics (6.29) evolve as an autonomous system without inputs. Suppose now that $\hat{x}_i \neq \bar{x}_i$. Notice that $B_i^{\mathrm{b}} F_i f_i + B_i^{\mathrm{b}}(\hat{x} - \bar{x}) \in \operatorname{Im}(B_i^{\mathrm{b}})$. Then, since $(E_i, A_i, B_i^{\mathrm{b}}, C_i)$ has no invariant zeros, each residual $r_j(t)$ is nonzero for some t. □

As a consequence of Lemma 6.2.5 the region under attack can be identified through a distributed procedure. Indeed, the i-th area is safe if either of the following two criteria is satisfied:

(C1) the associated residual r_i is identically zero, or

(C2) the neighboring areas $j \in \mathcal{N}_i^{\mathrm{out}}$ feature both zero and nonzero residuals r_j.

Consider now the case of several simultaneously corrupted subsystems. Then, if the graphical distance between any two corrupted areas is at least 2, that is, if there are at least two uncorrupted areas between any two corrupted areas, corrupted areas can be identified via our distributed method and criteria (C1) and (C2). An upper bound on the maximum number of identifiable concurrent corrupted areas can consequently be derived (see the related *set packing* problem in [34]).

(S4: local identification) Once the corrupted regions have been identified, the identification method in Section 6.2 is used to identify the local attack set.

Lemma 6.2.6 ***(Local identification)*** *Consider the decoupled system* (6.27)*. Assume that the i-th region is under the attack K_i whereas the neighboring regions $\mathcal{N}_i^{\mathrm{out}}$ are uncorrupted. Assume that each control center $j \in \mathcal{N}_i^{\mathrm{in}}$ transmits the estimate $\hat{x}_j$ and the uncertainty subspace F_i to the i-th control center. Then, the attack set K_i is identifiable by the i-th control center if, for any attack set R_i, with $|R_i| \leq |K_i|$, $(E_i, A_i, [B_i^{\mathrm{b}} F_i \; B_{K_i} \; B_{R_i}], C_i, [D_{K_i} \; D_{R_i}])$ has no invariant zeros.*

Proof: Notice that each control center j, with $j \neq i$, can correctly estimate the state x_j modulo $\mathcal{F}_j$. Since this estimation is transmitted to the i-th control center, the statement follows from the discussion in Chapter 4. □

The final identification procedure **(S4)** is implemented only on the corrupted regions. Consequently, the combinatorial complexity of our distributed identification procedure is $\sum_{i=1}^{\ell} \binom{n_i+p_i}{|K_i|}$, where ℓ is the number of corrupted regions. Hence, the distributed identification procedure greatly reduces the combinatorial complexity of the centralized procedure presented in Subsection 6.2.2, which requires the implementation of $\binom{n+p}{|K|}$ filters. Finally, the assumptions of Lemma

6.2.5 and Lemma 6.2.6 clearly improve upon the limitations (L3) and (L4) of the naive decoupled approach presented in Subsection 6.2.3. We conclude this section with an example showing that, contrary to the limitation (L5) of the naive fully decoupled approach, boundary attacks $K_i \subseteq K_i^{\mathrm{b}}$ can be identified by our cooperative attack identification method.

6.3 State Reconstruction for Descriptor Systems

In this section we present an algebraic technique to reconstruct the state of a descriptor system. Our method builds upon the results presented in [8]. Consider the descriptor model (4.1) written in the form

$$\begin{aligned} \dot{x}_1(t) &= A_{11}x_1(t) + A_{12}x_2(t) + B_1u(t)\,, \\ 0 &= A_{21}x_1(t) + A_{22}x_2(t) + B_2u(t)\,, \\ y(t) &= C_1x_1(t) + C_2x_2(t) + Du(t)\,. \end{aligned} \tag{6.30}$$

We aim at characterizing the largest subspace of the state space of (6.30) that can be reconstructed through the measurements y. Consider the associated nonsingular system

$$\begin{aligned} \dot{\tilde{x}}_1(t) &= A_{11}\tilde{x}_1(t) + B_1\tilde{u}(t) + A_{12}\tilde{x}_2(t), \\ \tilde{y}(t) &= \begin{bmatrix} \tilde{y}_1(t) \\ \tilde{y}_2(t) \end{bmatrix} = \begin{bmatrix} A_{21} \\ C_1 \end{bmatrix} \tilde{x}_1(t) + \begin{bmatrix} A_{22} & B_2 \\ C_2 & D \end{bmatrix} \begin{bmatrix} \tilde{x}_2(t) \\ \tilde{u}(t) \end{bmatrix}. \end{aligned} \tag{6.31}$$

Recall from [6, Section 4] that the state of the system (6.31) can be reconstructed modulo its largest controlled invariant subspace $\mathcal{V}_1^*$ contained in the null space of the output matrix.

Lemma 6.3.1 ***(Reconstruction of the state*** x_1***)*** *Let* $\mathcal{V}_1^*$ *be the largest controlled invariant subspace of the system* (6.31)*. The state* x_1 *of the system* (6.30) *can be reconstructed only modulo* $\mathcal{V}_1^*$ *through the measurements* y*.*

Proof: We start by showing that for every $x_1(0) \in \mathcal{V}_1^*$ there exist x_2 and u such that y is identically zero. Due to the linearity of (6.30), we conclude that the projection of $x_1(t)$ onto $\mathcal{V}_1^*$ cannot be reconstructed. Notice that for every $\tilde{x}_1(0)$, $\tilde{x}_2$, and $\tilde{u}$ yielding $\tilde{y}_1(t) = 0$ at all times, the state trajectory $[\tilde{x}_1 \; \tilde{x}_2]$ is a solution to (6.30) with input $u = \tilde{u}$ and output $y = \tilde{y}_2$. Since for every $\tilde{x}_1(0) \in \mathcal{V}_1^*$, there exists $\tilde{x}_2$ and $\tilde{u}$ such that $\tilde{y}$ is identically zero, we conclude that every state $x_1(0) \in \mathcal{V}_1^*$ cannot be reconstructed.

We now show that the state x_1 can be reconstructed modulo $\mathcal{V}_1^*$. Let $x_1(0)$ be orthogonal to $\mathcal{V}_1^*$, and let x_1, x_2, and y be the solution to (6.30) subject to the input u. Notice that $\tilde{x}_1 = x_1$, $\tilde{y}_1 = 0$, and $\tilde{y}_2 = y$ is the solution to (6.31) with inputs $\tilde{x}_2 = x_2$ and $\tilde{u} = u$. Since $\tilde{x}_1(0)$ is orthogonal to $\mathcal{V}_1^*$, we conclude that $\tilde{x}_1(0) = x_1(0)$, and in fact the subspace $(\mathcal{V}^*)^\perp$, can be reconstructed through the measurements $\tilde{y}_2 = y$. □

In Lemma 6.3.1 we show that the state x_1 of (6.30) can be reconstructed modulo $\mathcal{V}_1^*$. We now show that the state x_2 can generally not be completely reconstructed.

Lemma 6.3.2 (Reconstruction of the state x_2) *Let $\mathcal{V}_1^* = \operatorname{Im}(V_1)$ be the largest controlled invariant subspace of the system* (6.31). *The state x_2 of the system* (6.30) *can be reconstructed only modulo $\mathcal{V}_2^* = A_{22}^{-1} \operatorname{Im}\left(\left[A_{21}V_1 \; B_2\right]\right)$.*

Proof: Let $x_1 = \bar{x}_1 + \hat{x}_1$, where $\bar{x}_1 \in \mathcal{V}_1^*$ and $\hat{x}_1$ is orthogonal to $\mathcal{V}_1^*$. From Lemma 6.3.1, the signal $\hat{x}_1$ can be entirely reconstructed via y. Notice that

$$\begin{aligned} 0 &= A_{21}x_1(t) + A_{22}x_2(t) + B_2u(t), \\ &= A_{21}V_1v_1(t) + A_{21}\hat{x}_1(t) + A_{22}x_2(t) + B_2u(t). \end{aligned}$$

Let W be such that $\operatorname{Ker}(W) = \operatorname{Im}([A_{21}V_1 \; B_2])$. Then, $0 = WA_{21}\hat{x}_1 + WA_{22}x_2$, and hence $x_2 = \bar{x}_2 + \hat{x}_2$, where $\hat{x}_2 = (WA_{22})^{\dagger}WA_{21}\hat{x}_1$, and $\bar{x}_2 \in \operatorname{Ker}(WA_{22}) = A_{22}^{-1} \operatorname{Im}\left(\left[A_{21}V_1 \; B_2\right]\right)$. The statement follows. □

To conclude this part, we remark the following points. First, our characterization of $\mathcal{V}_1^*$ and $\mathcal{V}_2^*$ is equivalent to the definition of *weakly unobservable* subspace in [35], and of maximal *output-nulling* subspace in [54]. Hence, we proposed an optimal state estimator for our distributed attack identification procedure, and the matrix V_i in **(S1: estimation and communication)** can be computed as in [35, 54]. Second, a reconstruction of x_1 modulo $\mathcal{V}_1^*$ and x_2 modulo $\mathcal{V}_2^*$ can be obtained through standard algebraic techniques [6]. Third and finally, Lemma 6.3.1 and Lemma 6.3.2 extend the results in [8] by characterizing the subspaces of the state space that can be reconstructed with an algebraic method by processing the measurements y and their derivatives.

6.4 Illustrative Examples

6.4.1 An example of centralized detection and identification

In this section we apply our centralized attack detection and identification methods to the IEEE RTS96 power network [39] illustrated in Fig. 6.3. In particular, we first consider the nominal case, in which the power network dynamics evolve as nominal linear time-invariant descriptor system, as described in Section 3.1. Second, we consider the case of additive state and measurement noise, and we show the robustness of the attack detection and identification monitors. Third, we consider the case of nonlinear differential-algebraic power network dynamics and show the effectiveness of our methods in the presence of unmodeled nonlinear dynamics.

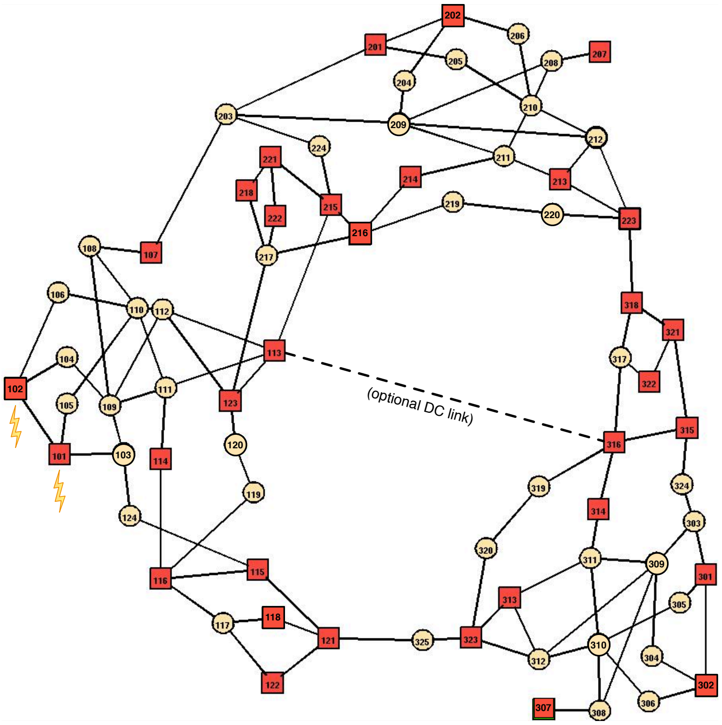


Figure 6.3: Diagram of the IEEE RTS96 power network [39]. The dynamics of the generators $\{101, 102\}$ are affected by an attacker.

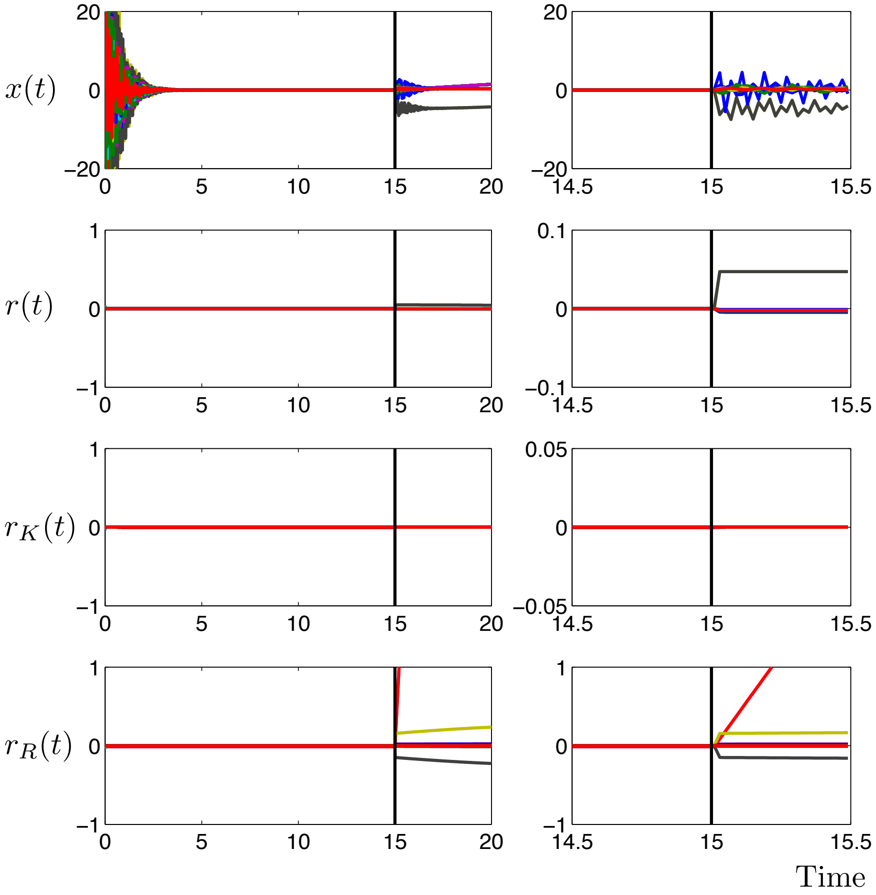


Figure 6.4: In this figure we report our simulation results for the case of linear network dynamics without noise and for the proposed detection monitor (6.1) and identification monitor (6.24), respectively. The state trajectory x consists of the generators angles and frequencies. The detection residual r becomes nonzero after time 15s, and it reveals the presence of the attack. The identification residual r_K is identically zero even after time 15s, and it reveals that the attack set is $K = \{101, 102\}$. The identification residual r_R is nonzero after time 15s, and it reveals that R is not the attack set.

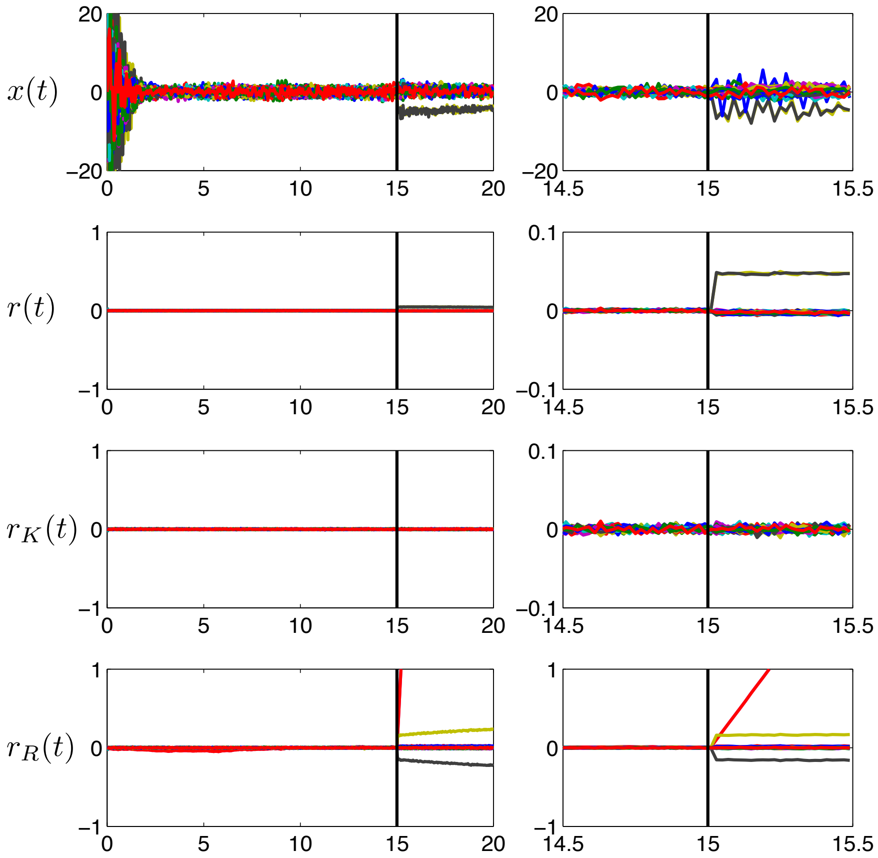


Figure 6.5: In this figure we report our simulation results for the case of linear network dynamics driven by state and measurements noise. For this case, we choose the output injection matrices of the detection and identification filters as the corresponding optimal Kalman gain (see Remark 7 and Remark 11). Due to the presence of noise, the residuals deviate from their nominal behavior reported in Fig. 6.4. Although the attack is clearly still detectable and identifiable, additional statistical tools such as hypothesis testing [7] may be adopted to analyze the residuals r, r_K, and r_R.

For our numerical studies, we assume the angles and frequencies of every generator to be measured. Additionally, we let the attacker affect the angles of the generators $\{101, 102\}$ with a random signal starting from time 15s. Since the considered power network dynamics are of index one, the filters are implemented using the nonsingular Kron-reduced system representation. The results of our simulations are in Fig. 6.4, Fig. 6.5, and Fig. 6.6. In conclusion, our centralized detection and identification filters appears robust to state and measurements noise and unmodeled dynamics.

6.4.2 An example of distributed detection

The IEEE 118 bus system shown in Fig. 5.1 represents a portion of the Midwestern American Electric Power System as of December 1962. This test case system is composed of 118 buses and 54 generators, and its parameters can be found, for example, in [122]. A linear continuous-time descriptor model of the network dynamics under attack assumes the form (4.1).

For estimation and attack detection purposes, we partition the IEEE 118 bus system into 5 disjoint areas, we assign a control center to each area, and we implement our detection procedure via the filter (6.12); see Fig. 5.3 for a graphical illustration. Suppose that each control center continuously measures the angle of the generators in its area, and suppose that an attacker compromises the measure-

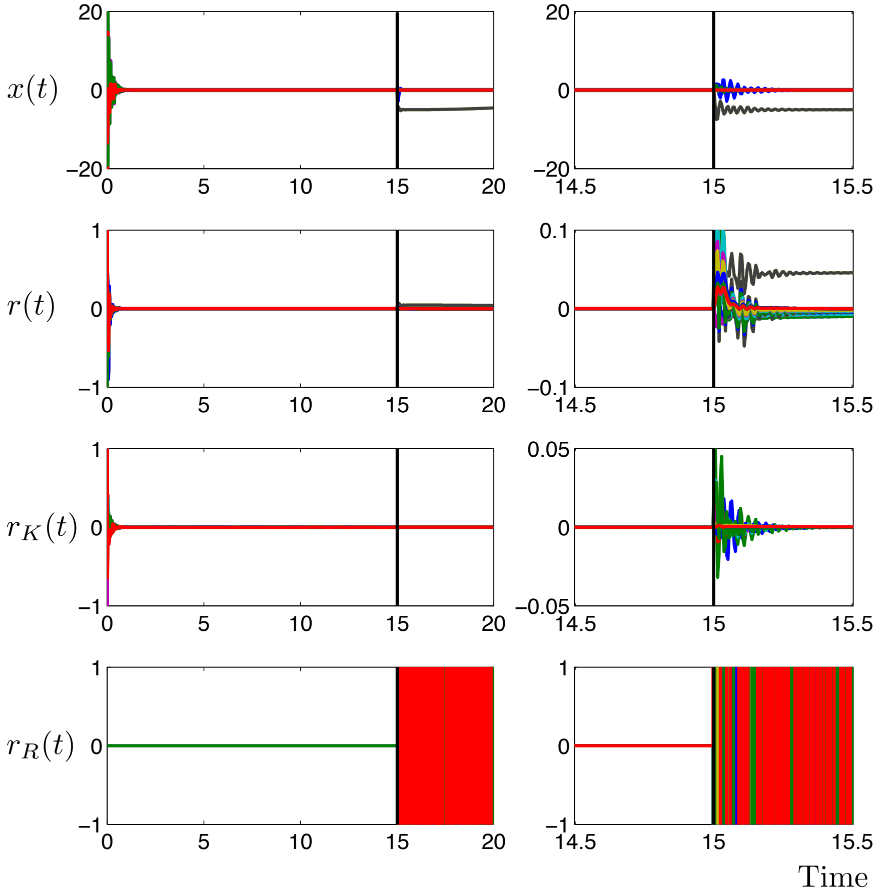


Figure 6.6: In this figure we report our simulation results for the case of nonlinear network dynamics without noise. For this case, the detection and identification filters are designed for the *nominal linearized dynamics* with output injection matrices as the corresponding optimal Kalman gain (see Remark 7 and Remark 11). Despite the presence of unmodeled nonlinear dynamics, the residuals reflect their nominal behavior reported in Fig. 6.4.

ments of all the generators of the first area. In particular, starting at time 30s, the attacker adds a signal u_K to all measurements in area 1. It can be verified that the attack set K is detectable. According to assumption (A3), the attack signal u_K needs to be continuous to guarantee a continuous state trajectory (since the power network is a descriptor system of index 1). In order to show the robustness of our detection filter (6.12), we let u_K be randomly distributed in the interval $[0, 0.5]$ rad.

The control centers implement the distributed attack detection procedure described in (6.12), with $G = AC^{\mathsf{T}}$. It can be verified that the pair $(E, A_D + GC)$ is Hurwitz stable, and that $\rho\,(j\omega E - A_D - GC)^{-1}A_C) < 1$ for all $\omega \in \mathbb{R}$. As predicted by Theorem 6.1.4, our distributed attack detection filter is convergent; see Fig. 6.7. For completeness, in Fig. 6.8 we illustrate the convergence of our waveform relaxation-based filter as a function of the number of iterations k. Notice that the number of iterations directly reflects the communication complexity of our detection scheme.

6.4.3 An example of distributed identification

Consider the sensor network in Fig. 6.9, where the state of the blue nodes $\{2, 5, 7, 12, 13, 15\}$ is measured and the state of the red node $\{3\}$ is corrupted by an attacker. Assume that the network evolves according to nonsingular, linear,

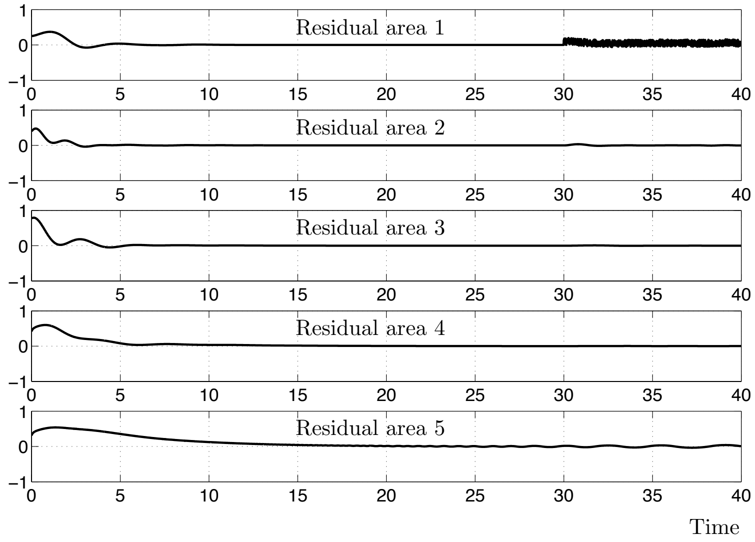


Figure 6.7: In this figure we show the residual functions computed through the distributed attack detection filter (6.12). The attacker compromises the measurements of all the generators in area 1 from time 30 with a signal uniformly distributed in the interval $[0, 0.5]$. The attack is correctly detected, because the residual functions do not decay to zero. For the simulation, we run $k = 100$ iterations of the attack detection method.

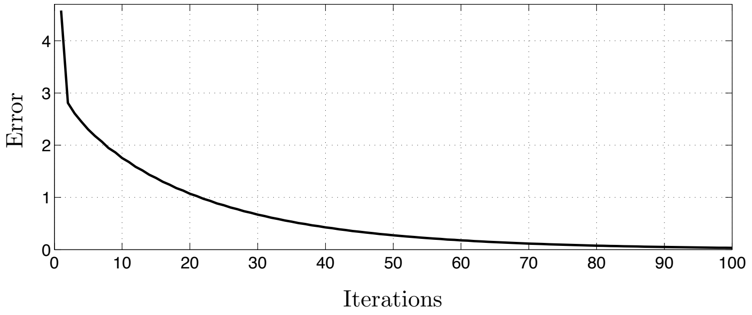


Figure 6.8: The plot represents the error of our waveform relaxation based filter (6.12) with respect to the corresponding decentralized filter (6.7). Here the error is $\max_{t\in[0,T]} \left\| w^{(k)}(t) - w(t) \right\|_\infty$, that is, the worst-case difference of the outputs of the two filters. As predicted by Theorem 6.1.4, the error is convergent.

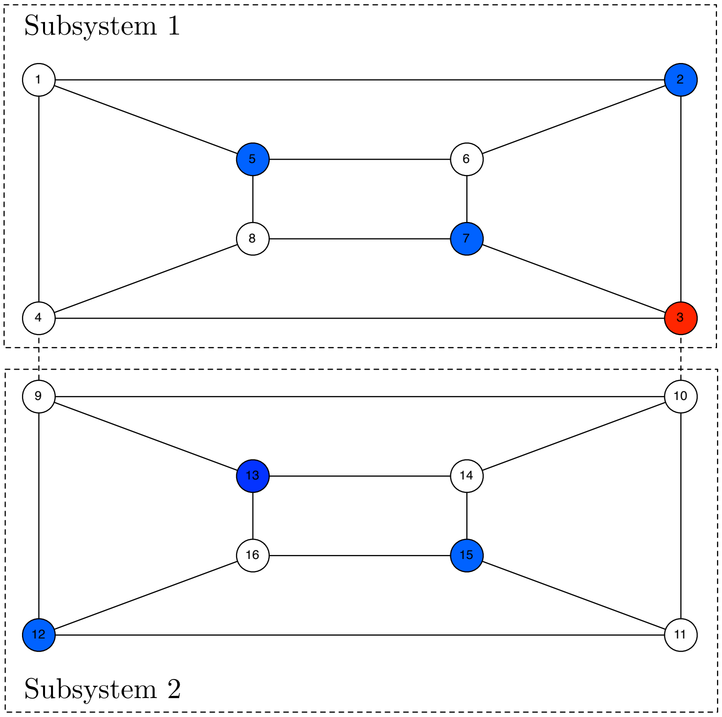


Figure 6.9: This figure shows a network composed of two subsystems. A control center is assigned to each subsystem. Each control center knows only the dynamics of its local subsystem. The state of the blue nodes $\{2, 5, 7, 12, 13, 15\}$ is continuously measured by the corresponding control center, and the state of the red node $\{3\}$ is corrupted by an attacker. The decoupled identification procedure presented in Subsection 6.2.3 fails at detecting the attack. Instead, by means of our cooperative identification procedure, the attack can be detected and identified via distributed computation.

time-invariant dynamics. Assume further that the network has been partitioned into the two areas $V_1 = \{1, \dots, 8\}$ and $V_2 = \{9, \dots, 16\}$ and at most one area is under attack. Since $\{3, 4\}$ are the boundary nodes for the first area, the attack set $K = 3$ is neither detectable nor identifiable by the two control centers via the fully decoupled procedure in Section 6.2.3.

Consider now the second subsystem with the boundary nodes $K_2^{\mathrm{b}} = \{9, 10\}$. It can be shown that, generically, the second subsystem with unknown input $B_2^{\mathrm{b}} f_2$ has no invariant zeros. Hence, the state of the second subsystem can be entirely reconstructed. Analogously, since the attack is on the boundary of the first subsystem, the state of the first subsystem can be reconstructed, so that the residual r_2 is identically zero; see Lemma 6.2.5.

Suppose that the state of the second subsystem is continuously transmitted to the control center of the first subsystem. Then, the only unknown input in the first subsystem is due to the attack, which is now generically detectable and identifiable, since the associated system has no invariant zeros; see Lemma 6.2.6. We conclude that our cooperative identification procedure outperforms the decoupled counterpart in Section 6.2.3.

Chapter 7

Synthesis of Attacks

In this chapter we propose a geometric-based technique to design undetectable and unidentifiable attacks. Differently from the existing literature, where specific attacks have been designed [2, 40, 69], we provide a general characterization of *all* undetectable and unidentifiable attacks. We focus on attacks that can be cast without knowledge of the system state, and, for the ease of notation, we only consider non-descriptor systems.

7.1 Problem Setup

Consider the system

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

where $x \in \mathbb{R}^n$, $u \in \mathbb{R}^m$, $y \in \mathbb{R}^p$, and A, B, C, and D are constant matrices of appropriate dimension. The signal u represent a known control input, while the signal y is a measurement output. Consider the case of an attacker able to affect the evolution of some states and measurements. In particular, let the dynamics under attack be

$$\begin{aligned} \dot{x}(t) &= Ax + Bu(t) + Ew(t), \\ y(t) &= Cx(t) + Du(t) + Hw(t), \end{aligned} \tag{7.2}$$

where $w : [0, \infty) \to \mathbb{R}^k$ is the attack signal, and E, H are constant matrices of appropriate dimension. The objective is for the attacker to inject a nonzero signal w to compromise the system dynamics while avoiding detection or identification by a dynamic monitor (see Chapter 4). We remark that the attack design described in the following section for continuous time system is also applicable to discrete time systems.

7.2 Design of Undetectable and Unidentifiable Attacks

In this section we design undetectable and unidentifiable attacks. We start from undetectable attacks.

Theorem 7.2.1 ***(Design of undetectable malicious attacks)*** *Consider the system under attack (7.2). Let $\mathcal{V}^* \subseteq \mathbb{R}^n$ be the largest subspace satisfying*

$$A\mathcal{V}^* \subseteq \mathcal{V}^* + \mathrm{Im}(E), \text{ and } \mathrm{Ker}(C)\mathcal{V}^* \subseteq \mathrm{Im}(H),$$

and let F be such that

$$(A + EF)\mathcal{V}^* \subseteq \mathcal{V}^*, \text{ and } \mathcal{V}^* \subseteq \mathrm{Ker}(C + HF).$$

Let $\bar{E} = \mathrm{Basis}(\mathcal{V}^* \cap \mathrm{Im}(E))$, and let $\mathcal{S}^*$ be the smallest subspace of the state space satisfying

$$A(\mathcal{S}^* \cap \mathrm{Ker}(C)) \subseteq \mathcal{S}^*, \text{ (and) } \mathrm{Ker}(H) \subseteq E^{-1}\mathcal{S}^*.$$

Then,

(i) for every input v of appropriate dimension, the attack $w = Fx + \bar{E}^\dagger v$ is undetectable;

(ii) the subspace $\mathcal{V}^* \cap \mathcal{S}^*$ denotes the set of states reachable by the attacker while remaining undetected; and

(iii) any state in $\mathcal{V}^* \cap \mathcal{S}^*$ can be reached with an input $w = Fx + \bar{E}^\dagger v$.

Proof: Define the nonsingular transformation matrix $T = [T_1, T_2, T_3]$, with $T_1 = \mathrm{Basis}(\mathcal{V}^* \cap \mathcal{S}^*)$, $T_2 = \mathrm{Basis}(\mathcal{V}^*)$, and T_3 such that T is nonsingular. In the new

$z = T^{-1}x$ coordinates, the system matrices are

$$T^{-1}(A+BF)T = \begin{bmatrix} A_{11} & A_{12} & A_{13} \\ 0 & A_{22} & A_{23} \\ 0 & 0 & A_{33} \end{bmatrix}, \quad T^{-1}E = \begin{bmatrix} E_1 \\ 0 \\ E_3 \end{bmatrix},$$

$$(C+HF)T = \begin{bmatrix} 0 & 0 & C_3 \end{bmatrix}, \quad \text{Basis}(T^{-1}E\bar{B}^{\dagger}) = \begin{bmatrix} E_1^{\mathsf{T}} & 0 & 0 \end{bmatrix}^{\mathsf{T}}, \tag{7.3}$$

where the zero pattern is due to the invariance properties of $\mathcal{V}^*$ and $\mathcal{S}^*$. As a consequence of the above decomposition, any input $u = Fx + \bar{E}^{\dagger}v$ does not affect the output, and therefore it is undetectable. Statements (ii) and (iii) are a direct consequence of the above decomposition; see [6]. □

The following remarks are in order. First, Theorem 7.2.1 characterizes the states reachable by an attacker with input matrices (E, H). If a specific desired state should be contained within this reachable set, then the attack matrices should be selected accordingly. We leave this interesting aspect of coordinated attack design as the subject of future research. Second, the inputs w in Theorem 7.2.1 correspond to the attacks that can be cast by the attacker independently of the system state while remaining undetectable; see Theorem 4.2.5 and the notions of left-invertibility [6]. Third, as a consequence of Theorem 4.3.3, if $H = 0$ and $\text{Rank}(E) > \text{Rank}(C)$, then there exist attacks as in Theorem 7.2.1. Finally, the input v can be designed as to optimize some performance function, such as, for instance, the effect of the malicious control on the sacrificial machines, the energy of the malicious control, or the information pattern required to implement the malicious control. A relative example is in Section 7.3.

We now focus on the design of unidentifiable attacks.

Theorem 7.2.2 ***(Design of unidentifiable malicious attacks)*** *Consider the system under attack (7.2). Let* $M \in \mathbb{R}^{n \times k}$*, where* $k = \mathit{Rank}(E)$ *and* $\operatorname{Im}(M) \cap \operatorname{Im}(E) = \{0\}$*. Let* $\mathcal{V}^* \subseteq \mathbb{R}^n$ *be the largest subspace satisfying*

$$A\mathcal{V}^* \subseteq \mathcal{V}^* + \operatorname{Im}([E\ M]), \text{ and } \operatorname{Ker}(C)\mathcal{V}^* \subseteq \operatorname{Im}([H\ N]),$$

and let $F = [F_1^\mathsf{T}\ F_2^\mathsf{T}]^\mathsf{T}$ *be such that*

$$(A + EF_1 + MF_2)\mathcal{V}^* \subseteq \mathcal{V}^*, \text{ and } \mathcal{V}^* \subseteq \operatorname{Ker}(C + HF_1 + NF_2).$$

Let $\bar{E} = \operatorname{Basis}(\mathcal{V}^* \cap \operatorname{Im}(E))$*, and let* $\mathcal{S}^*$ *be the smallest subspace of the state space satisfying*

$$A(\mathcal{S}^* \cap \operatorname{Ker}(C)) \subseteq \mathcal{S}^*, \text{ (and) } \operatorname{Ker}(H) \subseteq E^{-1}\mathcal{S}^*.$$

Then,

(i) for every input v*, the attack* $w = F_1 x + \bar{E}^\dagger v$ *is unidentifiable;*

(ii) the subspace $\mathcal{V}^* \cap \mathcal{S}^*$ *denotes the set of states reachable by the attacker while remaining unidentified; and*

(iii) any state in $\mathcal{V}^* \cap \mathcal{S}^*$ can be reached with an input $w = F_1 x + \bar{E}^\dagger v$.

Proof: From Theorem 7.2.1, the attack signal $\hat{w} = EF_1 x + MF_2 x + \hat{E}^\dagger \hat{v}$, where $\hat{E} = \text{Basis}\left(\mathcal{V}^* \cap \text{Im}\left(\begin{bmatrix} E & M \end{bmatrix}\right)\right)$, is undetectable from the output for any signal $\hat{v}$. Then, the output $y(0, w, t)$ generated by the input w from the zero state is identically zero. Due to linearity of (7.2), it follows $y(0, EF_1 x + \hat{E}^\dagger \hat{v}, t) = -y(0, MF_2 x, t)$ at all times t. Hence the attacker with input matrices (E, H) and signal $w = EF_1 x + \hat{E}^\dagger \hat{v}$ is unidentifiable from an attacker with input matrices (M, N) and signal $w = -EF_2 x$, independently of $\hat{v}$. To conclude the proof, notice that $\text{Im}(\bar{E}) \subseteq \text{Im}(\hat{E})$, and that statements (ii) and (iii) follow from Theorem 7.2.1. □

7.3 An Illustrative Example

Motivated by [26], in this section we study malicious attacks in a competitive power generation environment. Consider a connected power transmission network with n generators $G_{\text{m}} = \{g_1, \dots, g_n\}$, where the rotor dynamics of each generator are modeled by second-order linear swing equations subject to governor control, and the power flows along lines are modeled by the DC approximation. Assume that a subset $K = \{k_1, \dots, k_m\}$ of m generators is driven by an additional control action besides the primary frequency control. After elimination of the load bus variables through Kron reduction, the power network dynamics subject to the additional control u at the generators K read as (see Chapter 3)

$$\dot{x}(t) = Ax(t) + Bu(t), \tag{7.4}$$

where $x = [\theta^\mathsf{T}, \omega^\mathsf{T}]^\mathsf{T}$ contains the generator rotor angles and frequencies at time t, $A \in \mathbb{R}^{2n\times 2n}$, $C \in \mathbb{R}^{m\times 2n}$, and $B = I_K \in \mathbb{R}^{2n\times m}$, where $I_K = [e_{n+k_1} \cdots e_{n+k_m}]$ with e_i being the i-th canonical vector in $\mathbb{R}^{2n}$.

In [26] the following competitive scenario is considered: the group of generators K form a coalition, one sacrificial machine $\bar{k} \in K$ is selected in the coalition, and a specific coordinated control strategy is proposed for the generators K to destabilize the other machines $G_\mathrm{m} \setminus K$, while maintaining satisfactory performance within the group $K \setminus \{\bar{k}\}$. It can be shown that the result in [26] is a special case of Theorem 7.2.1, since a destabilizing state feedback can be obtained by properly choosing v.

For illustration purposes, consider an aggregated model of the Western North American power grid as illustrated in Fig. 7.1. This model is often studied [113] in the context of wide-area oscillations. Assume that the generators $\{1, 9\}$ are being controlled, and that generator 9 is the sacrificial machine. Following Theorem 7.2.1, a malicious attack $u = Fx + \bar{B}^\dagger v$ is cast by the generators $\{1, 9\}$ such that generator 1 is not affected by the attack. Additionally, the input v is optimally chosen such that generator 2 maintains an acceptable working condition even in the presence of the attack, and large frequency deviations are induced at all

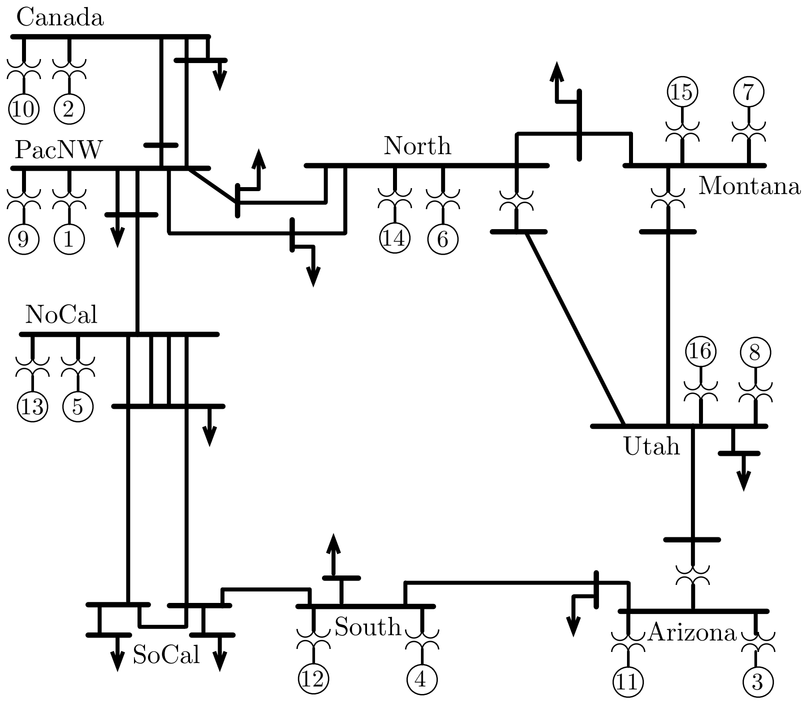


Figure 7.1: A schematic diagram of the Western North American power grid.

other generators $G_{\mathrm{m}} \setminus K$. As a consequence, the linear model (7.4) is driven far away from the operating point, and the corresponding original nonlinear model eventually loses synchrony. In a real-world scenario the affected generators $G_{\mathrm{m}} \setminus K$ would be disconnected for safety reasons.

In the above scenario, assume that each generator monitors its own state variables, and that at most two generators may be colluding to disrupt the network. Notice that detectability of the malicious attacks designed in Theorem 7.2.1 is guaranteed for each generator affected by the attack. Unfortunately, no generator can identify the colluding generators while relying only on its own measurements.

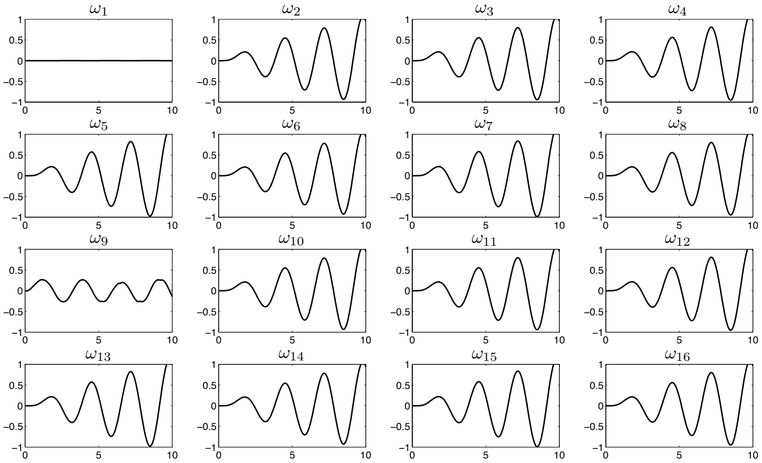


Figure 7.2: The deviations of the generators frequencies from their steady state value induced by a malicious attack is here reported. The attack is designed by using the result in Theorem 7.2.1. In particular, the input v is chosen such that the infinity norm of ω_9 is minimized, subject to the infinity norm of ω_{16} being no less than 1. Notice that generator 1 is not affected by the attack, and that generator 9 maintains satisfactory performance. Instead, the other generators are severely affected by the coordinated attack.

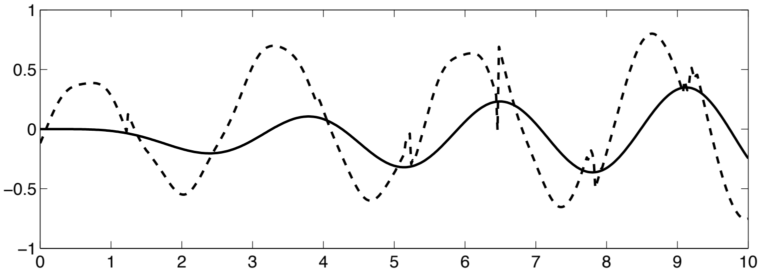


Figure 7.3: This figures shows the governor control input injected by generator 1 (solid) and by generator 9 (dashed). Both plots are in p.u. values and for the linear system (7.4), that is, measured as deviation from the steady state.

To see this, let B_K be the input matrix associated with any set K of two generators, and let $C_i = e_i^{\mathsf{T}}$ be the output matrix associated with generator i. It can be verified that for every K and i the system (A, B_K, C_i) is right-invertible [6]. Hence, no generator alone can identify the malicious generators, and a coalition of multiple sensors becomes necessary.

Chapter 8

Consensus Computation with Misbehaving Agents

In this Chapter we specify the previously presented results on security to the important case of linear consensus algorithms with misbehaving agents. We will consider discrete-time linear consensus algorithms, for which we will first characterize the resilience to Byzantine attacks as a function of the connectivity of the underlying network. Then, we will exploit a notion of network decentralization to develop an efficient decentralized detection and identification algorithm.

8.1 Problem Setup

Consider the consensus system (3.7). We allow for some agents to update their state differently than specified by the matrix A by adding an exogenous input to the consensus system. Let u_i, $i \in V$, be the input associated with the

i-th agent, and let u be the vector of the functions u_i. The consensus system becomes $x(t+1) = Ax(t) + u(t)$.

Definition 6 (Misbehaving agent) *An agent j is* misbehaving *if there exists a time $t \in \mathbb{N}$ such that $u_j(t) \neq 0$.*

In Section 8.2 we will give a precise definition of the distinction, made already in the Introduction, between *faulty* and *malicious* agents on the basis of their inputs.

Let $K = \{i_1, i_2, \dots\} \subseteq V$ denote a set of misbehaving agents, and let $B_K = \begin{bmatrix} e_{i_1} & e_{i_2} & \cdots \end{bmatrix}$, where e_i is the i-th vector of the canonical basis. The consensus system with misbehaving agents K reads as

$$x(t+1) = Ax(t) + B_K u_K(t). \tag{8.1}$$

As it is shown in [76], algorithms of the form (3.7) have no resilience to malfunctions, and the presence of a misbehaving agent may prevent the entire network from reaching consensus. As an example, let $c \in \mathbb{R}$, and let $u_i = -A_i x + c$, being A_i the i-th row of A. After reordering the variables in a way that the well-behaving nodes come first, the consensus system can be rewritten as

$$\tilde{x}(t+1) = \begin{bmatrix} Q & R \\ 0 & 1 \end{bmatrix} \tilde{x}(t), \tag{8.2}$$

where the matrix Q corresponds to the interaction among the nodes $V \setminus \{i\}$, while R denotes the connection between the sets $V \setminus \{i\}$ and $\{i\}$. Recall that a matrix is said to be Schur stable if all its eigenvalues lie in the open unit disk.

Lemma 8.1.1 (Quasi-stochastic submatrices) *Let A be an $n \times n$ consensus matrix, and let J be a proper subset of $\{1, \dots, n\}$. The submatrix with entries $A_{i,k}$, $i, k \in J$, is Schur stable.*

Proof: Reorder the nodes such that the indexes in J come first in the matrix A. Let A_J be the leading principal submatrix of dimension $|J|$. Let $\tilde{A}_J = \left[\begin{smallmatrix} A_J & 0 \\ 0 & 0 \end{smallmatrix}\right]$, where the zeros are such that $\tilde{A}_J$ is $n \times n$, and note that $\rho(A_J) = \rho(\tilde{A}_J)$, where $\rho(A_J)$ denotes the spectral radius of the matrix A_J [65]. Since A is a consensus matrix, it has only one eigenvalue of unitary modulus, and $\rho(A) = 1$. Moreover, $A \geq |\tilde{A}_J|$, and $A \neq |\tilde{A}_J|$, where $|\tilde{A}_J|$ is such that its (i,j)-th entry equals the absolute value of the (i,j)-th entry of $\tilde{A}_J$, $\forall i, j$. It is known that $\rho(A_J) \leq \rho(A) = 1$, and that if equality holds, then there exists a diagonal matrix D with nonzero diagonal entries, such that $A = D\tilde{A}_J D^{-1}$ [65, Wielandt's Theorem]. Because A is irreducible, there exists no diagonal D with nonzero diagonal entries such that $A = D\tilde{A}_J D^{-1}$ and the statement follows. □

Because of Lemma 8.1.1, the matrix Q in (8.2) is Schur stable, so that the steady state value of the well-behaving agents in (8.2) depends upon the action of the misbehaving node, and it corresponds to $(I - Q)^{-1}Rc$. In particular, since $(I-Q)^{-1}R = [1 \, \cdots \, 1]^\mathsf{T}$, a single misbehaving agent can steer the network towards any consensus value by choosing the constant c.[1]

It should be noticed that a different model for the misbehaving nodes consists in the modification of the entries of A corresponding to their incoming communication edges. However, since the resulting network evolution can be obtained by properly choosing the input u_K and letting the matrix A fixed, our model does not limit generality, while being convenient for the analysis. For the same reason, system (8.1) also models the case of defective communication edges. Indeed, if the edge from the node i to the node j is defective, then the message received by the agent j at time t is incorrect, and hence also the state $x_j(\bar{t})$, $\bar{t} \geq t$. Since the values $x_j(\bar{t})$ can be produced with an input $u_j(t)$, the failure of the edge (i, j) can be regarded as the j-th misbehaving action. Finally, the following key difference between our model and the setup in [30] should be noticed. If the communication graph is complete, then up to $n - 1$ (instead of $\lfloor n/3 \rfloor$) misbehaving agents can be identified in our model by a well-behaving agent. Indeed, since with a complete communication graph the initial state $x(0)$ is correctly received by every

[1] If the misbehaving input is not constant, then the network may not achieve consensus. In particular, the effect of a misbehaving input u_K on the network state at time t is given by $\sum_{\tau=0}^{t} A^{t-\tau} B_K u_K(\tau)$ (see also Section 8.3).

node, the consensus value is computed after one communication round, so that the misbehaving agents cannot influence the dynamics of the network.

8.2 Detection and Identification of Misbehaving Agents

The problem of ensuring trustworthy computation among the agents of a network can be divided into a detection phase, in which the presence of misbehaving agents is revealed, and an identification phase, in which the identity of the intruders is discovered. A set of misbehaving agents may remain undetected from the observations of a node j if there exists a normal operating condition under which the node would receive the same information as under the perturbation due to the misbehavior. To be more precise, let $C_j = [e_{n_1} \ \dots \ e_{n_p}]^{\mathsf{T}}$, $\{n_1, \dots, n_p\} = N_j$, denote the output matrix associated with the agent j, and let $y_j = C_j x$ denote the measurements vector of the j-th agent at time t. Let $x(x_0, \bar{u}, t)$ denote the network state trajectory generated from the initial state x_0 under the input sequence $\bar{u}$, and let $y_j(x_0, \bar{u}, t)$ be the sequence measured by the j-th node and corresponding to the same initial condition and input.

Definition 7 (Undetectable input) *For a linear consensus system of the form* (8.1)*, the input u_K introduced by a set K of misbehaving agents is* undetectable *if*

$$\exists\, x_1, x_2 \in \mathbb{R}^n, j \in V : \forall t \in \mathbb{N}, y_j(x_1, u_K, t) = y_j(x_2, 0, t).$$

A more general concern than detection is identifiability of intruders, i.e. the possibility to distinguish from measurements between the misbehaviors of two distinct agents, or, more generally, between two disjoint subsets of agents. Let $\mathcal{K} \subset 2^V$ contain all possible sets of misbehaving agents.[2]

Definition 8 (Unidentifiable input) *For a linear consensus system of the form* (8.1) *and a nonempty set $K_1 \in \mathcal{K}$, an input u_{K_1} is* unidentifiable *if there exist $K_2 \in \mathcal{K}$, with $K_1 \neq K_2$, and an input u_{K_2} such that*

$$\exists\, x_1, x_2 \in \mathbb{R}^n, j \in V : \forall t \in \mathbb{N}, y_j(x_1, u_{K_1}, t) = y_j(x_2, u_{K_2}, t).$$

Of course, an undetectable input is also unidentifiable, since it cannot be distinguished from the zero input. The converse does not hold. Unidentifiable inputs are a very specific class of inputs, to be precisely characterized later in this section. Correspondingly, we define

Definition 9 (Malicious behaviors) *A set of misbehaving agents K is malicious if its input u_K is unidentifiable. It is faulty otherwise.*

[2] An element of $\mathcal{K}$ is a subset of $\{1, \ldots, n\}$. For instance, $\mathcal{K}$ may contain all the subsets of $\{1, \ldots, n\}$ with a specific cardinality.

We provide now a characterization of malicious behaviors for the particularly important class of linear consensus networks. Notice however that, if the matrix A below is not restricted to be a consensus matrix, then the following Theorem extends the results in [107] by fully characterizing the inputs for which a group of misbehaving agents remains unidentified from the output observations of a certain node.

Theorem 8.2.1 (Characterization of malicious behaviors) *For a linear consensus system of the form* (8.1) *and a nonempty set $K_1 \in \mathcal{K}$, an input u_{K_1} is unidentifiable if and only if*

$$C_j A^{t+1} \bar{x} = \sum_{\tau=0}^{t} C_j A^{t-\tau} \left(B_{K_1} u_{K_1}(\tau) - B_{K_2} u_{K_2}(\tau) \right),$$

for all $t \in \mathbb{N}$, and for some u_{K_2}, with $K_2 \in \mathcal{K}$, $K_1 \neq K_2$, and $\bar{x} \in \mathbb{R}^n$. If the same holds with $u_{K_2} \equiv 0$, the input is actually undetectable.

Proof: By definitions 7 and 8, an input u_{K_1} is unidentifiable if $y_j(x_1, u_{K_1}, t) = y_j(x_2, u_{K_2}, t)$, and it is undetectable if $y_j(x_1, u_{K_1}, t) = y_j(x_2, 0, t)$, for some x_1, x_2, and u_{K_2}. Due to linearity of the network, the statement follows. □

Remark 12 (Malicious behaviors are not generic) *Because an unidentifiable input must satisfy the equation in Theorem 8.2.1, excluding pathological cases, unidentifiable signals are not generic, and they can be injected only intentionally* *by colluding misbehaving agents. This motivates our definition of "malicious" for those agents which use unidentifiable inputs.* □

We consider now the resilience of a consensus network to faulty and malicious misbehaviors. Let I denote the identity matrix of appropriate dimensions. The zero dynamics of the linear system (A, B_K, C_j) are the (nontrivial) state trajectories invisible at the output, and can be characterized by means of the $(n+p) \times (n+|K|)$ pencil

$$P(z) = \begin{bmatrix} zI - A & B_K \\ C_j & 0 \end{bmatrix}.$$

The complex value $\bar{z}$ is said to be an invariant zero of the system (A, B_K, C_j) if there exists a state-zero direction x_0, $x_0 \neq 0$, and an input-zero direction g, such that $(\bar{z}I - A)x_0 + B_K g = 0$, and $C_j x_0 = 0$. Also, if $\text{rank}(P(z)) = n + |K|$ for all but finitely many complex values z, then the system (A, B_K, C_j) is left-invertible, i.e., starting from any initial condition, there are no two distinct inputs that give rise to the same output sequence [112]. We next characterize the relationship between the zero dynamics of a consensus system and the connectivity of the consensus graph.

Lemma 8.2.2 (Zero dynamics and connectivity) *Given a k-connected linear network with matrix A, there exists a set of agents K_1, with $|K_1| > k$, and a node*

j such that the consensus system (A, B_{K_1}, C_j) is not left-invertible. Furthermore, there exists a set of agents K_2, with $|K_2| = k$, and a node j such that the system (A, B_{K_2}, C_j) has nontrivial zero dynamics.

Proof: Let G be the digraph associated with A, and let k be the connectivity of G. Take a set K of $k+1$ misbehaving nodes, such that k of them form a vertex cut S of G. Note that, since the connectivity of G is k, such a set always exists. The network G is divided into two subnetworks G_1 and G_3, which communicate only through the nodes S. Assume that the misbehaving agent $K \setminus S$ belongs to G_3, while the observing node j belongs to G_1. After reordering the nodes such that the vertices of G_1 come first, the vertices S come second, and the vertices of G_3 come third, the consensus matrix A is of the form $\begin{bmatrix} A_{11} & A_{12} & 0 \\ A_{21} & A_{22} & A_{23} \\ 0 & A_{32} & A_{33} \end{bmatrix}$, where the zero matrices are due to the fact that S is a vertex cut. Let $u_S = -A_{23}x_3$, where x_3 is the vector containing the values of the nodes of G_3, and let $u_{K \setminus S}$ be any arbitrary nonzero function. Clearly, starting from the zero state, the values of the nodes of G_1 are constantly 0, while the subnetwork G_3 is driven by the misbehaving agent $K \setminus S$. We conclude that the triple (A, B_K, C_j) is not left-invertible.

Suppose now that $K \equiv S$ as previously defined, and let $u_K = -A_{23}x_3$. Let the initial condition of the nodes of G_1 and of S be zero. Since every state trajectory generated by $x_3 \neq 0$ does not appear in the output of the agent j, the triple (A, B_K, C_j) has nontrivial zero dynamics. □

Following Lemma 8.2.2, we next state an upper bound on the number of misbehaving agents that can be detected.

Theorem 8.2.3 (Detection bound) *Given a k-connected linear consensus network, there exist undetectable inputs for a specific set of k misbehaving agents.*

Proof: Let K , with $|K| = k$, be the misbehaving set, and let K form a vertex cut of the consensus network. Because of Lemma 8.2.2, for some output matrix C_j, the consensus system has nontrivial zero dynamics, i.e., there exists an initial condition $x(0)$ and an input u_K such that $y_j(t) = 0$ at all times. Hence, the input u_K is undetectable from the observations of j. □

We now consider the identification problem.

Theorem 8.2.4 (Identification of misbehaving agents) *For a set of misbehaving agents $K_1 \in \mathcal{K}$, every input is identifiable from j if and only if the consensus system $(A, [B_{K_1}\ \ B_{K_2}], C_j)$ has no zero dynamics for every $K_2 \in \mathcal{K}$.*

Proof: *(Only if)* By contradiction, let x_0 and $[u_{K_1}^\mathsf{T}\ \ -u_{K_2}^\mathsf{T}]^\mathsf{T}$ be a state-zero direction, and an input-zero sequence for the system $(A, [B_{K_1}\ \ B_{K_2}], C_j)$. We have

$$\begin{aligned} y_j(t) &= 0 \\ &= C_j\Big(A^t x_0 + \sum_{\tau=0}^{t-1} A^{t-\tau-1}\big(B_{K_1}u_{K_1}(\tau) - B_{K_2}u_{K_2}(\tau)\big)\Big). \end{aligned}$$

Therefore,

$$C_j\left(A^t x_0^1 + \sum_{\tau=0}^{t-1} A^{t-\tau-1} B_{K_1} u_{K_1}(\tau)\right) = C_j\left(A^t x_0^2 + \sum_{\tau=0}^{t-1} A^{t-\tau-1} B_{K_2} u_{K_2}(\tau)\right),$$

where $x_0^1 - x_0^2 = x_0$. Clearly, since the output sequence generated by K_1 coincide with the output sequence generated by K_2, the two inputs are unidentifiable.

(If) Suppose that, for any $K_2 \in \mathcal{K}$, the system $(A[B_{K_1}\ \ B_{K_2}])$ has no zero dynamics, i.e., there exists no initial condition x_0 and input $[u_{K_1}^\mathsf{T}\ u_{K_2}^\mathsf{T}]^\mathsf{T}$ that result in the output being zero at all times. By the linearity of the network, every input u_{K_1} is identifiable. □

As a consequence of Theorem 8.2.4, if up to k misbehaving agents are allowed to act in the network, then a necessary and sufficient condition to correctly identify the set of misbehaving nodes is that the consensus system subject to any set of $2k$ inputs has no nontrivial zero dynamics.

Theorem 8.2.5 (Identification bound) *Given a k-connected linear consensus network, there exist unidentifiable inputs for a specific set of $\lfloor \frac{k-1}{2} \rfloor + 1$ misbehaving agents.*

Proof: Since $2(\lfloor \frac{k-1}{2} \rfloor + 1) \geq k$, by Lemma 8.2.2 there exist K_1, K_2, with $|K_1| = |K_2| = \lfloor \frac{k-1}{2} \rfloor + 1$, and j such that the system $(A, [B_{K_1}\ \ B_{K_2}], C_j)$ has nontrivial zero dynamics. By Theorem 8.2.4, there exists an input and an initial condition such that K_1 is undistinguishable from K_2 to the agent j. □

In other words, in a k-connected network, at most $k-1$ (resp. $\lfloor \frac{k-1}{2} \rfloor$) misbehaving agents can be certainly detected (resp. identified) by every agent. Notice that, for a linear consensus network, Theorem 8.2.5 provides an alternative proof of the resilience bound presented in [30] and in [107].

We now focus on the faulty misbehavior case. Notice that, because such agents inject only identifiable inputs by definition, we only need to guarantee the existence of such inputs. We start by showing that, independent of the cardinality of a set K, there exist detectable inputs for a consensus system (A, B_K, C_j), so that any set of faulty agents is detectable. By using a result from [111], an input u_K is undetectable from the measurements of the j-th agent only if for all $t \in \mathbb{N}$, it holds $C_j A^v B_K u_K(t) = C_j A^{v+1} x(t)$, where $C_j A^v B_K$ is the first nonzero Markov parameter, and $x(t)$ is the network state at time t. Notice that, because of the irreducibility assumption of a consensus matrix, independently of the cardinality of the faulty set and of the observing node j, there exists a finite v such that $C_j A^v B_K \neq 0$, so that every input $u_K \neq (C_j A^v B_K)^\dagger C_j A^{v+1} x$ is detectable. We show that, if the number of misbehaving components is allowed to equal the connectivity of the consensus network, then there exists a set of misbehaving agents that are unidentifiable independent of their input.

Theorem 8.2.6 (Identification of faulty agents) *Given a k-connected linear consensus network, there exists no identifiable input for a specific set of k misbehaving agents*

Proof: Let K_1, with $|K_1| = k$, form a vertex cut. The network is divided into two subnetworks G_1 and G_2 by the agents K_1. Let K_2, with $|K_2| \leq k$, be the set of faulty agents, and suppose that the set K_2 belongs to the subnetwork G_2. Let j be an agent of G_1. Notice that, because K_1 forms a vertex cut, for every initial condition $x(0)$ and for every input u_{K_2}, there exists an input u_{K_1} such that the output sequences at the node j coincide. In other words, every input u_{K_2} is unidentifiable. □

Hence, in a k-connected network, a set of k faulty agents may remain unidentified independent of its input function. It should be noticed that Theorems 8.2.5 and 8.2.6 only give an upper bound on the maximum number of concurrent misbehaving agents that can be detected and identified. In Section 8.4 it will be shown that, generically, in a k-connected network, there exists only identifiable inputs for any set of $\lfloor \frac{k-1}{2} \rfloor$ misbehaving agents, and that there exist some identifiable inputs for any set of $k-1$ misbehaving agents. In other words, if there exists a set of misbehaving nodes that cannot be identified by an agent, then, provided that the connectivity of the communication graph is sufficiently high, a random and arbitrarily small change of the consensus matrix makes the misbehaving nodes detectable and identifiable with probability one.

8.3 Effects of Unidentified Misbehaving Agents

In the previous section, the importance of zero dynamics in the misbehavior detection and identification problem has been shown. In particular, we proved that a misbehaving agent may alter the nominal network behavior while remaining undetected by injecting an input-zero associated with the current network state. We now study the effect of an unidentifiable attack on the final consensus value. As a preliminary result, we prove the detectability of a consensus network.

Lemma 8.3.1 (Detectability) *Let the matrix A be row stochastic and irreducible. For any network node j, the pair (A, C_j) is detectable.*

Proof: If A is stochastic and irreducible, then it has at least $h \geq 1$ eigenvalues of unitary modulus. Precisely, the spectrum of A contains $\{1 = e^{i\theta_0}, e^{i\theta_1}, \dots, e^{i\theta_{h-1}}\}$. By Wielandt's theorem [65], we have $AD_k = e^{i\theta_k} D_k A$, where $k \in \{0, \dots, h-1\}$, and D_k is a full rank diagonal matrix. By multiplying both sides of the equality by the vector of all ones, we have $AD_k\mathbf{1} = e^{i\theta_k} D_k A\mathbf{1} = e^{i\theta_k} D_k\mathbf{1}$, so that $D_k\mathbf{1}$ is the eigenvector associated with the eigenvalue $e^{i\theta_k}$. Observe that the vector $D_k\mathbf{1}$ has no zero component, and that, by the eigenvector test [112], the pair (A, C_j) is detectable. Indeed, since A is irreducible, the neighbor set N_j is nonempty, and the eigenvector $D_k\mathbf{1}$, with $k \in \{0, \dots, h-1\}$, is not contained in $\mathrm{Ker}(C_j)$. □

Observe that the primitivity of the network matrix is not assumed Lemma 8.3.1. By duality, a result on the stabilizability of the pair (A, B_j) can also be asserted.

Lemma 8.3.2 (Stabilizability) *Let the matrix A be row stochastic and irreducible. For any network node j, the pair (A, B_j) is stabilizable.*

Remark 13 (State estimation via local computation) *If a linear system is detectable (resp. stabilizable), then a linear observer (resp. controller) exists to asymptotically estimate (resp. stabilize) the system state. By combining the above results with Lemma 8.1.1, we have that, under a mild assumption on the matrix A, the state of a linear network can be asymptotically observed (resp. stabilized) via local computation. Consider for instance the problem of designing an observer [6], and let $C_j = e_j^\mathsf{T}$. Take $G = -A_j$, where A_j denotes the j-th column of A. Notice that the matrix $A+GC_j$ can be written as a block-triangular matrix, and it is stable because of Lemma 8.1.1. Finally, since the nonzero entries of G correspond to the out-neighbors[3] of the node j, the output injection operation GC_j only requires local information.* □

[3] *The agent i is an out-neighbor of j if the (i, j)-th entry of A is nonzero, or, equivalently, if (j, i) belongs to the edge set.*

A class of undetectable attacks is now presented. Notice that misbehaving agents can arbitrarily change their initial state without being detected during the consensus iterations, and, by doing so, misbehaving components can cause at most a constant error on the final consensus value. Indeed, let A be a consensus matrix, and let K be the set of misbehaving agents. Let $x(0)$ be the network initial state, and suppose that the agents K alter their initial value, so that the network initial state becomes $x(0) + B_K c$, where $c \in \mathbb{R}^{|K|}$. Recall from [65] that $\lim_{t\to\infty} A^t = \mathbf{1}\pi$, where $\mathbf{1}$ is the vector of all ones, and π is such that $\pi A = \pi$. Therefore , the effect of the misbehaving set K on the final consensus state is $\mathbf{1}\pi B_K c$. Clearly, if the vector $x(0) + B_K c$ is a valid initial state, the misbehaving agents cannot be detected. On the other hand, since it is possible for uncompromised nodes to estimate the observable part of the initial state of the whole network, if an acceptability region (or an a priori probability distribution) is available on initial states, then, by analyzing the reconstructed state, a form of intrusion detection can be applied, e.g., see [71]. We conclude this paragraph by showing that, if the misbehaving vector $B_K c$ belongs to the unobservability subspace of (A, C_j), for some j, then the misbehaving agents do not alter the final consensus value. Let v be an eigenvector associated with the unobservable eigenvalue $\bar{z}$, i.e., $(\bar{z}I - A)v = 0$ and $C_j v = 0$. We have $\pi(\bar{z}I - A)v = (\bar{z} - 1)\pi v = 0$, and, because of the detectability of (A, C_j), $|\bar{z}| < 1$ (cf. Lemma 8.3.1). Hence $\pi v = 0$. Therefore, if the attack $B_K c$ is unobservable from any agent, then $\lim_{t\to\infty} A^t B_K c = \mathbf{1}\pi B_K c = 0$, so that the change of the initial states of misbehaving agents does not affect the final consensus value.

A different class of unidentifiable attacks consists of injecting a signal corresponding to an input-zero for the current network state. We start by characterizing the potential disruption caused by misbehaving nodes that introduce nonzero, but exponentially vanishing inputs.[4]

Lemma 8.3.3 (Exponentially stable input) *Let A be a consensus matrix, and let K be a set of agents. Let $u : \mathbb{N} \mapsto \mathbb{R}^{|K|}$ be exponentially decaying. There exists $z \in (0,1)$ and $\bar{u} \in \mathbb{R}^{|K|}$ such that*

$$\lim_{t\to\infty} \sum_{\tau=0}^{t} A^{t-\tau} B_K u(\tau) \preceq (1-z)^{-1} \mathbf{1}\pi B_K \bar{u},$$

where $\preceq$ denotes component-wise inequality, $\mathbf{1}$ is the vector of all ones of appropriate dimension, and π is such that $\pi A = \pi$.

Proof: Let $z \in (0,1)$ and $0 \preceq u_0 \in \mathbb{R}^{|K|}$ be such that $u(k) \preceq z^k u_0$. Then, since A is a nonnegative matrix, for all $t, \tau \in \mathbb{N}$, with $t \geq \tau$, we have $A^{t-\tau} B_K u(\tau) \preceq A^{t-\tau} B_K z^\tau u_0$, and hence $\lim_{t\to\infty} \sum_{\tau=0}^{t} A^{t-\tau} B_K u(\tau) \preceq \lim_{t\to\infty} \sum_{\tau=0}^{t} A^{t-\tau} B_K z^\tau u_0$.

[4] An output-zeroing input can always be written as $u(k) = -(CA^\nu B)^\dagger CA^{\nu+1}(K_\nu A)^k x(0) - (CA^\nu B)^\dagger CA^{\nu+1} \left(\sum_{l=0}^{k-1} (K_\nu A)^{k-1-l} B u_h(l)\right) + u_h(h)$, where $\nu \in \mathbb{N}$, $(CA^\nu B)$ is the first nonzero Markov parameter, $K_\nu = I - B(CA^\nu B)^\dagger CA^\nu$ is a projection matrix, $x(0) \in \bigcap_{l=0}^{\nu} \operatorname{Ker}(CA^l)$ is the system initial state, and $u_h(k)$ is such that $CA^\nu B u_h(k) = 0$ [111].

Notice that $(1-z)^{-1} = \lim_{t\to\infty} \sum_{\tau=0}^{t} z^{\tau}$. We now show that $\lim_{t\to\infty} \sum_{\tau=0}^{t} z^{\tau}(\mathbf{1}\pi - A^{t-\tau}) = \lim_{t\to\infty} \sum_{\tau=0}^{t} E(t,\tau) \preceq 0$, from which the theorem follows. Let $e(t,\tau)$ be any component of $E(t,\tau)$. Because $\lim_{t\to\infty} A^t = \mathbf{1}\pi$, there exist c and ρ, with $|z| \leq |\rho| < 1$, such that $e(t,\tau) \leq c z^{\tau} \rho^{t-\tau}$. We have

$$\lim_{t\to\infty} \sum_{\tau=0}^{t} c z^{\tau} \rho^{t-\tau} = \lim_{t\to\infty} c\rho^{t} \sum_{\tau=0}^{t} z^{\tau} \rho^{-\tau} = 0,$$

so that $\sum_{\tau=0}^{t} E(t,\tau)$ converges to zero as t approaches infinity. □

Following Lemma 8.3.3, if the zero dynamics are exponentially stable, then misbehaving agents can affect the final consensus value by a constant amount without being detected, if and only if they inject vanishing inputs along input-zero directions. If an admissible region is known for the network state, then a tight bound on the effect of misbehaving agents injecting vanishing inputs can be provided. Notice moreover that, in this situation, a well-behaving agent is able to detect misbehaving agents whose state is outside an admissible region by simply analyzing its state. Finally, for certain consensus networks, the effect of an exponentially stable input decreases to zero with the cardinality of the network. Indeed, let $\pi = \bar{\pi}/n$, where $\bar{\pi}$ is a constant row vector and n denotes the cardinality of the network. For instance, if A is doubly stochastic, then $\pi = \mathbf{1}^{\mathsf{T}}/n$ [65]. Then, when n grows, the effect of the input $u(t) = z^t \bar{u}$, with $|z| < 1$, on the consensus value becomes negligible.

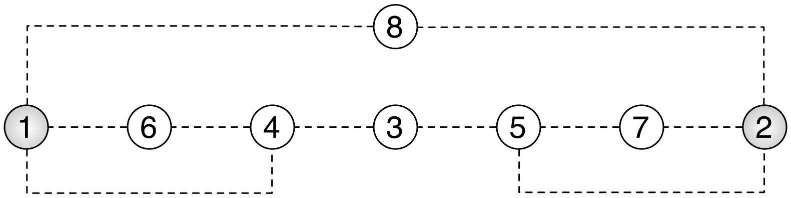


Figure 8.1: The agents $\{1, 2\}$ are misbehaving. The consensus system $(A, B_{\{1,2\}}, C_3)$ has unstable zeros.

The left-invertibility and the stability of the zero dynamics are not an inherent property of a consensus system. Consider for instance the graph of Fig. 8.1, where the agents $\{1, 2\}$ are malicious. If the network matrices are

$$A = \begin{bmatrix} 1/2 & 0 & 1/2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1/2 & 0 & 0 & 0 & 1/2 & 0 & 0 \\ 0 & 0 & 1/3 & 1/3 & 1/3 & 0 & 0 & 0 \\ 1/16 & 0 & 5/8 & 1/16 & 0 & 1/4 & 0 & 0 \\ 0 & 1/16 & 1/4 & 0 & 5/16 & 0 & 3/8 & 0 \\ 1/2 & 0 & 0 & 1/2 & 0 & 0 & 0 & 0 \\ 0 & 1/3 & 0 & 0 & 2/3 & 0 & 0 & 0 \\ 1/2 & 1/2 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \; B_{\{1,2\}} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix},$$

$$C_3 = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix},$$

then the system $(A, B_{\{1,2\}}, C_3)$ is left-invertible, but the invariant zeros are $\{0, +2, -2\}$. Hence, for some initial conditions, there exist non vanishing input sequences that do not appear in the output. Moreover, for the graph in Fig. 8.2, let the network matrices be

$$A = \begin{bmatrix} 1/3 & 1/3 & 0 & 0 & 0 & 0 & 0 & 0 & 1/3 \\ 1/3 & 1/3 & 1/3 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1/4 & 1/4 & 1/4 & 0 & 0 & 0 & 1/4 & 0 \\ 0 & 0 & 1/4 & 1/4 & 1/4 & 0 & 0 & 0 & 1/4 \\ 0 & 0 & 0 & 1/3 & 1/3 & 1/3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1/3 & 1/3 & 1/3 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1/3 & 1/3 & 1/3 & 0 \\ 0 & 0 & 1/4 & 0 & 0 & 0 & 1/4 & 1/4 & 1/4 \\ 1/4 & 0 & 0 & 1/4 & 0 & 0 & 0 & 1/4 & 1/4 \end{bmatrix}, \; B_{\{1,2\}} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix},$$

$$C_6 = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix}.$$

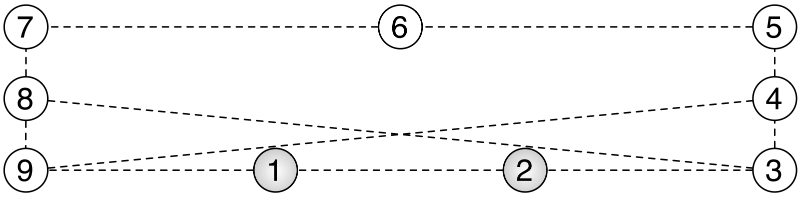


Figure 8.2: The agents $\{1, 2\}$ are misbehaving. The consensus system $(A, B_{\{1,2\}}, C_6)$ is not left-invertible.

It can be verified that the system $(A, B_{\{1,2\}}, C_6)$ is not left-invertible. Indeed, for zero initial conditions, any input of the form $u_1 = -u_2$ does not appear in the output sequence of the agent 6. In some cases, the left-invertibility of a consensus system can be asserted independently of the consensus matrix.

Theorem 8.3.4 (Left-invertibility, single intruder case) *Let A be a consensus matrix, and let $B_i = e_i$, $C_j = e_j^\mathsf{T}$. Then the system (A, B_i, C_j) is left-invertible.*

Proof: Suppose, by contradiction, that (A, B_i, C_j) is not left-invertible. Then there exist state trajectories that, starting from the origin, are invisible to the output. In other words, since the input is a scalar, the Markov parameters $C_j A^t B_i$ have to be zero for all t. Notice the (i, k)-th component of A^t is nonzero if there exists a path of length t from i to k. Because A is irreducible, there exists t such that $C_j A^t B_i \neq 0$, and therefore the consensus system is left-invertible. □

If in Theorem 8.3.4 one identifies the i-th node with a single intruder, and the j-th node with an observer node, the theorem states that, for known initial conditions of the network, any two distinct inputs generated by a single intruder produce different outputs at all observing nodes, and hence can be detected. Consider for example a flocking application, in which the agent are supposed to agree on the velocity to be maintained during the execution of the task [89]. Suppose that a linear consensus iteration is used to compute a common velocity vector, and suppose that the states of the agents are equal to each other. Then no single misbehaving agent can change the velocity of the team without being detected, because no zero dynamic can be generated by a single agent starting from a consensus state.

We now consider the case in which several misbehaving agents are allowed to act simultaneously. The following result relating the position of the misbehaving agents in the network and the zero dynamics of a consensus system can be asserted.

Theorem 8.3.5 (Stability of zero dynamics) *Let K be a set of agents and let j be a network node. The zero dynamics of the consensus system (A, B_K, C_j) are exponentially stable if one of the following is true:*

(i) the system (A, B_K, C_j) is left-invertible, and there are no edges from the nodes K to $V \setminus \{N_j \cup K\}$;

(ii) the system (A, B_K, C_j) is left-invertible, and there are no edges from the nodes $V \setminus \{N_j \cup K\}$ to N_j; or

(iii) the sets K and N_j are such that $K \subseteq N_j$.

Proof: Let z be an invariant zero, x and u a state-zero and input-zero direction, so that

$$(zI - A)x + B_K u = 0, \text{ and } C_j x = 0 \tag{8.3}$$

Reorder the nodes such that the set K comes first, the set $N_j \setminus K$ second, and the set $V \setminus \{K \cup N_j\}$ third. The consensus matrix and the vector x are accordingly partitioned as

$$A = \begin{bmatrix} A_{11} & A_{12} & A_{13} \\ A_{21} & A_{22} & A_{23} \\ A_{31} & A_{32} & A_{33} \end{bmatrix}, \quad x = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix},$$

and the input and output matrices become $B_K = [I\ 0\ 0]^\mathsf{T}$ and $C_j = [*\ I\ 0]$. For equations (8.3) to be verified, it has to be $x_2 = 0$, $zx_1 = A_{11}x_1 + A_{13}x_3 - u_k$, and

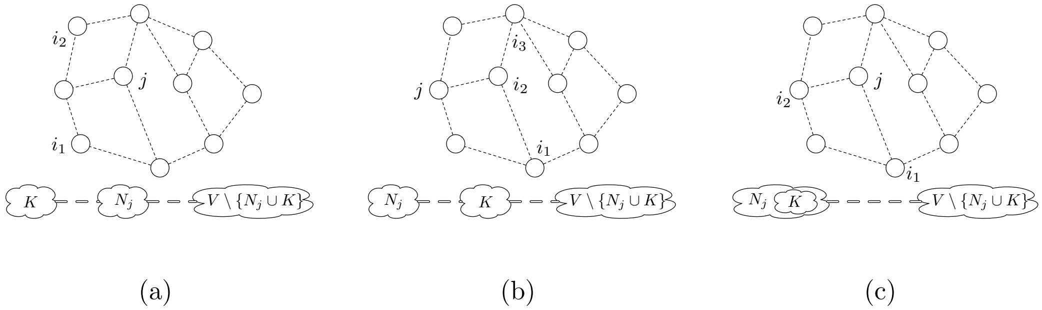
$$\begin{bmatrix} 0 \\ zx_3 \end{bmatrix} = \begin{bmatrix} A_{21} & A_{23} \\ A_{31} & A_{33} \end{bmatrix} \begin{bmatrix} x_1 \\ x_3 \end{bmatrix}.$$

Case (i). Since there are no edges from the nodes K to $V \setminus \{N_j \cup K\}$, we have $A_{31} = 0$, and hence it has to be $(zI - A_{33})x_3 = 0$, i.e., z needs to be an eigenvalue of A_{33}. We now show that $x_3 \neq 0$. Suppose by contradiction that $x_3 = 0$, and that z is an invariant zero, with state-zero and input-zero direction

$x = [x_1^\mathsf{T} \, 0 \, 0]^\mathsf{T}$ and $u_K = (zI - A_{11})x_1$, respectively. Then, for all complex value $\bar{z}$, the vectors x and $u_K = (\bar{z}I - A_{11})x_1$ constitute the state-zero and the input-zero direction associated with the invariant zero $\bar{z}$. Because the system is assumed to be left-invertible, there can only be a finite number of invariant zeros [111], so that we conclude that $x_3 \neq 0$ or that the system has no zero dynamics. Because z needs to be an eigenvalue of A_{33}, and because of Lemma 8.1.1, we conclude that the zero dynamics are asymptotically stable.

Case (ii). Since there are no edges from the nodes $V \setminus \{N_j \cup K\}$ to N_j, we have $A_{23} = 0$. We now show that $\mathrm{Ker}(A_{21}) = 0$. Suppose by contradiction that $0 \neq x_1 \in \mathrm{Ker}(A_{21})$. Consider the equation $(zI - A_{33})x_3 = A_{31}x_1$, and notice that, because of Lemma 8.1.1, for all z with $|z| \geq 1$, the matrix $zI - A_{33}$ is invertible. Therefore, if $|z| \geq 1$, the vector $[(x_1)^\mathsf{T} \, 0 \, ((zI - A_{33})^{-1}A_{31}x_1)^\mathsf{T}]^\mathsf{T}$ is a state-zero direction, with input-zero direction $u_K = -(zI - A_{11})x_1 + A_{13}x_3$. The system would have an infinite number of invariant zeros, being therefore not left-invertible. We conclude that $\mathrm{Ker}(A_{21}) = 0$. Consequently, we have $x_1 = 0$ and $(zI - A_{33})x_3 = 0$, so that $|z| < 1$.

Case (iii). Reorder the variables such that the nodes N_j come before $V \setminus N_j$. For the existence of a zero dynamics, it needs to hold $x_1 = 0$ and $(zI - A_{22})x_2 = 0$. Hence, $|z| < 1$. □


i_2
j
i_1
K
N_j
$V \setminus \{N_j \cup K\}$
i_3
j
i_2
i_1
N_j
K
$V \setminus \{N_j \cup K\}$
j
i_2
i_1
N_j
K
$V \setminus \{N_j \cup K\}$

(a) (b) (c)

Figure 8.3: The stability of the zero dynamics of a left-invertible consensus system can be asserted depending upon the location of the misbehaving agents in the network. Let j be the observer agent, and let K be the misbehaving set. Then, the zero dynamics are asymptotically stable if the set N_j separates the sets K and $V \setminus \{N_j \cup K\}$ (cfr. Fig. 8.3(a)), or if the set K separates the sets N_j and $V \setminus \{N_j \cup K\}$ (cfr. Fig. 8.3(b)), or if the set K is a subset of N_j (cfr. Fig. 8.3(c)).

We are left to study the case of a network with zeros outside the open unit disk, where intruders may inject non-vanishing inputs while remaining unidentified. For this situation, we only remark that a detection procedure based on an admissible region for the network state can be implemented to detect inputs evolving along unstable zero directions.

8.4 Generic Detection and Identification of Misbehaving Agents

In this section we adopt the theory of structured system presented in Section 4.3 to investigate the resilience of consensus algorithms from a structural perspective. Let the connectivity of a structured system $([A], [B], [C])$ be the connectivity of the graph defined by its nonzero parameters. In what follows, we assume $[D] = 0$, and we study the zero dynamics of a structured consensus system as a function of its connectivity. Let the generic rank of a structure matrix $[M]$ be the maximal rank over all possible numerical realizations of $[M]$.

Lemma 8.4.1 (Generic zero dynamics and connectivity) *Let* $([A], [B], [C])$ *be a* k*-connected structured system. If the generic rank of* $[B]$ *is less than* k*, then almost every numerical realization of* $([A], [B], [C])$ *has no zero dynamics.*

Proof: Since the system $([A], [B], [C])$ is k-connected and the generic rank r of $[B]$ is less than k, there are r disjoint paths from the input to the output [114]. Then, from Theorem 4.3 in [114], the system $([A], [B], [C])$ is generically left-invertible. Additionally, by using Lemma 3 in [106], it can be shown that $([A], [B], [C])$ has generically no invariant zeros. We conclude that almost every realization of $([A], [B], [C])$ has no nontrivial zero dynamics. □

Given a structured triple $([A], [B], [C])$ with d nonzero elements, the set of parameters that make $([A], [B], [C])$ a consensus system is a subset S of $\mathbb{R}^d$, because the matrix A needs to be row stochastic and primitive. A certain property that holds generically in $\mathbb{R}^d$ needs not be valid generically with respect to the feasible set S. Let $([A], [B], [C])$ be structure matrices, and let $S \subset \mathbb{R}^d$ be the set of parameters that make $([A], [B], [C])$ a consensus system. We next show that the left-invertibility and the number of invariant zeros are generic properties with respect to the parameter space S.

Theorem 8.4.2 (Genericity of consensus systems) *Let* $([A],[B],[C])$ *be a* k*-connected structured system. If the generic rank of* $[B]$ *is less than* k*, then almost every consensus realization of* $([A],[B],[C])$ *has no zero dynamics.*

Proof: Let d be the number of nonzero entries of the structured system $([A],[B],[C])$. From Theorem 8.4.1 we know that, generically with respect to the parameter space $\mathbb{R}^d$, a numerical realization of $([A],[B],[C])$ has no zero dynamics. Let $S \subset \mathbb{R}^d$ be the subset of parameters that makes $([A],[B],[C])$ a consensus system. We want to show that the absence of zero dynamics is a generic property with respect to the parameter space S. Observe that S is dense in $\mathbb{R}^{\bar{d}}$, where $\bar{d} \leq d-n$ and n is the dimension of $[A]$. Then [45, 109], it can be shown that, in order to prove that our property is generic with respect to S, it is sufficient to show that there exist some consensus systems which have no zero dynamics. To construct a consensus system with no zero dynamics consider the following procedure. Let (A,B,C) be a nonnegative and irreducible linear system with no zero dynamics, where the number of inputs is strictly less that the connectivity of the associated graph. Notice that, following the above discussion, such system can always be found. The Perron-Frobenius Theorem for nonnegative matrices ensures the existence of a positive eigenvector x of A associated with the eigenvalue of largest magnitude r [65]. Let D be the diagonal matrix whose main diagonal equals x, then the matrix $r^{-1}D^{-1}AD$ is a consensus matrix [67]. A change of coordinates of (A, B, C) using D yields the system $(D^{-1}AD, D^{-1}B, CD)$, which has no zero dynamics. Finally, the system $(r^{-1}D^{-1}AD, D^{-1}B, CD)$ is a k-connected consensus system with, generically, no zero dynamics. Indeed, if there exists a value $\bar{z}$, a state-zero direction x_0, and an input-zero direction g for the system $(r^{-1}D^{-1}AD, D^{-1}B, CD)$, then the value $\bar{z}r$, with state direction x_0/r and input direction g, is an invariant zero of $(D^{-1}AD, D^{-1}B, CD)$, which contradicts the hypothesis. □

Because a sufficiently connected consensus system has generically no zero dynamics, the following remarks about the robustness of a generic property should be considered. First, generic means open, i.e. some appropriately small perturbations of the matrices of the system having a generic property do not destroy this property. Second, generic implies dense, hence any consensus system which does not have a generic property can be changed into a system having this property just by arbitrarily small perturbations. We are now able to state our generic resilience results for consensus networks.

Theorem 8.4.3 (Generic identification of misbehaving agents) *Given a k-connected consensus network, generically, there exist only identifiable inputs for* *any set of $\lfloor \frac{k-1}{2} \rfloor$ misbehaving agents. Moreover, generically, there exist identifiable inputs for every set of $k-1$ misbehaving agents.*

Proof: Since $2\lfloor \frac{k-1}{2} \rfloor < k$, by Lemma 8.4.1 the consensus system with any set of $2\lfloor \frac{k-1}{2} \rfloor$ has generically no zero dynamics. By Theorem 8.2.4, any set of $\lfloor \frac{k-1}{2} \rfloor$ malicious agents is detectable and identifiable by every node in the network. We now consider the case of faulty agents. Let V be the set of nodes, and $K_1, K_2 \subset V$, with $|K_1| = |K_2| = k-1$, be two disjoint sets of faulty agents. Let $j \in V$. We need to show the existence of identifiable, i.e., faulty, inputs. By using a result of [114] on the generic rank of the matrix pencil of a structured system, since the given consensus network is k-connected and $|K_1| = k-1$, it can be shown that the system $(A, [B_{K_1} \;\; B_i], C_j)$, for all $i \in K_2$, is left-invertible, which confirms the existence of identifiable inputs for the current network state. By Definition 9, we conclude that the faulty set K_1 is generically identifiable by any well-behaving agent. □

In other words, in a k-connected network, up to $\lfloor \frac{k-1}{2} \rfloor$ (resp. $k-1$) malicious (resp. faulty) agents are generically identifiable by every well behaving agent. Analogously, it can be shown that generically up to $k-1$ misbehaving agents are generically detectable. In the next section, we describe three algorithms to detect and identify misbehaving agents.

8.5 Intrusion Detection Algorithms

In this section we present three decentralized algorithms to detect and identify misbehaving agents in a consensus network. Although the first two algorithms require only local measurements, the complete knowledge of the consensus network is necessary for the implementation. The third algorithm, instead, requires the agents to know only a certain neighborhood of the consensus graph, and it allows for a local identification of misbehaving agents. As it will be clear in the sequel, the third algorithm overcomes, under a reasonable set of assumptions, the limitations inherent to centralized detection and identification procedures. Our first algorithm is based upon the following result.

Theorem 8.5.1 (Detection filter) *Let K be the set of misbehaving agents. Assume that the zero dynamics of the consensus system (A, B_K, C_j) are exponentially stable, for some j. Let A_{N_j} denote the N_j columns of the matrix A. The filter*

$$\begin{aligned} z(t+1) &= (A + GC_j)z(t) - GC_j x(t), \\ \tilde{x}(t) &= Lz(t) + HC_j x(t), \end{aligned} \tag{8.4}$$

with $G = -A_{N_j}$, $H = C_j^\mathsf{T}$, and $L = I - HC_j$, is such that, in the limit for $t \to \infty$, the vector $\tilde{x}(t+1) - A\tilde{x}(t)$ is nonzero only if the input u_K is nonzero. Moreover, if $K \subset N_j$, then the filter (8.4) asymptotically estimates the state of the network, independent of the behavior of the misbehaving agents K.

Proof: Let $G = -A_{N_j}$, and consider the estimation error $e(t+1) = z(t+1) - x(t+1) = (A + GC_j)e(t) - B_K u_K(t)$. Notice that $Le = Lz + C_j^{\mathsf{T}} C_j x - x$, and hence $\tilde{x} = x + Le$. Consequently, $\tilde{x}(t+1) - A\tilde{x}(t) = B_K u_K(t) + Le(t+1) - ALe(t)$. By using Lemma 8.1.1, it is a straightforward matter to show that $(A + GC_j)$ is Schur stable. If $u_K = 0$, then $\tilde{x}(t+1) - A\tilde{x}(t)$ converges to zero. Suppose now that $K \subseteq N_j$. The reachable set of e, i.e., the minimum $(A + GC_j)$ invariant containing $\mathcal{B}_K$, coincides with $\mathcal{B}_K$. Indeed $(A + GC_j)\mathcal{B}_K = \emptyset$. Since $\mathcal{B}_K \subseteq \mathrm{Ker}(L)$ by construction, the vectors Le and $\tilde{x} - x$ converge to zero. □

By means of the filter described in the above theorem, a distributed intrusion detection procedure can be designed, see [76]. Here, each well-behaving agent only implements one detection filter, making the asymptotic detection task computationally easy to be accomplished. We remark that, since the filter converges exponentially, an exponentially decaying input of appropriate size may remain undetected (see Lemma 8.3.3 for a characterization of the effect of exponentially vanishing inputs on the final consensus value). For a finite time detection of misbehaving agents, and for the identification of misbehaving components, a more sophisticated algorithm is presented in Algorithm 5.

Theorem 8.5.2 (Complete identification) *Let A be a consensus matrix, let K be the set of misbehaving agents, and let c be the connectivity of the consensus network. Assume that:*

(i) every agent knows the matrix A and $k \geq |K|$, and

(ii) $k < c$, if the set K is faulty, and $2k < c$ if the set K is malicious.

Then the Complete Identification algorithm allows each well-behaving agent to generically detect and identify every misbehaving agent in finite time.

Proof: We focus on agent j. Let $k = |K|$, and let $\mathcal{K}$ be the set containing all the $\binom{n-1}{k+1}$ combinations of $k+1$ elements of $V \setminus \{j\}$. For each set $\tilde{K} \in \mathcal{K}$, consider the system $\Sigma_{\tilde{K}} = (A, B_{\tilde{K}}, C_j)$, and compute[5] a set of residual generator filters for $\Sigma_{\tilde{K}}$. If the connectivity of the communication graph is sufficiently high, then, generically, each residual function is nonzero if and only if the corresponding input is nonzero. Let K be the set of misbehaving nodes, then, whenever $K \subset \tilde{K}$, the residual function associated with the input $\tilde{K} \setminus K$ becomes zero after an initial transient, so that the agent $\tilde{K} \setminus K$ is recognized as well-behaving. By exclusion, because the residuals associated with the misbehaving agents are always nonzero, the set K is identified. □

By means of the Complete Identification algorithm, the detection and the identification of the misbehaving agents take place in finite time, because the residual generators can be designed as dead-beat filters, and independent of the misbe-

[5] We refer the interested reader to [62] for a design procedure of a dead beat residual generator. Notice that the possibility of detecting and identifying the misbehaving agents is, as discussed in Section 8.2 and 8.4, guaranteed by the absence of zero dynamics in the consensus system.

Algorithm 5 *Complete Identification* (j-th agent)

```
Input    : $A$; $k \geq |K|$;
Require  : The connectivity of $A$ to be $k+1$, if $K$ is faulty, and
           $2k+1$ otherwise;
Compute the residual generators for every set of $k+1$ misbehaving agents;
while the misbehaving agents are unidentified do
    Exchange data with the neighbors;
    Update the state;
    Evaluate the residual functions;
    if every $i_{th}$ residual is nonzero then
        Agent $i$ is recognized as misbehaving.
```

having input. It should be noticed that, although no communication overhead is introduced in the consensus protocol, the Complete Identification procedure relies on strong assumptions. First, each agent needs to know the entire graph topology, and second, the number of residual generators that each node needs to design is proportional to $\binom{n-1}{k}$. Because an agent needs to update these filters after each communication round, when the cardinality of the network grows, the computational burden may overcome the capabilities of the agents, making this procedure inapplicable.

In the remaining part of this section, we present a computationally efficient procedure that only assumes partial knowledge of the consensus network but yet allows for a local identification of the misbehaving agents. Let A be a consensus matrix, and observe that it can be written as $A_d + \varepsilon\Delta$, where $\|\Delta\|_\infty = 2$, $0 \leq \varepsilon \leq 1$, and A_d is block diagonal with a consensus matrix on each of the N diagonal blocks. For instance, let $A = [a_{kj}]$, and let $V_1, \dots, V_N$ be the subsets of agents associated with the blocks. Then the matrix $A_d = [\bar{a}_{kj}]$ can be defined as

(i) $\bar{a}_{kj} = a_{kj}$ if $k \neq j$, and $k, j \in V_i$, $i \in \{1, \dots, N\}$,

(ii) $\bar{a}_{kk} = 1 - \sum_{j \in V_i, j \neq k} a_{kj}$, and

(iii) $\bar{a}_{kj} = 0$ otherwise.

Moreover, $\Delta = 2(A - A_d)/\|(A - A_d)\|_\infty$, and $\varepsilon = \frac{1}{2}\|A - A_d\|_\infty$. Note that, if ε is "small", then the agents belonging to different groups are weakly coupled. We assume the groups of weakly coupled agents to be given, and we leave the problem of finding such partitions as the subject of future research, for which the ideas presented in [21, 83] constitute a very relevant result.

We now focus on the h-th block. Let $K = v \cup l$ be the set of misbehaving agents, where $v = V_h \cap K$, and $l = K \setminus v$. Assume that the set v is identifiable by agent $j \in V_h$ (see Section 8.2). Then, agent j can identify the set v by means of a set of residual generators, each one designed to decouple a different set of $|v| + 1$

inputs. To be more precise, let $i \in V_h \setminus v$, and consider the system

$$\begin{aligned}\begin{bmatrix} x \\ w_v \end{bmatrix}^+ &= \begin{bmatrix} A_d & 0 \\ E_v C_j & F_v \end{bmatrix} \begin{bmatrix} x \\ w_v \end{bmatrix} + \begin{bmatrix} B_v & B_i \\ 0 & 0 \end{bmatrix} \begin{bmatrix} u_v \\ u_i \end{bmatrix}, \\ r_v &= \begin{bmatrix} H_v C_j & M_v \end{bmatrix} \begin{bmatrix} x \\ w_v \end{bmatrix}, \end{aligned} \tag{8.5}$$

and the system

$$\begin{aligned}\begin{bmatrix} x \\ w_i \end{bmatrix}^+ &= \begin{bmatrix} A_d & 0 \\ E_i C_j & F_i \end{bmatrix} \begin{bmatrix} x \\ w_i \end{bmatrix} + \begin{bmatrix} B_v & B_i \\ 0 & 0 \end{bmatrix} \begin{bmatrix} u_v \\ u_i \end{bmatrix}, \\ r_i &= \begin{bmatrix} H_i C_j & M_i \end{bmatrix} \begin{bmatrix} x \\ w_i \end{bmatrix}, \end{aligned} \tag{8.6}$$

where the quadruple (F_v, E_v, M_v, H_v) (resp. (F_i, E_i, M_i, H_i)) describes a filter of the form (2.3), and it is designed as in [62]. Then the misbehaving agents v are identifiable by agent j because v is the only set such that, for every $i \in V_h \setminus v$, it holds $r_v \not\equiv 0$ and $r_i \equiv 0$ whenever $u_v \not\equiv 0$. It should be noticed that, since A_d is block diagonal, the residual generators to identify the set v can be designed by only knowing the h-th block of A_d, and hence only a finite region of the original consensus network. By applying the residual generators to the consensus system

$A_d + \varepsilon\Delta$ with misbehaving agents K we get

$$\begin{bmatrix} \hat{x} \\ \hat{w}_v \end{bmatrix}^+ = \bar{A}_{\varepsilon,v} \begin{bmatrix} \hat{x} \\ \hat{w}_v \end{bmatrix} + \begin{bmatrix} B_v & B_l & B_i \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} u_v \\ u_l \\ u_i \end{bmatrix},$$

$$\hat{r}_v = \begin{bmatrix} H_v C_j & M_v \end{bmatrix} \begin{bmatrix} \hat{x} \\ \hat{w}_v \end{bmatrix},$$

and

$$\begin{bmatrix} \hat{x} \\ \hat{w}_i \end{bmatrix}^+ = \bar{A}_{\varepsilon,i} \begin{bmatrix} \hat{x} \\ \hat{w}_i \end{bmatrix} + \begin{bmatrix} B_v & B_l & B_i \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} u_v \\ u_l \\ u_i \end{bmatrix},$$

$$\hat{r}_i = \begin{bmatrix} H_i C_j & M_i \end{bmatrix} \begin{bmatrix} \hat{x} \\ \hat{w}_i \end{bmatrix},$$

where

$$\bar{A}_{\varepsilon,v} = \begin{bmatrix} A_d + \varepsilon\Delta & 0 \\ E_v C_j & F_v \end{bmatrix}, \quad \bar{A}_{\varepsilon,i} = \begin{bmatrix} A_d + \varepsilon\Delta & 0 \\ E_i C_j & F_i \end{bmatrix}.$$

Because of the matrix Δ and the input u_l, the residual r_i is generally nonzero even if $u_i \equiv 0$. However, the misbehaving agents v remain identifiable by j if for each $i \in V_h \setminus v$ we have $\|\hat{r}_v\|_\infty > \|\hat{r}_i\|_\infty$ for all $u_v \not\equiv 0$.

Theorem 8.5.3 (Local identification) *Let V be the set of agents, let K be the set of misbehaving agents, and let $A_d + \varepsilon\Delta$ be a consensus matrix, where*

Algorithm 6 *Local Identification* (j-th agent)

```
Input      : $A_h$; $k_j \geq |K \cap V_h|$; threshold $T_h$
Require    : The connectivity of $A_d^j$ to be $k_j + 1$, if $K$ is faulty, and
             $2k_j + 1$ otherwise;
while the misbehaving agents are unidentified do
    Exchange data with the neighbors;
    Update the state;
    Evaluate the residual functions;
    if $i_{th}$ residual is greater than $T_h$ then
        Agent $i$ is recognized as misbehaving.
```

A_d is block diagonal, $\|\Delta\|_\infty = 2$, and $0 \leq \varepsilon \leq 1$. Let each block h of A_d be a consensus matrix with agents $V_h \subseteq V$, and with connectivity $|K \cap V_h| + 1$. There exists $\alpha > 0$ and $u_{\max} \geq 0$, such that, if each input signal u_i, $i \in K$, takes value in $\mathcal{U} = \{u : \varepsilon\alpha u_{\max} \leq \|u\|_\infty \leq u_{\max}\}$,[6] then each well-behaving agent $j \in V_h$ identifies in finite time the faulty agents $K \cap V_h$ by means of the Local Identification algorithm.

Proof: We focus on the agent $j \in V_h$, and, without loss of generality, we assume that $u_K(0) \neq 0$, and that the residual generators have a finite impulse response.

[6] *The norm $\|u\|_\infty$ is intended in the vector sense at every instant of time. The misbehaving input is here assumed to be nonzero at every instant of time.*

Let $d_j = \|V_h\|$, and note that d_j time steps are sufficient for each agent $j \in V_h$ to identify the misbehaving agents. Let u^t denote the input sequence up to time t. Let $v = K \cap V_h$, $l = K \setminus v$, and observe that $\hat{r}_v(d_j) = \left[\begin{smallmatrix} H_vC_j & M_v \end{smallmatrix}\right] \bar{A}^{d_j}_{\varepsilon,v}\bar{x}(0) + \hat{h}_v \star u_v^{d_j-1} + \hat{h}_l \star u_l^{d_j-1}$, where $\hat{h}_v$ and $\hat{h}_l$ denote the impulse response from u_v and u_l respectively, and $\star$ denotes the convolution operator. We now determine an upper bound for each term of $\hat{r}_v(d_j)$. Let the misbehaving inputs take value in $\mathcal{U} = \{u : \varepsilon\alpha u_{\max} \leq \|u\|_\infty \leq u_{\max}\}$. By using the triangle inequality on the impulse responses of the residual generator, it can be shown that $\|\hat{h}_l \star u_l^{d_j-1}\|_\infty \leq \|h_l \star u_l^{d_j-1}\|_\infty + \varepsilon c_1 u_{\max} = \varepsilon c_1 u_{\max}$, where h_l denotes the impulse response form u_l to r_v of the system (8.5), and c_1 is a finite positive constant independent of ε. Moreover, it can be shown that there exist two positive constant c_2 and c_3 such that $\| \left[\begin{smallmatrix} H_vC_j & M_v \end{smallmatrix}\right] \bar{A}^{d_j}_{\varepsilon,v}\bar{x}(0)\|_\infty \leq \varepsilon c_2 u_{\max}$, and $\min_{u_v \in \mathcal{U}} \|\hat{h}_v \star u_v^{d_j-1}\|_\infty \geq \min_{u_v \in \mathcal{U}} \|h_v \star u_v^{d_j-1}\|_\infty - \varepsilon c_3 u_{\max}$. Analogously, for the residual generator associated with the well-behaving agent i, we have $\hat{r}_i(d_j) = \left[\begin{smallmatrix} H_iC_j & M_i \end{smallmatrix}\right] \bar{A}^{d_j}_{\varepsilon,i}\bar{x}(0) + \hat{h}_v \star u_v^{d_j-1} + \hat{h}_l \star u_l^{d_j-1}$, and hence $\hat{r}_i(d_j) \leq \varepsilon(c_4^{(i)} + c_5^{(i)} + c_6^{(i)})u_{\max}$. Let $\bar{c} = c_1 + c_2 + c_3 + \max_{i \in V_h \setminus v}(c_4^{(i)} + c_5^{(i)} + c_6^{(i)})$, and let β be such that $\min_{u_v \in \mathcal{U}} \|h_v \star u_v^{d_j-1}\|_\infty > \beta u_{\min}$. Then a correct identification of the misbehaving agents v takes place if $\beta u_{\min} = \beta\varepsilon\alpha u_{\max} > \varepsilon\bar{c}u_{\max}$, and hence if $\alpha > \bar{c}/\beta$. □

Notice that the constant α in Theorem 8.5.3 can be computed by bounding the infinity norm of the impulse response of the residual generators. An example is in Section 8.6.2. A procedure to achieve local detection and identification of misbehaving agents is in Algorithm 6, where A_d^h denotes the h-th block of A_d, and T_h the corresponding threshold value. Observe that in the Local Identification procedure an agent only performs local computation, and it is assumed to have only local knowledge of the network structure.

Remark 14 (Local identifiability) *It is a nontrivial fact that the misbehaving agents become locally identifiable depending on the magnitude of ε. Indeed, as long as $\varepsilon > 0$, the effect of the perturbation $\varepsilon\Delta$ on the residuals becomes eventually relevant and prevents, after a certain time, a correct identification of the misbehaviors [83].* □

8.6 Numerical Examples

8.6.1 Complete detection and identification

Consider the network of Fig. 8.4, and let A be a randomly chosen consensus matrix. In particular, let

$$A = \begin{bmatrix} 0.2795 & 0.1628 & 0 & 0.1512 & 0.4066 & 0 & 0 & 0 \\ 0.0143 & 0.3363 & 0.3469 & 0 & 0 & 0.3025 & 0 & 0 \\ 0 & 0.0718 & 0.1904 & 0.2438 & 0 & 0 & 0.4941 & 0 \\ 0.0844 & 0 & 0.4457 & 0.0660 & 0 & 0 & 0 & 0.4040 \\ 0.1709 & 0 & 0 & 0 & 0.2694 & 0.2472 & 0 & 0.3125 \\ 0 & 0.4199 & 0 & 0 & 0.1575 & 0.3293 & 0.0932 & 0 \\ 0 & 0 & 0.0174 & 0 & 0 & 0.4241 & 0.2850 & 0.2735 \\ 0 & 0 & 0 & 0.3024 & 0.2039 & 0 & 0.2065 & 0.2873 \end{bmatrix}.$$

The network is 3-connected, and it can be verified that for any set K of 3 misbehaving agents, and for any observer node j, the triple (A, B_K, C_j) is left-invertible.

Also, for any set K of cardinality 2, and for any node j, the triple (A, B_K, C_j) has no invariant zeros. As previously discussed, any well-behaving node can detect and identify up to 2 faulty agents, or up to 1 malicious agent. Consider the observations of the agent 1, and suppose that the agents $\{3, 7\}$ inject a random signal into the network. As described in Algorithm 5, the agent 1 designs the residual generator filters and computes the residual functions for each of the $\binom{7}{3}$ possible sets of misbehaving nodes, and identify the well-behaving agents. Consider for example the system $x(t+1) = Ax(t) + B_3u_3(t) + B_4u_4(t) + B_7u_7(t)$, and suppose we want to design a filter of the form (2.3) which is only sensible to the signal u_4. The unobservability subspace $\mathcal{S}^M_{\{3,7\}} = (\mathcal{V}^*_{\{3,7\}} + \mathcal{S}^*_{\{3,7\}})$, is

$$\mathcal{S}^M_{\{3,7\}} = \text{Im}\left(\begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -0.6624 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & -0.4740 & -0.6597 & 0 \\ 0 & 0 & -0.8798 & 0.3548 & 0 \\ 0.4116 & 0 & -0.0327 & 0.0132 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0.9114 & 0 & 0.0148 & -0.0060 & 0 \end{bmatrix}\right),$$

and a possible choice for the matrices of the residual generator is

$$F = \begin{bmatrix} 0 & 0 & 0 \\ 0.0014 & -0.3222 & -0.3424 \\ -0.0013 & 0.3031 & 0.3222 \end{bmatrix},$$

$$E = \begin{bmatrix} 0.2795 & 0.1628 & 0.1512 & 0.4066 \\ 0.0138 & 0.4982 & -0.2280 & 0.2003 \\ 0.0082 & -0.6095 & 0.3012 & -0.1568 \end{bmatrix},$$

$$M = \begin{bmatrix} -1 & 0 & 0 \\ 0 & 0.9999 & 0.0128 \end{bmatrix}, \text{ and } H = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & -0.7491 & 0.5832 & -0.3142 \end{bmatrix}.$$

It can be checked that, independent of the initial condition of the network, the residual function associated with the input 4 is zero, as in 8.5, so that the agent

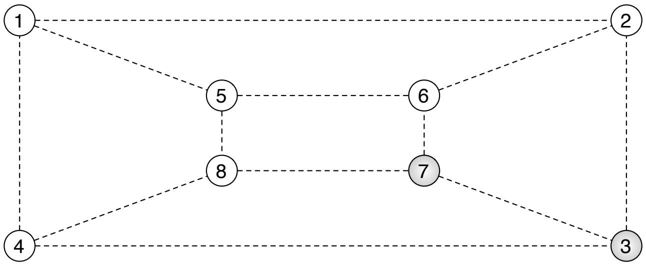


Figure 8.4: A consensus network where the nodes 3 and 7 are faulty.

4 is regarded as well-behaving. Agents 3, 7, instead, have always nonzero residual functions, and are recognized as misbehaving.

If the misbehaving nodes are allowed to be malicious, then no more than 1 misbehaving node can be tolerated. Indeed, because of Theorem 8.2.2, there exists a set $\bar{K}$ of 4 misbehaving agents such that the system $(A, B_{\bar{K}}, C_1)$ exhibits nontrivial zero dynamics. For instance, let $\bar{K} = \{2, 4, 6, 8\}$, and note that if the initial condition $x(0)$ belongs to

$$\mathcal{V}^* = \text{Im}\left(\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0.7842 & 0 \\ 0 & 0 & 1 \\ 0 & -0.6205 & 0 \end{bmatrix}\right),$$

then the input $u_K = F_b x$,[7] where

$$F_b = \begin{bmatrix} 0 & 0 & -0.3469 & 0 & 0 & -0.1860 & 0 & 0.1472 \\ 0 & 0 & -0.4457 & 0 & 0 & 0.1966 & 0 & -0.1555 \\ 0 & 0 & 0 & 0 & 0 & -0.1063 & -0.1148 & 0.0841 \\ 0 & 0 & 0 & 0 & 0 & 0.0636 & -0.1894 & -0.0503 \end{bmatrix},$$

[7] The malicious agents need to know the entire state to implement this feedback law. The case in which only local feedback is allowed is left as a direction for future research, for which the result in [107] is meaningful.

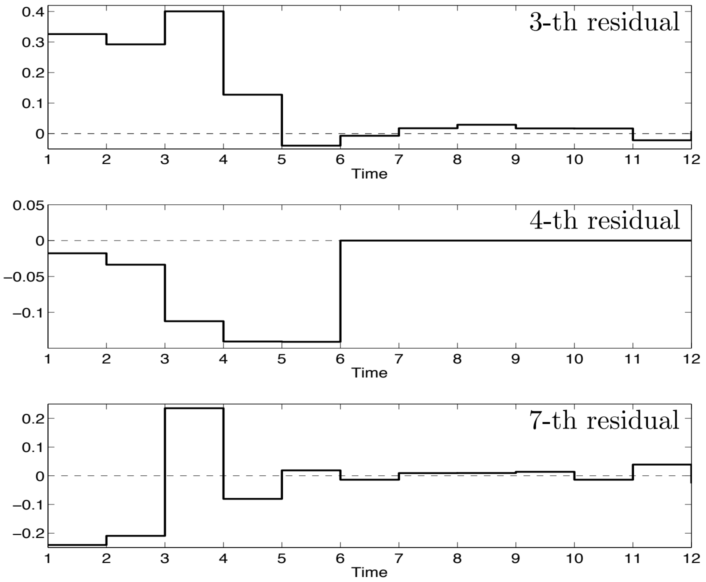


Figure 8.5: Residual functions computed by the agent 1 under the hypothesis that the misbehaving set is $\{3, 4, 7\}$.

is such that $y_1(t) = 0$ for all $t \geq 0$. Therefore, the two systems $(A, B_{\{2,4\}}, C_1)$ and $(A, B_{\{6,8\}}, C_1)$, with initial conditions $x_1(0)$ and $x_2(0) = x_1(0) - x(0)$, and inputs

$$u_{\{2,4\}}(t) = \left[\begin{smallmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{smallmatrix}\right] F_b(x_1(t) - x_2(t)),$$

$$u_{\{6,8\}}(t) = \left[\begin{smallmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{smallmatrix}\right] F_b(x_2(t) - x_1(t)),$$

have exactly the same output dynamics, so that the two sets $\{2, 4\}$ and $\{6, 8\}$ are indistinguishable by the agent 1.

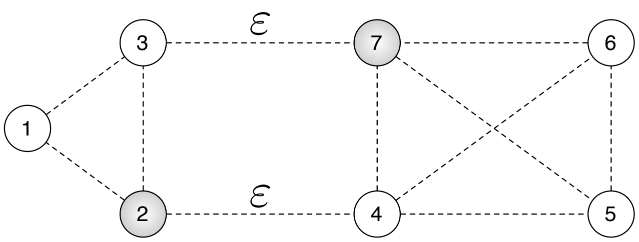


Figure 8.6: Consensus network with weak connections and misbehaving agents.

8.6.2 Local detection and identification

Consider the consensus network in Fig. 8.6, where $A = A_d + \varepsilon\Delta$, $\varepsilon \in \mathbb{R}$, $0 \leq \varepsilon \leq 1$, and

$$A_d = \begin{bmatrix} \frac{1}{3} & \frac{1}{3} & \frac{1}{3} & 0 & 0 & 0 & 0 \\ \frac{1}{3} & \frac{1}{3} & \frac{1}{3} & 0 & 0 & 0 & 0 \\ \frac{1}{3} & \frac{1}{3} & \frac{1}{3} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} \\ 0 & 0 & 0 & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} \\ 0 & 0 & 0 & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} \\ 0 & 0 & 0 & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} \end{bmatrix}, \Delta = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & -1 \end{bmatrix}.$$

Let $K = \{2, 7\}$ be the set of misbehaving agents, let $0.1 \leq u_2(t), u_7(t) \leq 3$ at each time t, and let $\|x(0)\|_\infty \leq 1$. Consider the agent 1, and let (F_2, E_2, M_2, H_2) and (F_3, E_3, M_3, H_3) be the residual generators as in (8.5) and (8.6), respectively, where

$$F_2 = \begin{bmatrix} -1/3 & -1/3 \\ 1/3 & 1/3 \end{bmatrix}, \quad E_2 = \begin{bmatrix} -2/3 & 0 & -1/3 \\ 2/3 & 0 & 1/3 \end{bmatrix},$$

$$M_2 = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}, \quad H_2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix},$$

and

$$F_3 = \begin{bmatrix} -1/3 & 1/3 \\ -1/3 & 1/3 \end{bmatrix}, \quad E_3 = \begin{bmatrix} -2/3 & -1/3 & 0 \\ -2/3 & -1/3 & 0 \end{bmatrix},$$

$$M_3 = \left[\begin{smallmatrix} -1 & 0 \\ 0 & 1 \end{smallmatrix}\right], \quad H_3 = \left[\begin{smallmatrix} -1 & 0 & 0 \\ 0 & 0 & 1 \end{smallmatrix}\right].$$

Let $\hat{h}_2^3$ (resp. $\hat{h}_7^3$) be the impulse response from the input u_2 (resp. u_7) to $\hat{r}_3$, and let u_2^1 (resp. u_7^1) denote the input signal u_2 (resp. u_7) up to time 1. Note that the misbehaving agent can be identified after 2 time steps, and that the residual associated with the agent 3 is

$$\hat{r}_3(2) = \left[\begin{smallmatrix} H_3C_1 & M_3 \end{smallmatrix}\right] \left[\begin{smallmatrix} A_d+\varepsilon\Delta & 0 \\ E_3C_1 & F_3 \end{smallmatrix}\right]^2 \left[\begin{smallmatrix} x(0) \\ 0 \end{smallmatrix}\right] + \hat{h}_2^3 \star u_2^1 + \hat{h}_7^3 \star u_7^1,$$

where $\star$ denotes the convolution operator. After some computation we obtain

$$\hat{r}_3(2) = \varepsilon \left[\begin{smallmatrix} H_3C_1 & M_3 \end{smallmatrix}\right] \begin{bmatrix} A_d\Delta+\Delta A_d+\varepsilon\Delta^2 & \Delta B_2 & \Delta B_7 \\ E_3C_1\Delta & 0 & 0 \end{bmatrix} \begin{bmatrix} x(0) \\ u_2(0) \\ u_7(0) \end{bmatrix}$$

and, analogously,

$$\hat{r}_2(2) = \varepsilon \left[\begin{smallmatrix} H_2C_1 & M_2 \end{smallmatrix}\right] \begin{bmatrix} A_d\Delta+\Delta A_d+\varepsilon\Delta^2 & \Delta B_2 & \Delta B_7 \\ E_2C_1\Delta & 0 & 0 \end{bmatrix} \begin{bmatrix} x(0) \\ u_2(0) \\ u_7(0) \end{bmatrix}$$

$$+ \left[\begin{smallmatrix} H_2C_1 & M_2 \end{smallmatrix}\right] \left[\begin{smallmatrix} A_dB_2 & B_2 \\ E_2C_1B_2 & 0 \end{smallmatrix}\right] \begin{bmatrix} u_2(0) \\ u_2(1) \end{bmatrix}$$

Recall that the agent 1 is able to identify the misbehaving agent 2 if, independent of u_2^1 and u_7^1, there exists a threshold T such that $\|\hat{r}_2(2)\|_\infty \geq T$, and $\|\hat{r}_3(2)\|_\infty < T$. The behavior of $\|\hat{r}_2(2)\|_\infty$ and $\|\hat{r}_3(2)\|_\infty$ as a function of ε is in Fig. 8.7. Note that for $\varepsilon = \varepsilon^* = 0.026$ we have $\|\hat{r}_2(2)\|_\infty = \|\hat{r}_3(2)\|_\infty = 0.07$. For instance, if

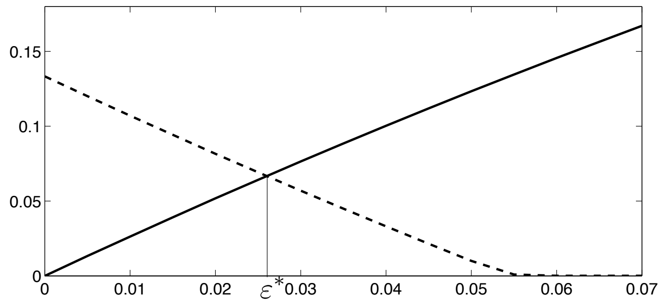


Figure 8.7: In this figure, the solid line corresponds to the largest magnitude of the residual associated with the well-behaving agent 3, while the dashed line denotes the smallest magnitude of the residual associated with the misbehaving agent 2, both as a function of the parameter ε. If $\varepsilon \leq \varepsilon^*$, then there exists a threshold that allows to identify the misbehaving agent 2.

$\varepsilon = 0.01$, then it can be verified that $\|\hat{r}_2(2)\|_\infty > 0.1$, and $\|\hat{r}_3(2)\|_\infty < 0.05$. It follows that a threshold $T = 0.1$ allows the agent 1 to identify the misbehaving agent 2. On the other hand, if $\varepsilon = 0.03$, then $\|\hat{r}_2(2)\|_\infty \geq 0.01$, and $\|\hat{r}_3(2)\|_\infty \leq 0.12$, so that the misbehaving agent 2 may remain unidentified. Indeed, if $x(0) = \left[\begin{smallmatrix} 1 & 1 & 1 & -1 & -1 & -1 & -1 \end{smallmatrix}\right]$, $u_2^1 = u_7^1 = \left[\begin{smallmatrix} 0.1 & 0.1 \end{smallmatrix}\right]$, then $\|\hat{r}_2(2)\|_\infty = 0.01$ and $\|\hat{r}_3(2)\|_\infty = 0.12$, so that the agent 3 is recognized as misbehaving instead of the agent 2.

As a final remark, note that the larger the consensus network, the more convenient the proposed approximation procedure becomes. For instance, consider the network presented in [10], and here reported in Fig. 8.8. Such a clustered interconnection structure, in which the edges connecting different clusters have a small weight, may be preferable in many applications because much simpler and efficient protocols can be implemented within each cluster. Assume the presence

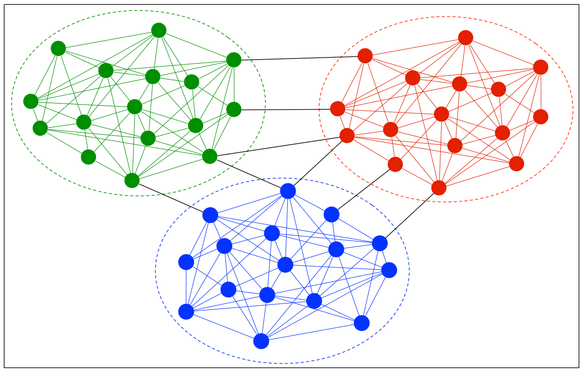


Figure 8.8: A consensus network partitioned into 3 areas. Each agent identifies the neighboring misbehaving agents by only knowing the topology of the subnetwork it belongs to.

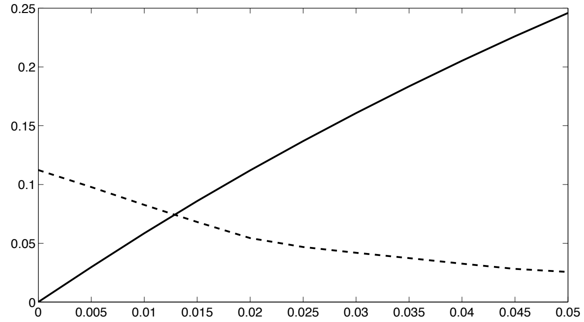


Figure 8.9: For the network in Fig. 8.8, the smallest magnitude of the residual associated with a misbehaving agent (dashed line) and the largest magnitude of the residual associated with a well-behaving agent (solid line) are plotted as a function of ε.

of a misbehaving agent in each cluster, and consider the residuals computed after 5 steps of the consensus algorithm. Let ε be the weight of the edges connecting different clusters. Fig. 8.9 shows, as a function of ε, the smallest magnitude of the residual associated with a misbehaving agent (dashed line) versus the largest magnitude of the residual associated with a well-behaving agent (solid line). If ε is sufficiently small, then our local identification method allows each well-behaving agent to promptly detect and identify the misbehaving agents belonging to the same group, and hence to restore the functionality of the network. For instance, if $\varepsilon \leq 0.01$, then, following Theorem 8.5.3, if the misbehaving input take value in $\{u : 0.1 \leq |u| \leq 3\}$, then a misbehaving agent is correctly detected and identified by a well-behaving agent.

Chapter 9

Conclusion and Future Work

One fundamental challenge for modern cyber-physical systems is to ensure a correct and reliable functionality in the face of failures and attacks. As recently highlighted by the Maroochy water breach [101] in March 2000, multiple recent power blackouts in Brazil [22], the SQL Slammer worm attack on the Davis-Besse nuclear plant in January 2003 [49], the StuxNet computer worm [33] in June 2010, and by various industrial security incidents [90], current security systems are not offering an adequate protection. This thesis has focused on (i) developing a comprehensive mathematical framework for the analysis of control systems vulnerabilities, (ii) designing advanced centralized and distributed monitors for attack detection and isolation, and (iii) constructing attack signals to avoid detection and identification. Our results have been rigorously constructed by relying on tools from geometric control theory, algebraic graph theory, and distributed computation. Finally, several illustrative examples show the effectiveness of our methods for several control systems.

9.1 Summary

In Chapter 3 we present several cyber-physical system, which are used in the subsequent chapters to illustrate our findings. In particular, we derive a mathematical model for power networks, water networks, and consensus networks.

In Chapter 4 we describe our modeling framework for cyber-physical systems, attacks, and monitors. In particular, we model cyber-physical systems under attack as linear descriptor systems subject to unknown inputs. We derive fundamental limitations for several classes of monitors. Finally we characterize system-theoretic and graph-theoretic conditions for the existence of undetectable and unidentifiable attacks for cyber-physical systems. We illustrate our techniques on examples of power and water networks.

In Chapters 5 and 6 we design monitors for attack detection and identification. In particular, Chapter 5 contains a distributed *static* monitor, which has proven useful for state estimation and false data detection in power networks. Chapter 6, instead, contains centralized and distributed *dynamic* monitors. These dynamic monitors outperform the static counterpart, and, in fact, are shown to achieve optimal detection and identification performance. These monitors have been successfully implemented for power networks, water networks, and sensor networks.

Chapter 7 contains our method to cast undetectable and unidentifiable attacks against cyber-physical systems. We show the effectiveness of this method to design attacks in a competitive power generation scenario.

Finally, in Chapter 8 we specialize some of our results to linear consensus systems. we provide alternative and constructive system-theoretic proofs of existing bounds on the number of identifiable misbehaving agents in a linear network, i.e., k Byzantine agents can be detected and identified if the network is $(2k+1)$-connected, and they cannot be identified if the network is $2k$-connected or less. We characterize the effect of specific attacks on the final consensus value, and we analyze the case of non-colluding (faulty) agents. We conclude the chapter by designing a distributed monitor based on a notion of network decentralization.

9.2 Directions for Future Research

In this Thesis we have studied various security issues for cyber-physical systems. Based on our models, we have characterized systems vulnerabilities, designed monitors, and cast undetectable and unidentifiable attacks. However, while this research has solved many security problems for cyber-physical systems, it has raised new questions. We next discuss some aspects requiring future investigation.

Nonlinearities and noisy dynamics. In our analysis we focused on cyber-physical systems described by linear descriptor systems. It is often the case, however, that complex systems obey nonlinear dynamics, and are affected by state and measurement noise. Although our main findings should apply despite nonlinearities and noise, it is an interesting direction to develop mathematical tools for the security assessment of nonlinear models driven by noise. We envision that the theory developed in [42, 82] may be useful for this extension.

Attackers with limited capabilities and specific goals. In this thesis we assume that attackers are omniscient and have unlimited computation capabilities. These assumptions allow us to understand fundamental limitations in monitoring and attack design. However, in general attackers have limited capabilities, and they have access only to an approximate model of the system or a part of it. In this situation it is not clear whether and how undetectable attacks can be cast. Preliminary results in this area are discussed in [51, 120].

Attackers with specific goals. We have considered attackers whose goal is to alter the system functionality while avoiding detection or identification. In a more realistic scenario, however, attackers may have specific goals, for instance driving the system to a particular state, or affecting only certain components.

Additionally, attack signals should be compared according to a properly defined notion of cost. In this case, attacks should be designed not only to be undetectable or unidentifiable, but also to achieve the desired goal while minimizing the attack cost. Our preliminary results for this aspect are in Chapter 7.

Efficient identification algorithm. In Chapter 4 we show that the attack identification problem is computationally hard. An interesting research direction consists of developing efficient approximation algorithms for this problem. In Chapter 6 we design a monitor that achieves exact attack identification at a high computational cost, and an efficient monitor for the identification of certain attacks. The problem of identifying output attacks is also studied in [40].

Optimal network for distributed monitors. In Chapters 5, 6, and 8 we have developed distributed monitors for attack detection and identification. For our monitors, we assume the network to be partitioned among geographically deployed control centers, which implement cooperative algorithms. As a matter of fact, the network partition affect the performance of our monitors. Hence, it is of interest to optimally partition the network to improve the performance of our monitors. Preliminary results in network partitioning can be found, among others, in [60, 92, 119].

Bibliography

[1] A. Abur and A. G. Exposito. *Power System State Estimation: Theory and Implementation.* CRC Press, 2004.

[2] S. Amin, A. Cárdenas, and S. Sastry. Safe and secure networked control systems under denial-of-service attacks. In *Hybrid Systems: Computation and Control*, volume 5469, pages 31–45, Apr. 2009.

[3] S. Amin, X. Litrico, S. S. Sastry, and A. M. Bayen. Stealthy deception attacks on water SCADA systems. In *Hybrid Systems: Computation and Control*, pages 161–170, Stockholm, Sweden, Apr. 2010.

[4] G. E. Apostolakis and D. M. Lemon. A screening methodology for the identification and ranking of infrastructure vulnerabilities due to terrorism. *Risk Analysis*, 25(2):361–376, 2005.

[5] Z. Z. Bai and X. Yang. On convergence conditions of waveform relaxation methods for linear differential-algebraic equations. *Journal of Computational and Applied Mathematics*, 235(8):2790–2804, 2011.

[6] G. Basile and G. Marro. *Controlled and Conditioned Invariants in Linear System Theory.* Prentice Hall, 1991.

[7] M. Basseville and I. V. Nikiforov. *Detection of Abrupt Changes: Theory and Application.* Prentice Hall, 1993.

[8] F. J. Bejarano, T. Floquet, W. Perruquetti, and G. Zheng. Observability and detectability analysis of singular linear systems with unknown inputs. In *IEEE Conf. on Decision and Control and European Control Conference*, pages 4005–4010, Orlando, FL, USA, Dec. 2011.

[9] D. S. Bernstein. *Matrix Mathematics.* Princeton University Press, 2 edition, 2009.

[10] E. Bıyık and M. Arcak. Area aggregation and time-scale modeling for sparse nonlinear networks. *Systems & Control Letters*, 57(2):142–149, 2007.

[11] P. F. Boulos, K. E. Lansey, and B. W. Karney. *Comprehensive Water Distribution Systems Analysis Handbook for Engineers and Planners*. American Water Works Association, 2006.

[12] F. Bullo, J. Cortés, and S. Martínez. *Distributed Control of Robotic Networks*. Applied Mathematics Series. Princeton University Press, 2009. Available at http://www.coordinationbook.info.

[13] J. Burgschweiger, B. Gnädig, and M. C. Steinbach. Optimization models for operative planning in drinking water networks. *Optimization and Engineering*, 10(1):43–73, 2009.

[14] E. J. Candes and T. Tao. Decoding by linear programming. *IEEE Transactions on Information Theory*, 51(12):4203–4215, 2005.

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

[16] A. A. Cárdenas, S. Amin, B. Sinopoli, A. Giani, A. A. Perrig, and S. S. Sastry. Challenges for securing cyber physical systems. In *Workshop on Future Directions in Cyber-physical Systems Security*, Newark, NJ, USA, July 2009.

[17] R. Carli, A. Chiuso, L. Schenato, and S. Zampieri. Distributed Kalman filtering based on consensus strategies. *IEEE Journal on Selected Areas in Communications*, 26(4):622–633, 2008.

[18] F. S. Cattivelli, C. G. Lopes, and A. H. Sayed. Diffusion recursive least-squares for distributed estimation over adaptive networks. *IEEE Transactions on Signal Processing*, 56(5):1865–1877, 2008.

[19] F. S. Cattivelli and A. H. Sayed. Diffusion strategies for distributed Kalman filtering and smoothing. *IEEE Transactions on Automatic Control*, 55(9):2069–2084, 2010.

[20] Y. Censor. Row-action methods for huge and sparse systems and their applications. *SIAM Review*, 23(4):444–466, 1981.

[21] J. H. Chow, J. Cullum, and R. A. Willoughby. A sparsity-based technique for identifying slow-coherent areas in large power systems. *IEEE Transactions on Power Apparatus and Systems*, PAS-103(3):463–473, 1984.

[22] J. P. Conti. The day the samba stopped. *Engineering Technology*, 5(4):46–47, 06 March - 26 March, 2010.

[23] M. L. Crow and M. D. Ilić. The waveform relaxation method for systems of differential/algebraic equations. *Mathematical and Computer Modelling*, 19(12):67–84, 1994.

[24] L. Dai. *Singular Control Systems*. Springer, 1989.

[25] G. Dan and H. Sandberg. Stealth attacks and protection schemes for state estimators in power systems. In *IEEE Int. Conf. on Smart Grid Communications*, pages 214–219, Gaithersburg, MD, USA, Oct. 2010.

[26] C. L. DeMarco, J. V. Sariashkar, and F. Alvarado. The potential for malicious control in a competitive power systems environment. In *IEEE Int. Conf. on Control Applications*, pages 462–467, Dearborn, MI, USA, 1996.

[27] S. Demko, W. F. Moss, and P. W. Smith. Decay rates for inverses of band matrices. *Mathematics of Computation*, 43(168):491–499, 1984.

[28] S. X. Ding. *Model-Based Fault Diagnosis Techniques: Design Schemes, Algorithms, and Tools*. Springer, 2008.

[29] J. M. Dion, C. Commault, and J. van der Woude. Generic properties and control of linear structured systems: a survey. *Automatica*, 39(7):1125–1144, 2003.

[30] D. Dolev. The Byzantine generals strike again. *Journal of Algorithms*, 3:14–30, 1982.

[31] F. Dörfler and F. Bullo. Kron reduction of graphs with applications to electrical networks. *IEEE Transactions on Circuits and Systems*, Nov. 2011. To appear.

[32] D. G. Eliades and M. M. Polycarpou. A fault diagnosis and security framework for water systems. *IEEE Transactions on Control Systems Technology*, 18(6):1254–1265, 2010.

[33] J. P. Farwell and R. Rohozinski. Stuxnet and the future of cyber war. *Survival*, 53(1):23–40, 2011.

[34] M. R. Garey and D. S. Johnson. *Computers and Intractability.* Springer, 1979.

[35] T. Geerts. Invariant subspaces and invertibility properties for singular systems: The general case. *Linear Algebra and its Applications*, 183:61–88, 1993.

[36] C. D. Godsil and G. F. Royle. *Algebraic Graph Theory*, volume 207 of *Graduate Texts in Mathematics.* Springer, 2001.

[37] G. H. Golub and C. F. van Loan. *Matrix Computations.* Johns Hopkins University Press, 2 edition, 1989.

[38] R. Gordon, R. Bender, and G. T. Herman. Algebraic reconstruction techniques (ART) for three-dimensional electron microscopy and x-ray photography. *Journal of Theoretical Biology*, 29(3):471–481, 1970.

[39] C. Grigg, P. Wong, P. Albrecht, R. Allan, M. Bhavaraju, R. Billinton, Q. Chen, C. Fong, S. Haddad, S. Kuruganty, W. Li, R. Mukerji, D. Patton, N. Rau, D. Reppen, A. Schneider, M. Shahidehpour, and C. Singh. The IEEE Reliability Test System - 1996. A report prepared by the Reliability Test System Task Force of the Application of Probability Methods Subcommittee. *IEEE Transactions on Power Systems*, 14(3):1010–1020, 1999.

[40] F. Hamza, P. Tabuada, and S. Diggavi. Secure state-estimation for dynamical systems under active adversaries. In *Allerton Conf. on Communications, Control and Computing*, Sept. 2011.

[41] K. D. Ikramov. Matrix pencils: Theory, applications, and numerical methods. *Journal of Mathematical Sciences*, 64(2):783–853, 1993.

[42] A. Isidori. *Nonlinear Control Systems.* Communications and Control Engineering Series. Springer, 3 edition, 1995.

[43] A. Jadbabaie, J. Lin, and A. S. Morse. Coordination of groups of mobile autonomous agents using nearest neighbor rules. *IEEE Transactions on Automatic Control*, 48(6):988–1001, 2003.

[44] S. Kaczmarz. Angenäherte Auflösung von Systemen linearer Gleichungen. *Bull. Acad. Polon. Sci. Lett. A*, 35:355–357, 1937.

[45] A. N. Kolmogorov and S. V. Fomin. *Introductory Real Analysis.* Dover Publications, 1975.

[46] A. Kumar and P. Daoutidis. *Control of Nonlinear Differential Algebraic Equation Systems*. CRC Press, 1999.

[47] P. Kundur. *Power System Stability and Control.* McGraw-Hill, 1994.

[48] P. Kunkel and V. Mehrmann. *Differential-Algebraic Equations: Analysis and Numerical Solution.* European Mathematical Society, 2006.

[49] S. Kuvshinkova. SQL Slammer worm lessons learned for consideration by the electricity sector. *North American Electric Reliability Council*, 2003.

[50] L. Lamport, R. Shostak, and M. Pease. The Byzantine generals problem. *ACM Transactions on Programming Languages and Systems*, 4(3):382–401, 1982.

[51] H. J. LeBlanc, H. Zhang, S. Sundaram, and X. Koutsoukos. Consensus of multi-agent networks in the presence of adversaries using only local information. In *Proceedings of the 1st international conference on High Confidence Networked Systems*, pages 1–10, 2012.

[52] E. Lelarasmee, A. E. Ruehli, and A. L. Sangiovanni-Vincentelli. The waveform relaxation method for time-domain analysis of large scale integrated circuits. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 1(3):131–145, 1982.

[53] F. L. Lewis. A survey of linear singular systems. *Circuits, Systems, and Signal Processing*, 5(1):3–36, 1986.

[54] F. L. Lewis. Geometric design techniques for observers in singular systems. *Automatica*, 26(2):411–415, 1990.

[55] X. Litrico and V. Fromion. *Modeling and Control of Hydrosystems.* Springer, 2009.

[56] Y. Liu, M. K. Reiter, and P. Ning. False data injection attacks against state estimation in electric power grids. In *ACM Conference on Computer and Communications Security*, pages 21–32, Chicago, IL, USA, Nov. 2009.

[57] C. G. Lopes and A. H. Sayed. Incremental adaptive strategies over distributed networks. *IEEE Transactions on Signal Processing*, 55(8):4064–4077, 2007.

[58] C. G. Lopes and A. H. Sayed. Diffusion least-mean squares over adaptive networks: Formulation and performance analysis. *IEEE Transactions on Signal Processing*, 56(7):3122–3136, 2008.

[59] D. G. Luenberger. *Optimization by Vector Space Methods*. Wiley, 1969.

[60] U. V. Luxburg. A tutorial on spectral clustering. *Statistics and Computing*, 17(4):395–416, 2007.

[61] N. A. Lynch. *Distributed Algorithms*. Morgan Kaufmann, 1997.

[62] M.-A. Massoumnia, G. C. Verghese, and A. S. Willsky. Failure detection and identification. *IEEE Transactions on Automatic Control*, 34(3):316–321, 1989.

[63] A. V. Medvedev and H. T. Toivonen. Feedforward time-delay structures in state estimation-finite memory smoothing and continuous deadbeat observers. *IEE Proceedings. Control Theory & Applications*, 141(2):121–129, 1994.

[64] A. R. Metke and R. L. Ekl. Security technology for smart grid networks. *IEEE Transactions on Smart Grid*, 1(1):99–107, 2010.

[65] C. D. Meyer. *Matrix Analysis and Applied Linear Algebra*. SIAM, 2001.

[66] P. Michiardi and R. Molva. CORE: A collaborative reputation mechanism to enforce node cooperation in mobile ad hoc network. In *Communications and Multimedia Security Conference (CMS)*, Portoroz, Slovenia, Sept. 2002.

[67] H. Minc. *Nonnegative Matrices*. Wiley, 1988.

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

[69] Y. Mo and B. Sinopoli. Secure control against replay attacks. In *Allerton Conf. on Communications, Control and Computing*, pages 911–918, Monticello, IL, USA, Sept. 2010.

[70] A.-H. Mohsenian-Rad and A. Leon-Garcia. Distributed internet-based load altering attacks against smart power grids. *IEEE Transactions on Smart Grid*, 2(4):667 –674, 2011.

[71] E. Montijano, S. Martínez, and S. Sagués. De-RANSAC: Robust distributed consensus in sensor networks. *IEEE Transactions on Systems, Man & Cybernetics. Part B: Cybernetics*, May 2010. Submitted.

[72] Y. Ohta, D. Šiljak, and T. Matsumoto. Decentralized control using quasi-block diagonal dominance of transfer function matrices. *IEEE Transactions on Automatic Control*, 31(5):420–430, 1986.

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

[74] R. Olfati-Saber and R. M. Murray. Consensus problems in networks of agents with switching topology and time-delays. *IEEE Transactions on Automatic Control*, 49(9):1520–1533, 2004.

[75] A. Osiadacz. *Simulation and Analysis of Gas Networks.* Gulf Publishing Company, Houston, TX, USA, 1987.

[76] F. Pasqualetti, A. Bicchi, and F. Bullo. Distributed intrusion detection for secure consensus computations. In *IEEE Conf. on Decision and Control*, pages 5594–5599, New Orleans, LA, USA, Dec. 2007.

[77] F. Pasqualetti, A. Bicchi, and F. Bullo. A graph-theoretical characterization of power network vulnerabilities. In *American Control Conference*, pages 3918–3923, San Francisco, CA, USA, June 2011.

[78] F. Pasqualetti, A. Bicchi, and F. Bullo. Consensus computation in unreliable networks: A system theoretic approach. *IEEE Transactions on Automatic Control*, 57(1):90–104, 2012.

[79] F. Pasqualetti, R. Carli, A. Bicchi, and F. Bullo. Distributed estimation and detection under local information. In *IFAC Workshop on Distributed Estimation and Control in Networked Systems*, pages 263–268, Annecy, France, Sept. 2010.

[80] F. Pasqualetti, F. Dörfler, and F. Bullo. Cyber-physical attacks in power networks: Models, fundamental limitations and monitor design. In *IEEE Conf. on Decision and Control and European Control Conference*, pages 2195–2201, Orlando, FL, USA, Dec. 2011.

[81] F. Pasqualetti, F. Dörfler, and F. Bullo. Attack detection and identification in cyber-physical systems. *IEEE Transactions on Automatic Control*, Aug. 2012. Submitted.

[82] C. D. Persis and A. Isidori. A geometric approach to nonlinear fault detection and isolation. *IEEE Transactions on Automatic Control*, 46(6):853–865, 2001.

[83] R. G. Phillips and P. Kokotović. A singular perturbation approach to modeling and control of Markov chains. *IEEE Transactions on Automatic Control*, 26(5):1087–1094, 1981.

[84] M. G. Rabbat and R. D. Nowak. Quantized incremental algorithms for distributed optimization. *IEEE Journal on Selected Areas in Communications*, 23(4):798–808, 2005.

[85] T. Raff and F. Allgöwer. An observer that converges in finite time due to measurement-based state updates. In *IFAC World Congress*, pages 2693–2695, Seoul, Korea, July 2008.

[86] C. Rakpenthai, S. Premrudeepreechacharn, S. Uatrongjit, and N. R. Watson. Measurement placement for power system state estimation using decomposition technique. *Electric Power Systems Research*, 75(1):41–49, 2005.

[87] K. J. Reinschke. *Multivariable Control: A Graph-Theoretic Approach*. Springer, 1988.

[88] K. J. Reinschke. Graph-theoretic approach to symbolic analysis of linear descriptor systems. *Linear Algebra and its Applications*, 197:217–244, 1994.

[89] W. Ren, R. W. Beard, and E. M. Atkins. Information consensus in multivehicle cooperative control: Collective group behavior through local interaction. *IEEE Control Systems Magazine*, 27(2):71–82, 2007.

[90] G. Richards. Hackers vs slackers. *Engineering & Technology*, 3(19):40–43, 2008.

[91] L. A. Rossman. Epanet 2, water distribution system modeling software. Technical report, US Environmental Protection Agency, Water Supply and Water Resources Division, 2000.

[92] T. Sahai, A. Speranzon, and A. Banaszuk. Hearing the clusters in a graph: A distributed algorithm. *CoRR*, abs/0911.4729, 2009.

[93] M. Saif and Y. Guan. Decentralized state estimation in large-scale interconnected dynamical systems. *Automatica*, 28(1):215–219, 1992.

[94] A. H. Sayed and C. G. Lopes. Adaptive processing over distributed networks. *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, E90-A(8):1504–1510, 2007.

[95] I. Schizas, A. Ribeiro, and G. Giannakis. Consensus in ad hoc WSNs with noisy links - Part I: Distributed estimation of deterministic signals. *IEEE Transactions on Signal Processing*, 56(1):350–364, 2007.

[96] I. D. Schizas, G. Mateos, and G. B. Giannakis. Distributed LMS for consensus-based in-network adaptive processing. *IEEE Transactions on Signal Processing*, 57(6):2365–2382, 2009.

[97] E. Scholtz. *Observer-based monitors and distributed wave controllers for electromechanical disturbances in power systems*. PhD thesis, Massachusetts Institute of Technology, 2004.

[98] F. C. Schweppe and J. Wildes. Power system static-state estimation, Part I: Exact model. *IEEE Transactions on Power Apparatus and Systems*, 89(1):120–125, 1970.

[99] F. C. Schweppe and J. Wildes. Power system static-state estimation, Part II: Approximate model. *IEEE Transactions on Power Apparatus and Systems*, 89(1):125–130, 1970.

[100] S. Skogestad and I. Postlethwaite. *Multivariable Feedback Control Analysis and Design*. Wiley, 2 edition, 2005.

[101] J. Slay and M. Miller. Lessons learned from the Maroochy water breach. *Critical Infrastructure Protection*, 253:73–82, 2007.

[102] R. Smith. A decoupled feedback structure for covertly appropriating network control systems. In *IFAC World Congress*, pages 90–95, Milan, Italy, Aug. 2011.

[103] S. Sridhar, A. Hahn, and M. Govindarasu. Cyber–physical system security for the electric power grid. *Proceedings of the IEEE*, 99(1):1–15, 2012.

[104] S. S. Stankovic, M. S. Stankovic, and D. M. Stipanovic. Consensus based overlapping decentralized estimation with missing observations and communication faults. *Automatica*, 45(6):1397–1406, 2009.

[105] S. Sundaram and C. Hadjicostis. Distributed function calculation via linear iterative strategies in the presence of malicious agents. *IEEE Transactions on Automatic Control*, 56(7):1495–1508, 2011.

[106] S. Sundaram and C. N. Hadjicostis. Distributed function calculation via linear iterations in the presence of malicious - Part II: Overcoming malicious behavior. In *American Control Conference*, pages 1356–1361, Seattle, WA, June 2008.

[107] S. Sundaram and C. N. Hadjicostis. Distributed function calculation via linear iterations in the presence of malicious agents - Part I: Attacking the network. In *American Control Conference*, pages 1350–1355, Seattle, WA, June 2008.

[108] K. Tanabe. Projection method for solving a singular system of linear equations and its applications. *Numerische Mathematik*, 17(3):203–214, 1971.

[109] K. Tchoń. On generic properties of linear systems: An overview. *Kybernetika*, 19(6):467–474, 1983.

[110] A. Teixeira, S. Amin, H. Sandberg, K. H. Johansson, and S. Sastry. Cyber security analysis of state estimators in electric power systems. In *IEEE Conf. on Decision and Control*, pages 5991–5998, Atlanta, GA, USA, Dec. 2010.

[111] J. Tokarzewski. *Finite Zeros in Discrete Time Control Systems.* Lecture notes in control and information sciences. Springer, 2006.

[112] H. L. Trentelman, A. Stoorvogel, and M. Hautus. *Control Theory for Linear Systems.* Springer, 2001.

[113] D. J. Trudnowski, J. R. Smith, T. A. Short, and D. A. Pierre. An application of Prony methods in PSS design for multimachine systems. *IEEE Transactions on Power Systems*, 6(1):118–126, 1991.

[114] J. van der Woude. The generic number of invariant zeros of a structured linear system. *SIAM Journal on Control and Optimization*, 38(1):1–21, 1999.

[115] J. W. van der Woude. A graph-theoretic characterization for the rank of the transfer matrix of a structured system. *Mathematics of Control, Signals and Systems*, 4(1):33–40, 1991.

[116] M. Vidyasagar. *Input-Output Analysis of Large-Scale Interconnected Systems: Decomposition, Well-Posedness and Stability.* Springer, 1981.

[117] W. M. Wonham. *Linear Multivariable Control: A Geometric Approach.* Springer, 3 edition, 1985.

[118] L. Xiao and S. Boyd. Fast linear iterations for distributed averaging. *Systems & Control Letters*, 53:65–78, 2004.

[119] J. Zaborszky, K. W. Whang, G. Huang, L. J. Chiang, and S. Y. Lin. A clustered dynamic model for a class of linear autonomous systems unsing simple enumerative sorting. *IEEE Transactions on Circuits and Systems*, 29(11):747–758, 1982.

[120] H. Zhang and S. Sundaram. Robustness of information diffusion algorithms to locally bounded adversaries. In *American Control Conference*, 2012. To appear.

[121] M. Zhu and S. Martínez. Stackelberg-game analysis of correlated attacks in cyber-physical systems. In *American Control Conference*, pages 4063–4068, San Francisco, CA, USA, July 2011.

[122] R. D. Zimmerman, C. E. Murillo-Sánchez, and D. Gan. MATPOWER: Steady-state operations, planning, and analysis tools for power systems research and education. *IEEE Transactions on Power Systems*, 26(1):12–19, 2011.