Active Detection for Resilient Cyber-Physical Systems

Submitted in partial fulfillment of the requirements for

the degree of

Doctor of Philosophy

in

Electrical and Computer Engineering

Sean C. Weerakkody

B.S., Electrical Engineering, University of Maryland, College Park
B.S., Mathematics, University of Maryland, College Park

Carnegie Mellon University
Pittsburgh, PA

December, 2018

Acknowledgements

The contributions made in this thesis would not have been possible without the aid and support of numerous people and organizations. First, I would like to thank my advisor and chair of my thesis committee Bruno Sinopoli. From the time Bruno unknowingly called me on my 22nd birthday to discuss the opportunity to attend Carnegie Mellon to the final months of my graduate studies, Bruno has always offered me encouragement and advice. He provided me the freedom to find my own research path and the guidance to make sure I never strayed too far. I appreciate everything he has taught me both in my graduate studies and in my personal life. I could not have asked for a better advisor.

Next I would like to thank the rest of my thesis committee Anupam Datta, Geir Dullerud, Soummya Kar, and George Pappas. They have all had a great involvement and impact on my graduate studies. I had met Professor Pappas as an undergraduate student visiting the University of Pennsylvania. He was kind enough to schedule time to meet with me and arrange for me to see the GRASP Lab. I am extremely grateful for his presence, advice, and support as a member of my thesis committee. Professor Dullerud was generous enough to help me arrange a visit to the Coordinated Science Lab at the University of Illinois and meet with me in person. I have enjoyed our multiple research discussions and appreciate the time he has taken to participate in my defense. Professor Kar has been a constant positive presence in my graduate career, as a Professor of the first class I took at Carnegie Mellon, a member of my qualifying committee, a collaborator, and now a member of my defense committee. I thank him for all his help and guidance and for participating in my committee. Last but not least, Professor Datta has also played a significant role in my graduate studies. Like Professor Kar, I have known him as a professor, a collaborator, and a member of my qualifying committee. I have always appreciated his unique vantage point and I am thankful for his presence on my committee.

I would like to thank everyone in the B level of Porter Hall. I especially would like to thank Yilin Mo and Sergio Pequito. The guidance they volunteered to me during the beginning of my

PhD was instrumental in setting me on the right path. I would not have developed the confidence I needed as a researcher without them. I would like to thank Rohan Chabukswar, Xiaoqi Yin, Mihovil Bartulovic, and Omur Ozel for bearing with my presence and the number of used water bottles in Porter B23. I would like to also thank the rest of the members of Bruno's group including John Costanzo, Paul Griffioen, Xiaofei Liu, Niranjini Rajagopal, Steven Aday, Sabina Zejnilovic, Dragana Bajovic, Lucas Balthazar, Raffaele Romagnoli, Nicola Forti, Elias Bou-Harb, and Walter Lucia. To all of you, your feedback, collaboration, and friendship has been invaluable. I also would like to thank Claire Bauerle for all the administrative work she did to help me behind the scenes and for always managing to handle my reimbursements.

Finally, I would like to thank my family. My parents Sunil and Maya Weerakkody and sister Tanya Weerakkody have been a constant source of emotional support for me throughout my PhD. When I did not believe in myself they continued to believe in me. I knew regardless of the hour I could always count on any of them to lend a ear or provide advice. I would like to thank my parents for all the trips they made to see me in Pittsburgh and my mom in particular for making sure that I ate reasonably healthy during my PhD with the never-ending supply of frozen home cooked meals. Despite their lack of familiarity with the subject matter, I strongly feel that this PhD is as much theirs as it is mine.

Of course, this research would not be possible without the financial support of grants and other sponsors. I am grateful for receiving the Benjamin Garver Lamme/Westinghouse Graduate Fellowship, which provided support for me during the Fall of 2012. In addition, I would like to thank the Department of Defense (DoD) for supporting me through the National Defense Science & Engineering Graduate Fellowship (NDSEG) Program from the summer of 2014 to the summer of 2017. Finally, I would like to thank the Siebel foundation for awarding me the Siebel Scholarship in Energy Science in 2018. In addition I would like to thank the National Science Foundation for supporting my research through grants 0955111 CAREER: Efficient, Secure and Robust Control of Cyber-Physical Systems, 1135895 CPS: Medium: Collaborative Research: The Cyber-Physical Challenges of Transient Stability and Security in Power Grids, CNS-1329936 CPS: Synergy:

Collaborative Research: Event-Based Information Acquisition, Learning, and Control in High-Dimensional Cyber-Physical Systems, and 1646526: CPS: Synergy: Information Flow Analysis for Cyber-Physical System Security. I would like to thank the Department of Energy for supporting our research through award DE-OE0000779. Finally, I would like to thank Army Research Office Foundation and Cylab for supporting my work through grant DAAD19-02-1-0389. .

Abstract

Cyber-physical systems (CPS) face tremendous threats in modern society. Indeed their presence in critical infrastructures such as transportation, energy delivery, and health care make such systems a target of malevolent entities while their complexity, connectivity, and heterogeneity offer surfaces for attackers to leverage. One important aim of potential attackers is to remain stealthy. An attacker that avoids detection is able to disrupt CPS for long periods of time, without having to worry about defender interference, allowing an adversary to potentially maximize their impact. Intelligent attackers can leverage their system knowledge, disruption resources, and disclosure resources to impart critical damage to systems, all the while remaining stealthy.

In this dissertation we consider the development of active methods to detect intelligent, powerful, and malicious adversaries in cyber-physical systems. While standard attack detection involves producing intelligent algorithms to process information about a system, active detection involves the intelligent design and modification of the inputs, parameters, and structure of a system in order to impede an adversary's ability to generate stealthy attacks. This thesis will propose several methods for active detection in cyber-physical systems.

We will first consider the design of secret random perturbations at the control input, which we term as physical watermarking. We will evaluate this approach against both replay attacks and model aware adversaries. Next, we will consider how naturally occurring stochastic phenomena in a CPS can be utilized for the purposes of active detection. Specifically, we will evaluate how packet drops at the control input can act as an environmental watermark for the benefit of security. Then, we will consider how changing parameters of the plant itself can be used to thwart otherwise model aware attackers. We term this the moving target approach. Two designs are explored. We will consider a switched system model where parameters of the plant are directly changed. Alternatively, we evaluate an authenticating subsystem model where we use an extended system to detect attacks on the CPS under consideration. The moving target involves online changes to the system. Instead, we can consider robust offline design. In particular, we use structural system

theory to analyze and design distributed control systems, which can not be targeted by a class of stealthy attacks. To conclude, motivated by studies in software security, we explore how tools of information flow analysis can be used for the analysis and design of active detection techniques.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Cyber-physical systems (CPS) are computationally capable systems that directly interact with a physical environment and allow people to intelligently and efficiently manage physical processes. CPS are the foundation of key infrastructures such as the smart grid, water distribution systems, and waste management. Their role in transportation, smart buildings, and medical technologies are also burgeoning as new application areas are discovered.

CPS are enabled by technologies which perform sensing, computing, and communication. For example, CPS leverage sensing technologies to gather relevant data about physical systems. In transportation this could for instance be the position and velocity of vehicles. Alternatively, in medical technologies, this may be the heart rate or blood pressure of a patient. Combined with a mathematical model of a system's physical dynamics, sensing can enable accurate state estimation and prediction. This in turn allows the monitoring of physical processes. Sensing technologies have significantly improved. Systems can be sampled more frequently and with less delay. Additionally, sensing devices are in many cases cheap and economically viable.

In addition to monitoring physical processes, it is typically desirable to physically manipulate a system to achieve some objective. In a waste management system, a relevant task would be to treat and purify the wastewater. Alternatively, in smart buildings we wish to regulate the environment (i.e. using HVAC systems) in an energy efficient manner. Cyber-physical systems allow us in many

cases to automate this process using computing technologies. The intelligent control of physical systems is generally a time sensitive task. Thus, a key to incorporating CPS is improvement in the processing speed of our computers. Today, programmable logic controllers (PLCs) and microcontrollers are able to quickly process sensory information and automatically implement an intelligent algorithm for control. The speed at which this can be done has allowed humans to explore new frontiers. As an example, the ability to safely incorporate safe driving cars to transportation systems is in part a result of the vast computational abilities of the embedded systems in today's vehicles.

Finally, a sophisticated communication infrastructure allows operators to control cyber-physical systems remotely while also enabling them to reliably control large scale systems. Many systems have transitioned from wired to wireless communication technologies, which allows for ease of maintenance and installation, lower costs, as well as automation in geographically disparate systems. As an example, wireless communication technologies play a major role in supervisory control and data acquisition (SCADA) systems, see, e.g., [1]. A SCADA system is a hierarchical system, which enables the supervisory management of a control system. The lowest layer consists of field devices such as sensors and actuators, which directly interact with the physical environment. Remote terminal units (RTUs) and PLCs are often used to implement autonomous local control. These units typically interface with both field devices such as pumps, valves, and switches as well as a centralized supervisory control layer which monitors the system. SCADA systems are regularly seen in the smart grid as well as water distribution and waste management systems.

Unfortunately, CPS have become a target of malicious attacks. Indeed, there exists ample motivation to target cyber-physical systems because they are linked to our critical infrastructures such as energy delivery, transportation, and health care. Economically driven adversaries can wage attacks for instance to obtain an advantage in the electricity market [2] or improve fuel mileage while driving [3]. On the other hand, truly malicious actors such as terrorists can pursue attacks that lead to widespread damages, the disruption of critical services, and potentially the loss of life.

There has been a precedence for attacks. One example is Stuxnet, a malware that attacked

uranium enrichment facilities in Iran, causing damage to a 1000 centrifuges at these plants [4]. Stuxnet was able to spread across networks using USBs and shared printers. In addition to leveraging two stolen certificates from chip manufacturers, four zero day exploits, and a PLC rootkit, the malware was able to avoid detection for long periods of time by using a replay attack [5]. In particular, infected devices sent prior measurements to the SCADA system, which were collected during normal operation. This prevented attacks varying the gas pressure and rotational speeds of centrifuges from being recognized. Another prominent attack was the Maroochy Shire incident in Queensland, Australia [6]. Here, a disgruntled former employee was able to hack a SCADA system performing waste management, causing millions of gallons of sewage to leak. The presence of a malicious insider with a fundamental understanding of the system posed a significant challenge for operators. Finally, another famous attack was the Ukraine power attack in December of 2015 [7, 8]. Here, attackers were able to harvest valid credentials at a control center by using the BlackEnergy malware to infect SCADA systems. The attackers then used their access to hack workstations and remotely trip circuit breakers. Additionally, the KillDisk malware destroyed data at the control center while a telephone denial of service was used to cut off communication between customers and providers.

Finally, there still remains ample opportunity for adversaries. Increased automation and improved sensing have allowed system designers to remotely monitor and control critical infrastructures. The ability to perform sensing and control over a communication network creates the opportunity for an attacker to cause damage via network intrusions. Adversaries can find weaknesses in protocols including DNP3 and the older Modbus protocol or leverage poorly designed firewalls to penetrate the network. Attackers can also target trusted peer utility links. For instance, adversaries can attempt to hijack VPN connections. Alternatively, adversaries can steal valid credentials allowing them unencumbered remote access to perform the same actions as a trusted user as in the Ukraine power attack [7].

We remark that CPS operation relies on the use of small heterogeneous components and devices that are potentially prone to failure or attack. For example, in the Stuxnet attack, introducing infected

USB devices into CPS allowed this malware to spread. Meanwhile in the case of the Ukraine power attack, infected email attachments allowed attackers access to system workstations. Attackers can also target field devices such as sensors and actuators as well as networking devices which can interface with both field devices and the monitoring layer. For instance, in SCADA systems, remote terminal units often allow for dial up access and may not even require authentication. An attacker can also take the initiative to introduce vulnerabilities to CPS devices by targeting supply chains. If production is not performed securely, adversaries can install backdoors in components, which can later be leveraged to compromise the CPS.

Beyond attempting to access CPS through a network, an attacker can simply attempt to target the physical plant itself. In many cases, due to the scale of CPS it is impossible to physically monitor and protect all devices and components. As an example, it is often the case that substations as well as smart meters and PMUs are left unattended in the electricity grid. Likewise, it is impractical to guard all the sensors, pumps, and valves in a water distribution system or traffic lights and vehicles in a transportation system. The defender must also account for the actions of malicious insiders. Malicious insiders can leverage their understanding of a CPS and their access to the system in order to target the infrastructure as was done in the Maroochy Shire incident.

Given the ample motivation, opportunity, and precedence for attacks, it is important to design CPS, which preserve the fundamental security properties of secrecy, availability, and integrity. With respect to **secrecy**, the release of sensitive information in CPS can have significant privacy repercussions. In the smart grid, consumers do not want their electricity consumption released, travelers in transportation systems do not want their locations disclosed, and patients receiving health care do not want their medical histories revealed.

Availability is also a critical property in CPS. Jamming or denial of service attacks can be used to restrict the flow of information in a CPS. In a jamming attack, an adversary emits a signal which interferes with the messages being sent between the plant and SCADA operator, preventing the receiving party from obtaining the proper message. In a similar vain, a denial of service attack restricts availability by flooding a system with requests. This can delay or prevent legitimate

requests from being addressed. While the availability of real time data streams is often not critical in typical software systems, in CPS availability of sensor information and control commands may be intrinsically linked to the safe and reliable operation of the underlying control system. In the absence of sensor measurements, a SCADA system fails to monitor the plant. This in turn can prevent an operator from determining proper corrective actions. Likewise, the absence of control commands prevents proper control actions from being delivered to the plant. This results in sub-optimal or possibly unsafe operation. Previous work [9, 10] has shown that open loop unstable systems have critical packet delivery rates that are necessary to ensure that the resulting closed loop system can be stabilized.

While the availability of real time information is important in CPS, these violations are often easily detected. As such, it remains to design systems, which can adequately respond to these attacks when they do occur. Since denial of service and jamming attacks can be easily recognized, the focus of this dissertation, and the remaining discussion will be centered around developing methods that can counter **integrity** attacks.

In an integrity attack, the adversary typically modifies control commands or sensor measurements transmitted in a CPS. Both sensor and control command attacks can be realized through the cyber realm or the physical realm. For example, these integrity attacks can be performed in the cyber space via a man in the middle attack occurring over the network. Here, the adversary can intercept true data packets and replace them with his own falsified data packets before forwarding the resulting message to the operator or plant. Alternatively, a physical sensor integrity attack can occur if an adversary changes the environment around a sensor. For instance, a temperature sensor can be compromised through local heating and cooling. We also note that control commands can be directly modified by an attacker who has physical access to actuators in CPS. Misleading sensory information may cause operators to make incorrect control decisions, which in turn leads to physical damage at the plant. Modified control commands can also cause significant damage, for instance allowing an attacker to cause blackouts on the grid by tripping breakers. If done intelligently, integrity attacks can be performed in a stealthy manner, allowing an adversary to

perturb a system for long periods of time without defender interference. An example of this is a replay attack, which was utilized in Stuxnet.

One may wonder whether existing tools in cyber security are sufficient for countering integrity attacks. One possible option is to consider attack prevention. However, it is often infeasible to remove all access points for potential attackers. The large scale of a CPS means physical protection is often impractical. Additionally, device heterogeneity provides ample entry points for an attacker to leverage while system connectivity allows adversaries to maximize the opportunities they receive. Last but not the least, human error, which could allow corrupted devices to enter a system (as in Stuxnet) or allow malware to infect workstations (as in the Ukraine attack) can not be completely eliminated.

Without the ability to prevent integrity attacks, we must consider mechanisms for response. Cryptographic primitives and protocols can detect integrity attacks. Authentication protocols, for instance, can be used to verify the identity of different devices, components, and operators. Moreover, authenticated encryption simultaneously guarantees secrecy and integrity in attacks from remote adversaries. The root of trust in such a system is a set of secret keys. In public key cryptography each object will have a public key used for encryption and a private key used for decryption while in symmetric key cryptography each pair of communicating objects has a shared key.

However, cryptographic primitives can often be broken or compromised. Moreover, in certain systems, introducing computationally demanding encryption can be costly or impractical for devices that can only support lightweight protocols. In addition, encryption schemes can fail against purely physical attacks. The integrity of sensor measurements can be modified by changing a sensor's local environment while control inputs can be changed by directly manipulating system actuators. In such a scenario, message authentication codes or digital signatures fail to recognize an attack. Furthermore, upon detection of attacks, cyber security is woefully inadequate in providing mechanisms for recovery. For instance, upon detecting an attack, one common countermeasure in cyber security is to take a system offline. However, the inertia of CPS and the need to provide

continued service can make such a decision impractical. Moreover, achieving resilience in CPS requires a defender to design countermeasures that preserve stability and control performance.

As a result, we argue that in order to achieve resilient CPS, it is necessary to augment existing techniques in cyber security. A common approach is to invoke methods from system theory. In particular, the defender's understanding of a system's dynamics can be used to detect, isolate, and respond to attacks. Here, sensor measurements can be used as outputs, which allow a defender to evaluate a system's health. For example, if the sensor measurements closely follow expected behavior as determined by a physical model, a detector can accept the hypothesis that the system is operating normally. However, if the measurements significantly deviate from the model, the detector may determine that their exists faults or malicious behavior in the CPS. Upon attack detection, resilient algorithms for estimation and control can be implemented to allow a CPS to recover.

From an adversarial perspective, generating stealthy attacks is often a desirable outcome. Remaining stealthy allows an adversary to act on a system for long periods of time without a defender's knowledge. This prevents a defender from deploying effective countermeasures, which can otherwise hinder an adversary and limit his or her impact. Unfortunately, the process of attack detection, even when enhanced with system theoretic methods, can fail against clever, knowledgeable, and resourceful adversaries. For example, attackers with a strong understanding of the dynamics of a system can carefully construct attack inputs so that the sensor measurements received by the operator under attack have the same statistics as the sensor measurements that would be received during normal operation.

This threat is amplified by malicious insiders who have a strong understanding of the system as in the Maroochy Shire incident and skilled hackers who are able to modify a significant fraction of a system's inputs and outputs. Detailed system knowledge may not even be a requirement as replay attacks are often provably stealthy. Traditional detection theory, which we refer to as passive detection, alone is provably ineffective against attacker's who are able to generate convincing counterfeit sensor outputs because no tests exist that can differentiate between sequences of outputs

with the same statistics.

We remark, that the defender has additional degrees of freedom beyond the design of statistical algorithms for detection. We argue that a defender can design his control strategy, system parameters, and sensors in a manner that limits or altogether prevents an attacker from constructing stealthy attack sequences. Specifically, in this thesis we consider the the development of **active detection**. Here, active detection refers to the intelligent design of systems and controllers, both offline and online, which force attackers to use strategies that result in outputs that are statistically different from the outputs that are obtained during normal operation. Given active strategies that limit an attacker's ability to generate convincing fabricated outputs, passive detectors can again become an effective tool for recognizing an attacker's presence.

1.1 Contributions

In this thesis, we propose and evaluate several techniques for active detection. We first consider how changing the defender's control strategy can be leveraged to detect attacks. Specifically, we consider the process of physical watermarking, where a noisy additive Gaussian input is introduced on top of the control input, in order to authenticate sensor measurements. We extend prior work, which considers the design of IID Gaussian watermarks by exploring the design of stationary Gaussian watermarks. We demonstrate the effectiveness of this approach in detecting replay attack strategies. Additionally, we explore how physical watermarking can still be effective against certain classes of model aware adversaries with access to a subset of inputs. Here, we propose a robust watermark for attack detection.

Next, we consider how naturally occurring, stochastic phenomena can allow a defender to perform active detection. As an example, we explore how packet drops at the control input in a TCP like system act as an environmental watermark. Specifically, the randomness of packet drops can be used by a defender to detect adversaries who construct outputs which are independent of the packet dropping sequence. We then evaluate the effectiveness of intentionally introducing packet

drops. Moreover, we examine the design of a Gaussian watermark in the presence of a stochastic drop process.

Beyond changing the control strategy, we investigate introducing time varying changes to the parameters of the system. This can thwart a model aware defender that utilizes their knowledge of the plant to construct harmful, stealthy outputs. We refer to this method for active detection as a moving target. In particular, the time varying dynamics act as a moving target for an attacker who attempts to alter their strategy. Two main methods are considered here. The first is an authentication system based moving target. Here, the dynamics of the original plant are unmodified. However, an additional system with time varying dynamics correlated to the original system is introduced. The idea is that if the nominal plant is harmed, this will be reflected in the authenticating subsystem. An attacker will be unable to adapt his attack to counter the changing authenticating subsystem and is detected. We will alternatively consider a separate moving target approach where the dynamics of the original plant are indeed altered. Design recommendations here are provided.

We will also consider how structural changes to the plant can improve the resiliency of cyber-physical systems to stealthy attacks. We will formulate this problem as the design of a distributed control system where the operator has the degree of freedom to change the communication and sensing topology. The goal is to design the system structurally in such a way that a class of stealthy attacks is unfeasible for a resource limited, model aware attacker. Necessary and sufficient conditions for the design of such systems will be provided. In addition, optimization problems with efficient solutions are formulated to design minimal robust control systems.

Finally, we conclude the thesis by proposing an overarching formalism to consider the problem of detection in cyber-physical systems. Here, we are motivated by the notion of information flows in software security. Here, we use information flows as a means to quantify the detectability of different attack strategies as a function of a particular defense strategy. This in turn, will provide operators with helpful guidelines to design systems in order to prevent, detect, or mitigate classes of harmful and stealthy attacks.

The rest of the thesis is formulated as follows. In Chapter 2, we consider the design of physical

watermarks for active detection. In Chapter 3, we evaluate how naturally occurring watermarks, specifically packet drops at the control input, can aid in detection. In Chapter 4, we investigate the moving target approach and study two unique methods for parameter modification. In Chapter 5, we discuss how structural changes to systems can eliminate several classes of stealthy attacks. Chapter 6 introduces information flow as a formalism for considering active detection. Finally, Chapter 7 concludes the thesis.

Chapter 2

Physical Watermarking

In this chapter, we introduce our first technique for active detection, physical watermarking. Here, we will investigate how introducing random perturbations at the control input allow us to detect otherwise stealthy attacks. In section 2.1, we extend prior work in the design of IID Gaussian watermarks by investigating the design of more general stationary Gaussian watermarks. We demonstrate evidence of significant improvement over the IID design. In section 2.2, we consider watermarking against an alternative attacker who violates standard assumptions by having access to a subset of control inputs and leveraging model knowledge. We introduce a robust watermarking design to counteract this attacker. The results in this chapter are largely based on [11] and [12].

2.1 Stationary Gaussian Watermarks

In this section, we introduce the approach of physical watermarking for actively detecting attacks in control systems.

Definition 2.1. *A physical watermark is a secret noisy (random) control input inserted in addition to or in place of an intended control input u_k^* to authenticate the system.*

We will show the approach of physical watermarking is effective in detecting replay attacks.

We note that watermarking is in part motivated by the use of nonces in cyber security described below.

Example: Let us consider the Needham Schroeder protocol in [13], which establishes a session key between 2 users, Alice $\underline{A}$ and Bob $\underline{B}$, by leveraging access to a trusted third party, server $\underline{S}$. In this protocol, Alice shares a session key K_{AB} with Bob by sending $\{K_{AB}, \underline{A}\}_{K_{BS}}$ where K_{BS} is Bob's shared key with $\underline{S}$ and $\{\}_{K^*}$ denotes encryption with key K^*. This message is vulnerable to a replay attack. For instance, suppose Eve $\underline{E}$ recovers an old session key K^*_{AB}. She can replay the message $\{K^*_{AB}, \underline{A}\}_{K_{BS}}$ to Bob. Bob now believes he shares key K^*_{AB} with Alice, when he truly shares a key with Eve. This lets Eve engage in a man in the middle attack.

To counter this attack, Alice receives a nonce or random number, N_B, from Bob encrypted with K_{BS}. After communicating with $\underline{S}$, Alice sends $\{K_{AB}, \underline{A}, N_B\}_{K_{BS}}$ to Bob. The random nonce serves as a challenge to Alice. By including the encrypted nonce in her response to Bob, Alice proves that the message is fresh, and has not been replayed.

2.1.1 System Description

The physical watermarking strategy is given for a class of general control systems. The control system is modeled as a linear, time invariant (LTI) system, the state dynamics of which are given by

$$x_{k+1} = Ax_k + Bu_k + w_k, \tag{2.1}$$

where $x_k \in \mathbb{R}^n$ is the vector of state variables at time k, $u_k \in \mathbb{R}^p$ is the control input, and $w_k \in \mathbb{R}^n$ is the process noise at time k. w_k is assumed to be an IID Gaussian process with $w_k \sim \mathcal{N}(0, Q)$. Since the control system usually operates for an extended period of time, it is assumed that the system starts at time $-\infty$.

A sensor suite monitors the system described in (2.1). At each step, all the sensor readings are collected by a base station. The observation equation can be written as

$$y_k = Cx_k + v_k, \tag{2.2}$$

where $y_k \in \mathbb{R}^m$ is a vector of measurements from the sensors and $v_k \sim \mathcal{N}(0, R)$ is IID measurement noise independent of w_k. It is assumed that (A, B) and $(A, Q^{\frac{1}{2}})$ is stabilizable, (A, C) is detectable, and $R > 0$.

We assume that the system operator wants to minimize the infinite-horizon linear-quadratic-Gaussian (LQG) cost

$$J = \lim_{N\to\infty} \frac{1}{2N+1} \mathbb{E}\left[\sum_{k=-N}^{N} \left(x_k^T W x_k + u_k^T U u_k\right)\right], \tag{2.3}$$

where W, U are positive definite matrices. Since the separation principle holds in this case, the optimal solution of (2.3) is a combination of the Kalman filter and LQG controller [14]. The Kalman filter provides the optimal state estimate $\hat{x}_{k|k}$. Since the system is assumed to start at $-\infty$, the Kalman filter converges to a fixed gain linear estimator

$$\hat{x}_{k+1|k} = A\hat{x}_k + Bu_k,\ \hat{x}_{k|k} = \hat{x}_{k|k-1} + Kz_k. \tag{2.4}$$

where $z_k \triangleq y_k - C\hat{x}_{k|k-1}$ is the residue vector and the Kalman gain K is given by

$$K \triangleq PC^T \left(CPC^T + R\right)^{-1}, \tag{2.5}$$

where P is the solution of the Riccati equation

$$P = APA^T + Q - APC^T(CPC^T + R)^{-1}CPA^T. \tag{2.6}$$

The estimation error at time k is defined to be $e_k = x_k - \hat{x}_{k|k}$.

The LQG controller is a fixed gain linear controller based on the optimal state estimate $\hat{x}_{k|k}$. Specifically,

$$u_k^* = L\hat{x}_{k|k}, \tag{2.7}$$

where u_k^* is the optimal control input. The control gain matrix L is defined to be

$$L \triangleq -\left(B^T S B + U\right)^{-1} B^T S A, \tag{2.8}$$

where S satisfies the Riccati equation

$$S = A^T S A + W - A^T S B \left(B^T S B + U\right)^{-1} B^T S A. \tag{2.9}$$

Consider the case where, instead of directly applying the optimal LQG control u_k^* to the physical system, a physical watermarking scheme is used, in which the true control input u_k is given by

$$u_k = u_k^* + \Delta u_k, \tag{2.10}$$

where u_k^* is the optimal LQG control and Δu_k is the watermark signal. Physical watermarking was first introduced in [15] as an IID additive input sequence $\Delta u_k \sim \mathcal{N}(0, \mathcal{J})$ introduced on top of an optimal control sequence u_k^*. It is assumed that the adversary can not read the defender's control input u_k or watermark Δu_k in this scenario. In particular, the control input will serve as a secret in this approach for active detection. The watermarks act as a cyber-physical nonce. Under normal conditions, the watermark will be embedded in the sensor outputs due to the system dynamics, a valid response to the defender's challenge. However, under replay attack, the measurements contain physical responses to an earlier sequence of watermarks. Unable to detect recent watermarks in the sensor outputs, the defender can not verify freshness of the received sensor measurements. As a result, a passive detector can be designed to distinguish normal system behavior from a replay attack.

The process of physical watermarking is pictorially illustrated in Fig. 2.1. The first images represents the CPS with an optimal control input. A watermark (the second image) is embedded in the control input resulting in a noisy output (the third image). The defender designs a detector that allows him to recognize the presence of the watermark in the sensor outputs.

The watermark signal $\{\Delta u_k\}$ is assumed to be a $p-$dimensional stationary zero-mean Gaussian process independent from the noise processes $\{w_k\}$, $\{v_k\}$. Define the autocovariance function $\Gamma : \mathbb{Z} \to \mathbb{R}^{p\times p}$ to be

$$\Gamma(d) \triangleq \mathrm{Cov}(\Delta u_0, \Delta u_d) = \mathbb{E}[\Delta u_0 \Delta u_d^T]. \tag{2.11}$$

In this section, the watermark is assumed to be generated by a Hidden-Markov Model (HMM)

$$\xi_{k+1} = A_\omega \xi_k + \psi_k, \;\; \Delta u_k = C_h \xi_k, \tag{2.12}$$

where $\psi_k \in \mathbb{R}^{n_h}$, $k \in \mathbb{Z}$ is a sequence of IID zero-mean Gaussian random variables with covariance Ψ, and $\xi_k \in \mathbb{R}^{n_h}$ is the hidden state. To make $\{\Delta u_k\}$ a stationary process, the covariance of ξ_0 is

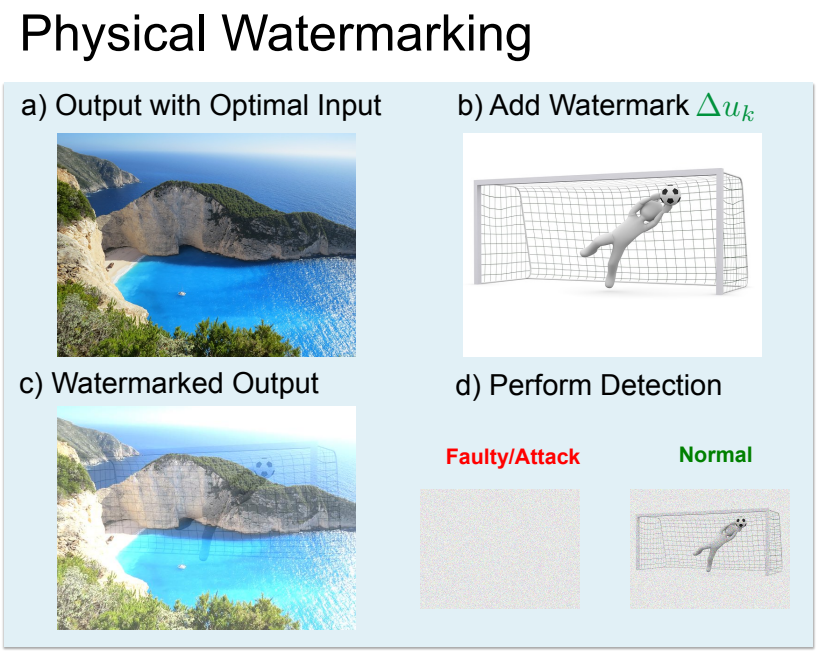


Figure 2.1: Physical Watermarking in Cyber-Physical Systems

assumed to be the solution of the following Lyapunov equation

$$\mathrm{Cov}(\xi_0) = A_\omega \,\mathrm{Cov}(\xi_0) A_\omega^T + \Psi.$$

All the matrices are of proper dimensions.

Remark 2.1. *It is worth noticing that* $\{\Delta u_k\}$ *is completely described by its finite dimensional distribution and hence the autocovariance function* Γ*. However, the watermarking signal is restricted to be generated from an HMM since any autocovariance function* Γ *can be approximated by an HMM, given that the dimension* n_h *of the hidden state is large enough. On the other hand, the HMM is easy to implement if* n_h *is small, which is the case for the optimal watermarking signal, as is illustrated later by Theorem 2.6.*

To ensure the freshness of the watermark signal, A_ω is assumed to be strictly stable, which implies that the correlation between the current watermark signal Δu_k and the future watermark signal $\Delta u_{k'}$ decays to 0 exponentially when $k' - k \to \infty$. The spectral radius of A_ω is denoted as $\rho(A_\omega) < 1$. In this section, it is assumed that the watermark signal is chosen from a Hidden-Markov

Model with $\rho(A_\omega) \leq \bar{\rho}$, where $\bar{\rho} < 1$ is a design parameter. A value of $\bar{\rho}$ close to 1 gives the system operator more freedom to design the watermark signal, while a value of $\bar{\rho}$ close to 0 improves the freshness of the watermark signal by reducing the correlation of Δu_k at different time steps. To simplify notations, define the feasible set $\mathcal{G}(\bar{\rho})$ as

$$\mathcal{G}(\bar{\rho}) = \{\Gamma : \Gamma \text{ is generated by an HMM (2.12) with } \rho(A_\omega) \leq \bar{\rho}\}. \tag{2.13}$$

Remark 2.2. *Since it is assumed that (A, B) is stabilizable and (A, C) is detectable, the closed-loop system is stable regardless of the watermark signal. Furthermore, by the separation principle, the Kalman filter is the optimal filter regardless of the watermark signal Δu_k. However, the addition of Δu_k incurs an LQG control performance loss and the control input u_k is not optimal. The necessity of adding the watermark signal Δu_k is illustrated later in Theorem 2.1. Conceptually, if the system is under normal operation, then the effect of the watermark signal Δu_k can be found in the sensor measurements y_k. The presence of the watermark is possibly lost when the system is malfunctioning or under attack, which can be detected by the failure detector.*

If no watermark signal is present, that is if $\Delta u_k = 0$, then the optimal objective function J^* given by the Kalman filter and LQG controller is

$$J^* = \operatorname{tr}(SQ) + \operatorname{tr}\left[\left(A^T S A + W - S\right)(P - KCP)\right]. \tag{2.14}$$

A passive detector is used to detect abnormality of the system in conjunction with our physical watermarking scheme. In this section, the passive detector is assumed to trigger an alarm at time k if and only if the condition,

$$g(z_k, \Delta u_{k-1}, \Delta u_{k-2}, \ldots) \geq \eta, \tag{2.15}$$

is met where $g(z_k, \Delta u_{k-1}, \Delta u_{k-2}, \ldots)$ is a continuous real valued function of $z_k, \Delta u_{k-1}, \Delta u_{k-2}, \ldots$ and η is the threshold, which is a design parameter of the system. Under normal operation, denote the probability of false alarm to be α, defined as

$$\alpha \triangleq \Pr(g(z_k, \Delta u_{k-1}, \Delta u_{k-2}, \ldots) \geq \eta). \tag{2.16}$$

False alarms usually occur with low probability for practical systems. When the system is operating normally, z_k is a stationary process and hence α is a constant.

Remark 2.3. *A widely used passive detector is the χ^2 detector ([16], [17]), which satisfies*

$$g(z_k, \Delta u_{k-1}, \Delta u_{k-2}, \ldots) = z_k^T (CPC^T + R)^{-1} z_k.$$

The χ^2 detector will be introduced and used later in this thesis.

Fig 2.2 shows the system diagram described in this section. In this system, no adversary is present and as a result, the watermark input is present in the sensor outputs. By confirming the presence of a watermark in the sensor measurements, a passive detector can verify that the system is not under a replay attack.

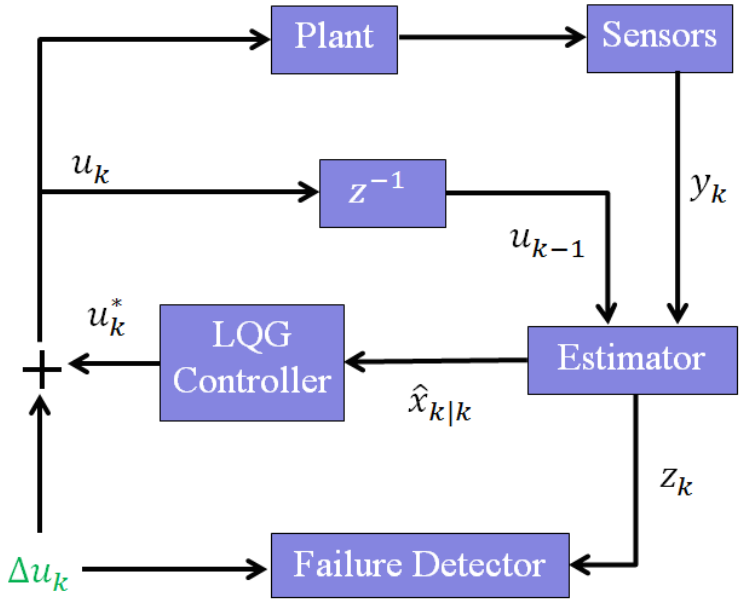


Figure 2.2: Diagram of system under normal operation

2.1.2 Attack Model

In this section, a model for a replay attack motivated by Stuxnet is given. To cause physical damage, a first version of Stuxnet implements control logic to increase pressure in the centrifuge

while a second version of the worm varies rotor speeds. To prevent detection in the first scenario, Stuxnet replayed previous sensor outputs to the SCADA system [18]. Since the system was in steady state, outputs from the past, collected in steady state, were statistically identical to outputs under normal operation, and as such were not detected. Motivated by Stuxnet, the following replay attack model is considered in this section.

Attacker's Knowledge and Resources

The adversary is first described through its knowledge and available resources.

1. The attacker has knowledge of all real time sensor measurements. In particular, he knows the true sensor outputs y_k for all k.

2. The attacker can violate the integrity of all sensor measurements. Specifically, he can modify the true sensor signals y_k to arbitrary sensor signals y_k^a.

   Remark 2.4. *The attack on the sensors can be carried out by breaking the cryptography algorithm. Another way to perform an attack, which is potentially much harder to defend, is to use physical attacks. Physical attacks can violate the basic properties of secrecy, integrity and availability without the need to attack the cyber part of the system. Consider for example a temperature sensor. Secrecy, integrity and availability of its sensing data can be affected by placing a sensor nearby, affecting the local temperature around the sensor, and enclosing the sensor with a metal cover respectively. In addition, the insider threat is critical in large infrastructures, as these systems usually involve many employees. These kinds of attacks may be easy to carry out when sensors are spatially distributed in remote locations.*

3. The attacker has access to a set of external actuators with control matrix $B^a \in \mathbb{R}^{n \times p_a}$ and can thus insert an external input $B^a u_k^a$ where $u_k^a \in \mathbb{R}^{p_a}$ is the control input. Moreover, assuming that u_k^a is intelligently chosen, the set of actuators B^a allows the adversary to achieve a malicious objective, for instance causing physical damage to the plant.

Remark 2.5. *The attacker could inject the external control input by controlling a subset of actuators of the system and/or deploying its own actuators. For example, to change the temperature distribution in a building, the attacker could take control of the HVAC (heating, ventilation, and air conditioning) system, deploy heaters of its own, or even commit arson.*

4. The attacker does not need to have full knowledge of the system parameters, namely the A, B, C, Q, R, K, L matrices and the Γ function. However, the attacker might have enough knowledge of the system model to design an input $u_k^a \in \mathbb{R}^{p_a}$, which may achieve its malicious objective such as physically damaging the plant.

Teixeira et al. [19], [20], demonstrate how an attacker can be succinctly characterized via a 3 dimensional attack space. The first dimension includes the attacker's system knowledge, including knowledge of the dynamics and controller. Knowledge of the model can for instance aid an attacker in constructing stealthy attack sequences that agree with a system's expected behavior while imparting maximal damage. The second dimension consists of the attacker's disclosure resources. This includes the information the attacker can gather about a system online. In our setting this refers to the ability to read sensor outputs and control inputs. Disclosure resources can allow an attacker to directly create harmful attacks, for instance through a replay attack. In addition, they can be utilized to enhance an attacker's understanding of the model, via system identification. Finally, the third dimension is an attacker's disruption resources, which characterize how an attacker can affect the system. If we limit ourselves to integrity attacks, this is characterized by the sensors and actuators an attacker can corrupt. Disruption resources allow an attacker to act on the system.

Figure 2.3 introduces the attack space. Here pure eavesdropping attacks only require disclosure resources, while a pure denial of service (DoS) attack requires only disruption resources. The covert attack is a stealthy attack that allows an attacker to completely appropriate a system [21]. A covert attacker can read all input and output sequences, disrupt all input and output channels, and has perfect model knowledge. A replay attack requires the disruption resources to read sensor

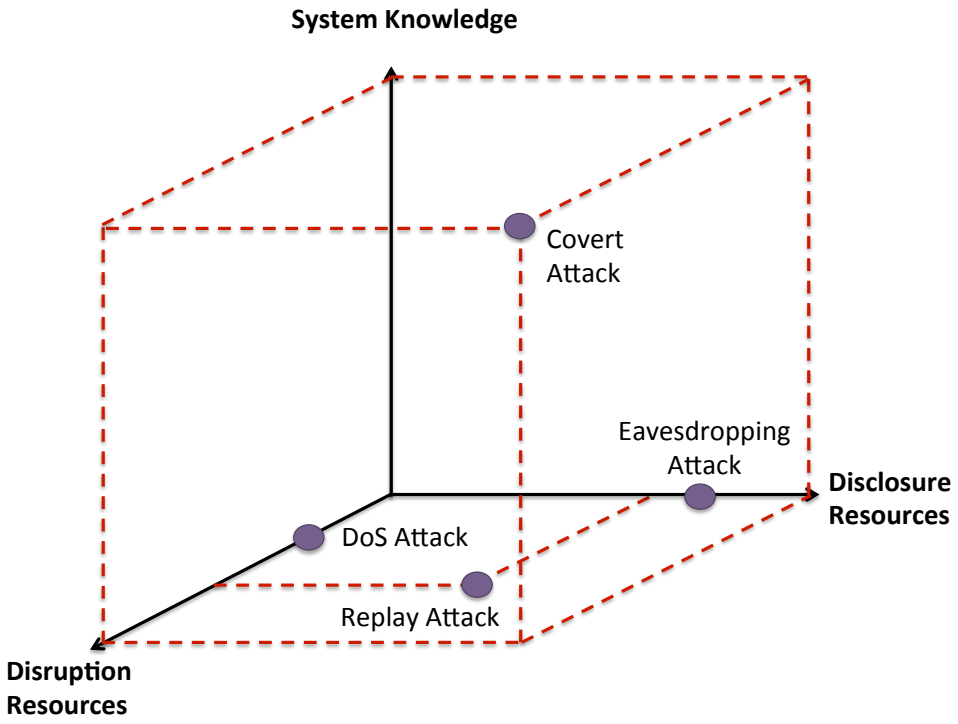


Figure 2.3: Cyber-Physical Attack Space

measurements. With a watermarking countermeasure, the control inputs must remain secret. The attacker also requires the ability to manipulate all sensor measurements. Moreover, the attacker may or may not need some minimal amount of system knowledge and access to (potentially external) actuators to cause damage to the system.

Attack Strategy

Given the adversary's knowledge and resources, the following attack strategy is considered.

1. The attacker records a sequence of sensor measurements from time $-T$ to time -1, where T is a large enough number to ensure that the attacker can replay the sequence later for an

extended period of time.

2. Starting at time 0 to time $T-1$, the attacker modifies the sensor signals to y_k^a, which is the same as the measurements recorded by the attacker at time $k-T$. In other words,

$$y_k^a = y_{k-T},\ 0 \leq k \leq T-1.$$

Remark 2.6. *For simplicity, the time that the replay starts is denoted as time* 0*. In reality, the attacker can freely choose the starting time, which is unknown to the system operator.*

3. Starting at time 0, the attacker injects an external control input $B^a u_k^a$, where $u_k^a \in \mathbb{R}^{p_a}$ is the control input and $B^a \in \mathbb{R}^{n \times p_a}$ denotes its direction.

Remark 2.7. *When the system is under attack, the controller cannot perform closed loop control since the true sensory information is not available. Therefore, control performance of the system cannot be guaranteed during the attack. In fact, the attacker can inject a bias on the state of the physical system along its controllable subspace, which is the column space of* $[B^a,\ AB^a, \ldots,\ A^{n-1}B^a]$*. The only way to counter this attack is to detect its presence.*

Remark 2.8. *For simplicity in this section, we consider the performance of watermarking against a replay attack. However, this active strategy can also be effective against alternative, otherwise stealthy adversaries. In the next section, we will consider an attacker who has access to a subset of control inputs and attempts to estimate the defender's state estimate in order to construct a stealthy attack output. In the next chapter, we will consider a simulation based attacker, who uses knowledge of the system model and the deterministic portion of a defender's control strategy to construct virtual sensor outputs.*

System Model Under Attack

To simplify notations, time-shifted variables,

$$\hat{x}_{k|k-1}^a \triangleq \hat{x}_{k-T|k-T-1},\ z_k^a = z_{k-T},\ \Delta u_k^a = \Delta u_{k-T},\ 0 \leq k \leq T-1, \tag{2.17}$$

are defined. During the replay ($0 \leq k \leq T-1$), the system dynamics changes to

$$x_{k+1} = Ax_k + Bu_k + B^a u_k^a + w_k,\ y_k = Cx_k + v_k, \tag{2.18}$$

$$\hat{x}_{k+1|k} = A\hat{x}_{k|k} + Bu_k,\ \hat{x}_{k|k} = \hat{x}_{k|k-1} + K\left(y_k^a - C\hat{x}_{k|k-1}\right), \tag{2.19}$$

$$u_k = L\hat{x}_{k|k} + \Delta u_k,\ z_k = y_k^a - C\hat{x}_{k|k-1}. \tag{2.20}$$

Notice that the fake measurement y_k^a is used instead of y_k for calculating the state estimate and residue. In addition, the probability of detection at time k is defined to be β_k given as

$$\beta_k \triangleq \Pr(g(z_k, \Delta u_{k-1}, \Delta u_{k-2}, \ldots) \geq \eta),\ \ 0 \leq k \leq T-1. \tag{2.21}$$

The following theorem characterizes the feasibility of the replay attack in the absence of the watermark signal Δu_k, which illustrates the necessity of the physical watermark.

Theorem 2.1. *Suppose $\Delta u_k = 0$ for all k. If $\mathcal{A} \triangleq (A+BL)(I-KC)$ is stable, $\rho((A+BL)(I-KC)) < 1$, then the detection rate β_k of all detectors g converges to the false alarm rate α during the attack, that is,*

$$\lim_{k\to\infty} \beta_k = \alpha. \tag{2.22}$$

On the other hand, if $\mathcal{A}$ is strictly unstable and g satisfies

$$\lim_{||z||\to\infty} g(z, 0, 0, \ldots) = \infty, \tag{2.23}$$

for some norm $||\cdot||$, then the detection rate β_k converges to 1, that is,

$$\lim_{k\to\infty} \beta_k = 1. \tag{2.24}$$

Proof. Part of the proof is reported in [15]. However, for the sake of completeness, the whole proof is included here. Manipulating (2.17)-(2.20) yields

$$\hat{x}_{k+1|k} = \mathcal{A}\hat{x}_{k|k-1} + (A+BL)Ky_k^a + B\Delta u_k \tag{2.25}$$

$$\hat{x}_{k+1|k}^a = \mathcal{A}\hat{x}_{k|k-1}^a + (A+BL)Ky_k^a + B\Delta u_k^a \tag{2.26}$$

$$z_{k+1} = z^a_{k+1} - C\mathcal{A}^{k+1}\left(\hat{x}_{0|-1} - \hat{x}^a_{0|-1}\right) - C\sum_{i=0}^{k}\mathcal{A}^{k-i}B\left(\Delta u_i - \Delta u^a_i\right). \tag{2.27}$$

If $\mathcal{A}$ is stable and $\Delta u_k = \Delta u^a_k = 0$, then the residue z_k of the system under the replay attack converges to the residue z^a_k of the virtual system, which is essentially z_{k-T}. Hence,

$$\lim_{k\to\infty}\beta_k = \lim_{k\to\infty}\Pr(g(z_k, 0, 0, \ldots) \geq \eta) = \Pr(g(z^a_k, 0, 0, \ldots) \geq \eta) = \Pr(g(z_{k-T}, 0, \ldots) \geq \eta) = \alpha$$

On the other hand, if $\mathcal{A}$ is strictly unstable, the second term on the RHS (right hand side) of (2.27) goes to infinity almost surely. Hence, if $g(z, 0, 0, \ldots) \to \infty$ when $||z|| \to \infty$,

$$\lim_{k\to\infty}\beta_k = \lim_{k\to\infty}\Pr(g(z_k, 0, 0, \ldots) \geq \eta) = 1,$$

which concludes the proof. □

Remark 2.9. *Notice that the stability of the "healthy" system depends only on the $A + BL$ and $A - KCA$ matrices, not on $\mathcal{A}$. Hence, it is entirely possible that the closed-loop system is stable while $\mathcal{A}$ is unstable. As seen from (2.25) and (2.26), the stability of $\mathcal{A}$ implies that the open-loop cyber system, consisting of the controller and estimator, is stable. In the one dimensional case, the stability of $\mathcal{A}$ is easy to analyze since $\mathcal{A} = (A + BL)(A - KCA)A^{-1}$. Thus, due to the stability of $A + BL$ and $A - KCA$, $\mathcal{A}$ is stable if A is unstable. Such analysis does not hold for higher dimensional systems since the product of stable matrices may not be stable.*

Remark 2.10. *Additionally, observe that Theorem 2.1 considers the alarm rate β_k when k goes to infinity while in the attack model it is assumed that the replay is performed from time 0 to time $T - 1$. However, since T is assumed to be large and β_k typically converges quickly, as is illustrated by the numerical examples, the asymptotic performance of β_k serves as an indicator of the detection performance of the system.*

Based on Theorem 2.1, if $\mathcal{A}$ is strictly unstable, then the attacker can be detected efficiently as the detection rate β_k converges to 1. However, if $\mathcal{A}$ is stable, then the attacker can perform the replay attack for an extended period of time given that the false alarm rate α is insignificant, which implies

that the system is not resilient to this type of attack. In that case, one possible countermeasure is to redesign the estimation and control gain matrices K and L so that the closed-loop system is stable, while enforcing $\mathcal{A}$ strictly unstable. However, this approach is not always desirable, since the control and estimation gain matrices are usually designed to satisfy certain safety and performance constraints and hence cannot be changed arbitrarily. In these scenarios, instead of redesigning K and L, the watermark signal can be used to enable intrusion detection.

We remark that other defense strategies can be vulnerable to replay attacks, beyond the LQG controller. A general condition characterizing the stealthiness of a replay attack given a particular estimator and controller can be found in [15]. More specifically, it can be shown simple output feedback strategy is susceptible to failure.

Corollary 2.1. *Suppose $\Delta u_k = 0$ for all k. Moreover, suppose the defender has a control strategy $u_k = h_k(y_k)$ where $h_k : \mathbb{R}^m \to \mathbb{R}^p$ is some deterministic function of y_k. If $A(I - KC)$ is Schur stable, then the detection rate β_k of all detectors g converges to the false alarm rate α during the attack, that is,*

$$\lim_{k\to\infty} \beta_k = \alpha. \tag{2.28}$$

The proof is similar to the proof of Theorem 2.1 and is thus omitted. Under the given system assumptions, $A(I-KC)$ is stable when using a Kalman filter and as a result a replay attack strategy against an output feedback strategy is asymptotically stealthy.

2.1.3 Watermark Design and Detection

This section is devoted to developing a design methodology for the watermark signal and the anomaly detector. To begin, the following assumption is made on the control system.

Assumption 2.1.1. *$\mathcal{A}$ is stable. That is, $\rho((A + BL)(I - KC)) < 1$.*

Throughout this section, it is assumed that $\mathcal{A}$ is stable, since otherwise the watermark signal would be unnecessary as a consequence of Theorem 2.1. To simplify notations, define the

symmetric part of a matrix X as

$$\mathrm{sym}(X) \triangleq \frac{X + X^T}{2}. \tag{2.29}$$

Performance Loss

When altering one's control strategy in order to improve detection performance, one needs to consider the very real fact that introducing a watermark degrades control performance. As such, when one considers this technique for active detection, the cost of control and the benefits of security must be carefully weighed against each other in the design of a potential watermarking scheme. We attempt to characterize control performance by examining the increase in the LQG cost. The following theorem provides the LQG control performance loss incurred by the watermark signal.

Theorem 2.2. *The LQG performance of the system described by* (2.1)*,* (2.2)*,* (2.4) *and* (2.10) *is given by*

$$J = J^* + \Delta J, \tag{2.30}$$

where J^* is the optimal LQG cost without the watermark signal and

$$\Delta J = \mathrm{tr}\left\{U\Gamma(0) + 2U\,\mathrm{sym}\left[L\sum_{d=0}^{\infty}(A+BL)^d B\Gamma(1+d)\right]\right\} + \mathrm{tr}\left[(W + L^T U L)\Theta_1\right], \tag{2.31}$$

where

$$\Theta_1 \triangleq 2\sum_{d=0}^{\infty}\mathrm{sym}\left[(A+BL)^d\mathcal{L}_1(\Gamma(d))\right] - \mathcal{L}_1(\Gamma(0)),$$

and $\mathcal{L}_1 : \mathbb{C}^{p\times p} \to \mathbb{C}^{n\times n}$ is a linear operator defined as

$$\mathcal{L}_1(X) = \sum_{i=0}^{\infty}(A+BL)^i BXB^T((A+BL)^i)^T = (A+BL)\mathcal{L}_1(X)(A+BL)^T + BXB^T.$$

Proof. The "healthy" control system follows

$$\begin{bmatrix} x_{k+1} \\ e_{k+1} \end{bmatrix} = \begin{bmatrix} A+BL & -BL \\ 0 & A-KCA \end{bmatrix}\begin{bmatrix} x_k \\ e_k \end{bmatrix} + \begin{bmatrix} I & 0 \\ I-KC & -K \end{bmatrix}\begin{bmatrix} w_k \\ v_{k+1} \end{bmatrix} + \begin{bmatrix} B\Delta u_k \\ 0 \end{bmatrix}, \tag{2.32}$$

and

$$u_k = L\hat{x}_{k|k} + \Delta u_k = Lx_k - Le_k + \Delta u_k. \tag{2.33}$$

Since the control system is closed-loop stable, $\{x_k\}$, $\{e_k\}$ and $\{u_k\}$ are all stationary Gaussian processes. Hence,

$$J = \mathbb{E}[x_1^T W x_1 + u_1^T U u_1] = \operatorname{tr}(W \operatorname{Cov}(x_1)) + \operatorname{tr}(U \operatorname{Cov}(u_1)).$$

By (2.32),

$$x_1 = l_1(w_0, w_{-1}, \dots, v_0, v_{-1}, \dots) + \sum_{i=0}^{\infty} (A+BL)^i B \Delta u_{-i},\ e_1 = l_2(w_0, w_{-1}, \dots, v_1, v_0 \dots),$$

where l_1 and l_2 are linear functions. As a result,

$$u_1 = l_3(w_0, w_{-1}, \dots, v_1, v_0, \dots) + L\sum_{i=0}^{\infty} (A+BL)^i B \Delta u_{-i} + \Delta u_1,$$

where l_3 is another linear function. Since the watermark signal is independent from the process noise $\{w_k\}$ and sensor noise $\{v_k\}$,

$$\operatorname{Cov}(x_1) = \operatorname{Cov}\left(l_1(w_0, w_{-1}, \dots, v_0, v_{-1}, \dots)\right) + \operatorname{Cov}\left(\sum_{i=0}^{\infty} (A+BL)^i B \Delta u_{-i}\right),$$

and

$$\operatorname{Cov}(u_1) = \operatorname{Cov}\left(l_3(w_0, w_{-1}, \dots, v_1, v_0, \dots)\right) + \operatorname{Cov}\left(L\sum_{i=0}^{\infty} (A+BL)^i B \Delta u_{-i} + \Delta u_1\right).$$

When $\Delta u_k = 0$, the optimal LQG cost is J^*. Thus, $J = J^* + \Delta J$ where

$$\Delta J = \operatorname{tr}\left[W \operatorname{Cov}\left(\sum_{i=0}^{\infty} (A+BL)^i B \Delta u_{-i}\right)\right] + \operatorname{tr}\left[U \operatorname{Cov}\left(L\sum_{i=0}^{\infty} (A+BL)^i B \Delta u_{-i} + \Delta u_1\right)\right]. \tag{2.34}$$

Manipulating the RHS of (2.34) leads to (2.31), which finishes the proof. □

Remark 2.11. *While the expression for ΔJ is complicated, it is linear with respect to the autocovariance functions $\Gamma(d)$. This linearity will be important as we attempt to formulate an efficiently solvable optimization problem which addresses the tradeoff between security and control performance when introducing physical watermarking as a means for active detection*

Optimal Detector

This subsection derives the asymptotically optimal detector. A detector has real time knowledge of the residue z_k, obtained from the estimator, as well as real time knowledge of the trajectory of the watermark, $\{\Delta u_k\}$. Define the covariance of the residue z_k of the healthy system to be

$$\bar{P} \triangleq CPC^T + R. \tag{2.35}$$

For the "healthy" system, z_k is Gaussian distributed with mean 0 and covariance $\bar{P}$.

By (2.27), for the system under the replay attack

$$z_{k+1} = -C\mathcal{A}^{k+1}\left(\hat{x}_{0|-1} - \hat{x}^a_{0|-1}\right) - C\sum_{i=0}^{k}\mathcal{A}^{k-i}B\Delta u_i + C\sum_{i=0}^{k}\mathcal{A}^{k-i}B\Delta u^a_i + z^a_{k+1}. \tag{2.36}$$

The first term on the RHS of (2.36) converges to 0 since $\mathcal{A}$ is stable. The second term is a function of the watermark signal, which is generated and thereby known by the control system and the failure detector. The third and fourth terms are independent from each other since z_k is the residue vector of the Kalman filter. Further define

$$\mu_k \triangleq -C\sum_{i=-\infty}^{k}\mathcal{A}^{k-i}B\Delta u_i, \tag{2.37}$$

and

$$\Sigma \triangleq \lim_{k\to\infty}\mathrm{Cov}\left[C\sum_{i=0}^{k}\mathcal{A}^{k-i}B\Delta u^a_i\right] = \mathrm{Cov}\left[C\sum_{i=0}^{\infty}\mathcal{A}^{i}B\Delta u_{-i}\right]. \tag{2.38}$$

Expanding the RHS of (2.38),

$$\Sigma = 2\sum_{d=0}^{\infty}C\,\mathrm{sym}\left[\mathcal{A}^d\mathcal{L}_2(\Gamma(d))\right]C^T - C\mathcal{L}_2(\Gamma(0))C^T, \tag{2.39}$$

where $\mathcal{L}_2 : \mathbb{C}^{p\times p} \to \mathbb{C}^{n\times n}$ is a linear operator on the space of $p \times p$ matrices, which is defined as

$$\mathcal{L}_2(X) \triangleq \sum_{i=0}^{\infty}\mathcal{A}^iBXB^T(\mathcal{A}^i)^T = \mathcal{A}\mathcal{L}_2(X)\mathcal{A}^T + BXB^T.$$

Therefore, z_k has a distribution that converges to a Gaussian distribution with mean μ_{k-1} and covariance $\bar{P} + \Sigma$. As a result, the null hypothesis is

$$\mathcal{H}_0 : \text{ the residue } z_k \text{ follows a Gaussian distribution } \mathcal{N}_0(0, \bar{P}).$$

The alternative hypothesis is

$$\mathcal{H}_1 : \text{ the residue } z_k \text{ follows a Gaussian distribution } \mathcal{N}_1(\mu_{k-1}, \bar{P} + \Sigma).$$

By the Neyman-Pearson lemma [22], the optimal detector is given by the Neyman-Pearson detector as discussed in Theorem 2.3.

Theorem 2.3. *The optimal Neyman-Pearson detector rejects $\mathcal{H}_0$ in favor of $\mathcal{H}_1$ if*

$$g_{NP}(z_k, \Delta u_{k-1}, \Delta u_{k-2}, \ldots) = z_k^T \bar{P}^{-1} z_k - (z_k - \mu_{k-1})^T (\bar{P} + \Sigma)^{-1} (z_k - \mu_{k-1}) \geq \eta. \quad (2.40)$$

Otherwise, hypothesis $\mathcal{H}_0$ is accepted.

To characterize the performance of the detector, ideally the asymptotic detection rate $\lim_{k\to\infty} \beta_k$ or expected time to detection is considered. However, the detection rate and expected time to detection involve integrating a Gaussian distribution, which usually does not have an analytical solution. In this section, the Kullback-Leibler (KL) divergence, which measures the "distance" between the two distributions, is used to characterize the detection performance. This choice rests on the observation that as the KL divergence between two distributions increases, the distributions become, roughly speaking, easier to distinguish.

The Kullback-Liebler divergence, first described in [23], is a measure of the difference between two distributions $f_1(z)$ and $f_0(z)$. For continuous probability density functions f_1 and f_0, the KL divergence is given as

$$D_{KL}(f_1 \| f_0) = \int_z f_1(z) \log \left(\frac{f_1(z)}{f_0(z)} \right) \mathrm{d}z. \quad (2.41)$$

It can be shown that $D_{KL}(f_1 \| f_0) \geq 0$. Moreover, equality holds if and only if $f_1(z) = f_0(z)$ for almost all z. Thus, if the distribution $f_1(z)$ is close to $f_0(z)$, the KL divergence likely approaches 0.

The KL divergence between distributions f_1 and f_0 can be related to the Neyman-Pearson detector associated with a binary hypothesis test. Here, consider $f_1(z)$ to be the distribution of the observations z under the alternative hypothesis $\mathcal{H}_1$ and $f_0(z)$ to be the distribution of the

observations z under the null hypothesis $\mathcal{H}_0$. The optimal Neyman-Pearson detector is a threshold detector on the log likelihood $l(z) = \log\left(\frac{f_1(z)}{f_0(z)}\right)$, where if $l(z)$ is greater than a constant c the alternative hypothesis is chosen. Observe that the KL divergence $D(f_1\|f_0)$ satisfies

$$D_{KL}(f_1\|f_0) = \mathbb{E}[l(z)|\mathcal{H}_1] \tag{2.42}$$

Thus, maximizing the KL divergence over a subset of possible distributions f_1 potentially increases the probability of an observation z such that $l(z) > c$, when the alternative hypothesis is true. As a result, the probability of detection also increases. For additional discussion of the relationship between the KL divergence and Neyman Pearson lemma, see [24].

The expected KL divergence of the two Gaussian distributions in $\mathcal{H}_1$ and $\mathcal{H}_0$ is given by the next theorem

Theorem 2.4. *The expected KL divergence of distribution $\mathcal{N}_1$ and $\mathcal{N}_0$ is*

$$\mathbb{E}[D_{KL}(\mathcal{N}_1\|\mathcal{N}_0)] = \operatorname{tr}(\Sigma\bar{P}^{-1}) - \frac{1}{2}\log\det(I + \Sigma\bar{P}^{-1}). \tag{2.43}$$

Furthermore, the expected KL divergence satisfies the inequality

$$\frac{1}{2}\operatorname{tr}(\Sigma\bar{P}^{-1}) \leq \mathbb{E}[D_{KL}(\mathcal{N}_1\|\mathcal{N}_0)] \leq \operatorname{tr}(\Sigma\bar{P}^{-1}) - \frac{1}{2}\log\left[1 + \operatorname{tr}(\Sigma\bar{P}^{-1})\right], \tag{2.44}$$

where the upper bound is tight if C is of rank 1.

Proof. By the definition of KL divergence, it is known that

$$\begin{aligned} D_{KL}(\mathcal{N}_1\|\mathcal{N}_0) =& \frac{1}{2}\operatorname{tr}\left[(\bar{P}+\Sigma)\bar{P}^{-1}\right] - \frac{m}{2} - \frac{1}{2}\log\det\left[(\bar{P}+\Sigma)\bar{P}^{-1}\right] + \frac{1}{2}{\mu_k}^T\bar{P}^{-1}\mu_k, \\ =& \frac{1}{2}\operatorname{tr}(\Sigma\bar{P}^{-1}) - \frac{1}{2}\log\det(I+\Sigma\bar{P}^{-1}) + \frac{1}{2}\operatorname{tr}(\mu_k{\mu_k}^T\bar{P}^{-1}). \end{aligned}$$

Take the expectation on both sides. It is easy to verify that $\Sigma = \mathbb{E}[\mu_k{\mu_k}^T]$, which proves (2.43).

Now assume that the eigenvalues of $\Sigma\bar{P}^{-1}$ are $\lambda_1, \cdots, \lambda_m$. As a result,

$$\operatorname{tr}(\Sigma\bar{P}^{-1}) = \sum_{i=1}^{m}\lambda_i,$$

and

$$\log\det(I+\Sigma\bar{P}^{-1}) = \sum_{i=1}^{m}\log(1+\lambda_i).$$

Since $\bar{P}$ is positive semidefinite, there exists a positive semidefinite matrix $\bar{P}^{1/2}$, where $\bar{P}^{1/2}\bar{P}^{1/2} = \bar{P}$. Hence, $\Sigma\bar{P}^{-1}$ shares the same eigenvalues as $\bar{P}^{-1/2}\Sigma\bar{P}^{-1/2}$, which implies all λ_is are real and nonnegative. As a result, by the concavity of $\log$ function, it is known that

$$\log\left[1+\mathrm{tr}(\Sigma\bar{P}^{-1})\right] \le \log\det(I+\Sigma\bar{P}^{-1}) \le m\log\left(1+\frac{\mathrm{tr}(\Sigma\bar{P}^{-1})}{m}\right) \le \mathrm{tr}(\Sigma\bar{P}^{-1}). \quad (2.45)$$

The first inequality holds when $\lambda_1 = \mathrm{tr}(\Sigma\bar{P}^{-1})$ and $\lambda_2 = \cdots = \lambda_m = 0$. The second inequality holds when $\lambda_1 = \cdots = \lambda_m = \mathrm{tr}(\Sigma\bar{P}^{-1})/m$. The third inequality uses the fact that $\log(1+x) \le x$. Combining (2.45) and (2.43), (2.44) holds.

Furthermore, if C is of rank 1, then by (2.39),

$$\mathrm{rank}(\Sigma\bar{P}^{-1}) \le \mathrm{rank}(\Sigma) \le 1.$$

As a result, the first inequality of (2.45) is tight, which implies that the upper bound in (2.44) is tight. □

It is worth noticing that the expected KL divergence is a convex function of Σ. However, both the upper and lower bound of the expected KL divergence are monotonically increasing with respect to $\mathrm{tr}(\Sigma\bar{P}^{-1})$, which is linear in Σ.

Optimal Watermark Signal

This subsection derives the optimal watermark signal. Ideally, the following optimization problem should be solved.

$$\begin{aligned} &\underset{\Gamma(d)\in\mathcal{G}(\bar{\rho})}{\text{maximize}} && \mathbb{E}[D_{KL}(\mathcal{N}_1\|\mathcal{N}_0)] \\ &\text{subject to} && \Delta J \le \delta, \end{aligned} \quad (2.46)$$

where $\delta > 0$ is a design parameter.

However, it is computationally hard to solve this maximization problem since the expected KL divergence is not a concave function of $\Gamma(d)$. Hence, the ensuing optimization problem is solved.

$$\begin{aligned} &\underset{\Gamma(d)\in\mathcal{G}(\bar{\rho})}{\text{maximize}} && \operatorname{tr}(\Sigma\bar{P}^{-1}) \\ &\text{subject to} && \Delta J \leq \delta, \end{aligned} \tag{2.47}$$

Notice that the expected KL divergence is relaxed to $\operatorname{tr}(\Sigma\bar{P}^{-1})$, using the upper and lower bound derived in Theorem 2.4. Furthermore, if C is of rank 1, then by Theorem 2.4, optimizing $\operatorname{tr}(\Sigma\bar{P}^{-1})$ is equivalent to optimizing the expected KL divergence. For general cases, the optimality gap can be quantified using the upper and lower bound. It is unclear how we can guarantee that $\Gamma(d) \in \mathcal{G}(\bar{\rho})$. To address this we make the following additional assumption.

Assumption 2.1.2. $\tilde{\Gamma}(d) = \bar{\rho}^{-|d|}\Gamma(d)$ *is an autocovariance function.*

$\tilde{\Gamma}(d)$ can be potentially realized by an alternate HMM

$$\tilde{\xi}_{k+1} = (A_\omega/\bar{\rho})\tilde{\xi}_k + \tilde{\psi}_k, \;\; \Delta\tilde{u}_k = C_h\tilde{\xi}_k, \tag{2.48}$$

$$\text{Cov}(\tilde{\xi}_0) = A_\omega \text{Cov}(\tilde{\xi}_0)A_\omega^T + \Psi, \tag{2.49}$$

$$\tilde{\psi}_k \sim \mathcal{N}(0, \text{Cov}(\tilde{\xi}_0) - A_\omega \text{Cov}(\tilde{\xi}_0)A_\omega^T/\bar{\rho}^2). \tag{2.50}$$

Note, that if $\rho(A_\omega) > \bar{\rho}$, (2.48) can not be a stationary process. This HMM can be realized if and only if $\text{Cov}(\tilde{\xi}_0) - A_\omega \text{Cov}(\tilde{\xi}_0)A_\omega^T/\bar{\rho}^2$ is positive semidefinite. Intuitively, if $\rho(A_\omega)$ is marginally less than $\bar{\rho}$, there is a larger chance that $\text{Cov}(\tilde{\xi}_0) - A_\omega \text{Cov}(\tilde{\xi}_0)A_\omega^T/\bar{\rho}^2$ is positive semidefinite.

If $\bar{\rho} = 1$, the space is not constricted by assumption 2.1.2 and in fact one will be able to optimize over all stationary Gaussian watermarks. We rewrite assumption 2.1.2 by defining the set $\mathcal{H}(\bar{\rho})$ as follows

$$\mathcal{H}(\bar{\rho}) = \{\Gamma : \bar{\rho}^{-|d|}\Gamma(d) \text{ is an autocovariance function of a stationary process}\}. \tag{2.51}$$

The resulting formulation is given as

$$\begin{aligned} &\underset{\Gamma(d)\in\mathcal{G}(\bar{\rho}),\Gamma(d)\in\mathcal{H}(\bar{\rho})}{\text{maximize}} && \operatorname{tr}(\Sigma\bar{P}^{-1}) \\ &\text{subject to} && \Delta J \leq \delta, \end{aligned} \tag{2.52}$$

Although Σ and ΔJ are linear functionals of Γ, convex optimization techniques cannot be directly applied to solve (2.52), since Γ is in an infinite dimensional space.

As a result, (2.52) is transformed into the frequency domain. Before continuing on, the following definition is needed.

Definition 2.2. *ν is a positive Hermitian measure of size $p \times p$ on the interval $(-0.5, 0.5]$ if for a Borel set $S_B \subseteq (-0.5, 0.5]$, $\nu(S_B)$ is a positive semidefinite Hermitian matrix with size $p \times p$.*

The following theorem establishes the existence of a frequency domain representation for $\Gamma(d)$.

Theorem 2.5 (Bochner's Theorem [25, 26])**.** *$\Gamma(d)$ is the autocovariance function of a stationary Gaussian process $\{\Delta u_k\}$ if and only if there exists a unique positive Hermitian measure ν of size $p \times p$, such that*

$$\Gamma(d) = \int_{-1/2}^{1/2} \exp(2\pi j d\omega)\, \mathrm{d}\,\nu(\omega). \tag{2.53}$$

$\mathrm{d}\,\nu(\omega)$ can be interpreted as the discrete-time Fourier transform of the function $\Gamma(d)$. In fact, if $\nu(\omega)$ is absolutely continuous with respect to the Lebesgue measure, then

$$\mathrm{d}\,\nu(\omega) = f(\omega)\,\mathrm{d}\,\omega,$$

and

$$\Gamma(d) = \int_{-1/2}^{1/2} \exp(2\pi j d\omega) f(\omega)\,\mathrm{d}\,\omega,$$

where f is a mapping from $(-0.5, 0.5]$ to the set of positive semidefinite Hermitian matrices. f is exactly the "entrywise" Fourier transform of $\Gamma(d)$.

By the fact that $\Gamma(d)$ is real, the Hermitian measure ν satisfies the following property, which can be applied to the Fourier transform of the real valued signals.

Proposition 1. *$\Gamma(d)$ is real if and only if for all Borel-measureable sets $S_B \subseteq (-0.5, 0.5]$,*

$$\nu(S_B) = \overline{\nu(-S_B)}. \tag{2.54}$$

By (2.54), (2.53) can be simplified as

$$\Gamma(d) = 2\,\mathfrak{Re}\left(\int_0^{1/2} \exp(2\pi j d\omega)\,\mathrm{d}\,\nu(\omega)\right). \tag{2.55}$$

Theorem 2.6. *The optimal solution (not necessarily unique) of (2.52) is*

$$\Gamma_*(d) = 2\bar{\rho}^{|d|}\,\mathfrak{Re}\left[\exp(2\pi j d\omega_*)H_*\right], \tag{2.56}$$

where ω_* and H_* are the solution of the ensuing optimization problem.

$$\begin{aligned} &\underset{\omega,H}{\textit{maximize}} && \mathrm{tr}\left[\mathcal{F}_2(\omega,H)C^T\bar{P}^{-1}C\right] \\ &\textit{subject to} && \mathcal{F}_1(\omega,H) \le \delta,\ 0 \le \omega \le 0.5, \\ & && H \textit{ Hermitian and Positive Semidefinite,} \end{aligned} \tag{2.57}$$

where the function $\mathcal{F}_1$ is defined as

$$\mathcal{F}_1(\omega,H) \triangleq \mathrm{tr}\left[U\Theta_2\right] + \mathrm{tr}\left[(W + L^TUL)\Theta_3\right], \tag{2.58}$$

$$\Theta_2 \triangleq 2\,\mathfrak{Re}\left\{2\,\mathrm{sym}\left(s\bar{\rho}L[I - s\bar{\rho}(A+BL)]^{-1}BH\right) + H\right\},$$

$$\Theta_3 \triangleq 2\,\mathfrak{Re}\left\{2\,\mathrm{sym}\left[(I - s\bar{\rho}(A+BL))^{-1}\mathcal{L}_1(H)\right] - \mathcal{L}_1(H)\right\},$$

and $s \triangleq \exp(2\pi j\omega)$.

The function $\mathcal{F}_2$ is defined as

$$\mathcal{F}_2(\omega,H) \triangleq 2\,\mathfrak{Re}\left\{2\,\mathrm{sym}\left[(I - s\bar{\rho}\mathcal{A})^{-1}\mathcal{L}_2(H)\right] - \mathcal{L}_2(H)\right\}. \tag{2.59}$$

Furthermore, one optimal (not necessarily unique) H_* of Problem (2.57) is of the form

$$H_* = hh^H, \tag{2.60}$$

where $h \in \mathbb{C}^p$. The corresponding HMM is given by

$$\xi_{k+1} = \bar{\rho} \begin{bmatrix} \cos 2\pi\omega_* & -\sin 2\pi\omega_* \\ \sin 2\pi\omega_* & \cos 2\pi\omega_* \end{bmatrix} \xi_k + \psi_k, \quad \Delta u_k = \begin{bmatrix} \sqrt{2}h_r & \sqrt{2}h_i \end{bmatrix} \xi_k, \tag{2.61}$$

where $h_r,\ h_i \in \mathbb{R}^p$ are the real and imaginary part of h respectively and $\Psi = \mathrm{Cov}(\psi_k) = (1-\bar{\rho}^2)I$.

The proof is found in the appendix.

Remark 2.12. *By (2.56), $\Gamma_*(d)$ can be seen as a sinusoidal signal with a decay factor $\bar{\rho}$, where ω_* and H_* can be interpreted as the optimal frequency and direction respectively. Since $\mathcal{F}_1$ and $\mathcal{F}_2$ are linear with respect to H, when ω is fixed, (2.57) is a semidefinite programing problem and hence can be solved efficiently. Therefore, (2.57) can be solved in two steps by first calculating the optimal signal direction for every frequency $0 \leq \omega \leq 0.5$ and then searching over all possible frequencies ω. In practice, (2.57) can be solved for enough sample frequencies to obtain a near optimal watermarking signal.*

It is worth noticing that regardless of the dimensions of the physical system n or the control input p, the dimension of the hidden ξ_k is always 2, which is desirable from a computational perspective when dealing with a high-dimensional linear system.

There always exists an optimal solution that is a noisy sinusoid. The fact that a single frequency is optimal occurs because both the objective function and constraint can be expressed as infinite Riemann sums which are functions of $\nu(\omega)$. Specifically, both are linear functions of an infinite sequence $\{\nu(\omega_i)\}$. The fact the objective and constraint are linear across $\nu(\omega_i)$ means there is an optimal solution consisting of a single frequency. Removing these linearities, perhaps by considering nonlinear systems, could potentially result in an optimal watermark containing multiple frequencies.

2.1.4 Numerical Example

This section illustrates the utility of the watermarking scheme by analyzing detection performance on a control system, with parameters

$$A = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}, B = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, C = \begin{bmatrix} 1 & 0 \end{bmatrix}. \tag{2.62}$$

The cost matrices in this system, W and U, are equal to the identity. The covariance matrices, Q and R, are equal to 0.8 times the identity and the identity respectively. As a result, the eigenvalues of $\mathcal{A}$ are -0.339 and -0.105. Consequently, $\mathcal{A}$ is stable, thus motivating the use of a watermark signal for detection. Two watermarking designs are analyzed. First, a stationary watermark is generated using (2.61) where $\bar{\rho} = 0.6$. In the second case, an IID Gaussian process is considered, similar to the design presented in [15, 27]. Designing a stationary Gaussian watermark requires solving a semidefinite program for a set of frequencies sampled in $0 \leq \omega \leq 0.5$. A step size of 0.01 is chosen for this system, which requires solving 51 semidefinite programs. On a Macbook Pro with a 2.4 GHz processor, solving all 51 semidefinite programs takes 12.9 seconds using CVX [28, 29].

First, the asymptotic detection rate $\lim_{k\to\infty} \beta_k$ versus the false alarm rate α for each design is plotted in Fig 2.4. The additional cost ΔJ imposed by the watermark is 10 for each design, roughly 40 percent of the optimal cost $J^* = 23.1$.

The relationship between the asymptotic detection rate and false alarm rate is again considered in Fig 2.5. Here, α is chosen to be less than 0.1, which is typical for real systems, where the cost considerations of investigating possible attacks make it undesirable to have frequent false alarms during normal operation. The stationary watermarking design offers a visible improvement in the asymptotic rate of detection over an IID design. The stationary watermarking scheme with $\rho = 0.6$ obtains its best relative performance in comparison to independent and identically distributed (IID) watermarking schemes when the probability of false alarm approaches 0. The percent improvement in asymptotic detection rate $\lim_{k\to\infty} \beta_k$ of the stationary Gaussian design with $\bar{\rho} = 0.6$ over the IID approach is explicitly examined in Fig 2.6 for $\alpha \leq 0.1$. It can be seen that the stationary watermark

achieves its best relative performance for α in this range. In fact, a 60 percent improvement over the IID design in the asymptotic rate of detection is obtained when $\alpha \approx 0.005$ and $\bar{\rho} = 0.6$.

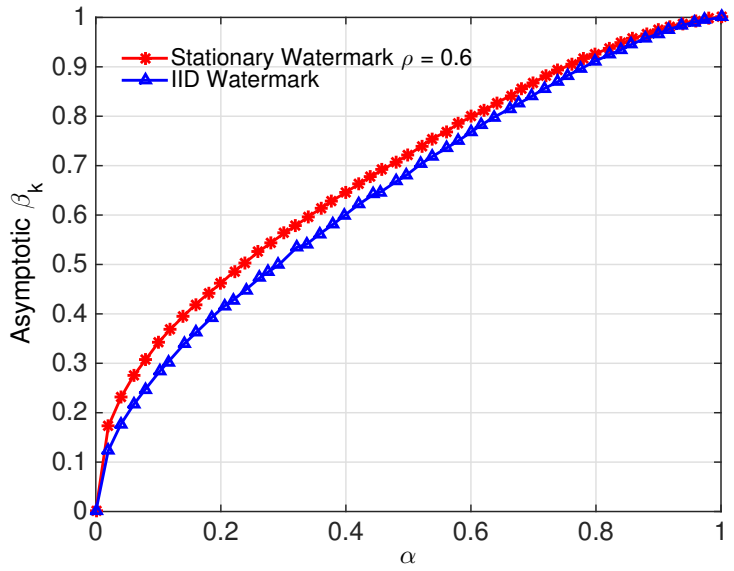


Figure 2.4: $\lim_{k\to\infty} \beta_k$ as a function of α for a stationary watermark with $\rho = 0.6$, and an independent and identically distributed (IID) watermark with $\Delta J = 10$.

Fig 2.7 and Fig 2.8 illustrate the tradeoff between the asymptotic detection rate $\lim_{k\to\infty} \beta_k$ and the LQG cost ΔJ for $\Delta J \leq 100$ and $\Delta J \leq 20$ respectively. For this simulation, the false alarm rate α is fixed to be 0.02. For practical systems, ΔJ needs to be carefully chosen to balance the control cost and the detection performance. These figures show that as more control effort is expended, the rate of detection increases. In particular, additional linear-quadratic-Gaussian (LQG) cost corresponds to increasing the magnitude of the watermark's autocovariances. Through the dynamics of the system, watermarks with larger autocovariances increase discrepancies between the replayed sensor outputs and the expected sensor outputs, thus resulting in a higher probability of detection.

Fig 2.9 shows the detection rate as a function of time k where $\Delta J = 10$ for the watermarking approaches and $\alpha = 0.02$. In this scenario, detection performance in the absence of physical watermarking is also considered. For this case, a χ^2 detector is used. It is assumed that the attacker

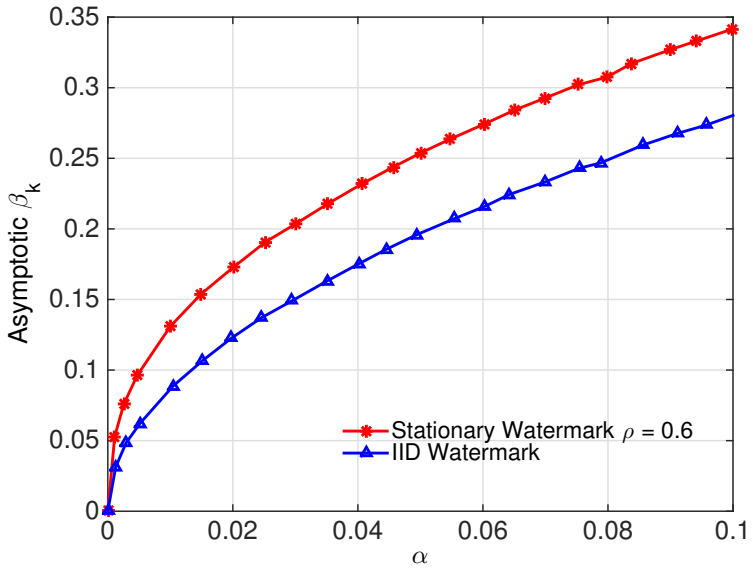


Figure 2.5: $\lim_{k\to\infty} \beta_k$ as a function of α,for $\alpha \leq 0.1$, for a stationary watermark with $\rho = 0.6$, and an independent and identically distributed (IID) watermark with $\Delta J = 10$.

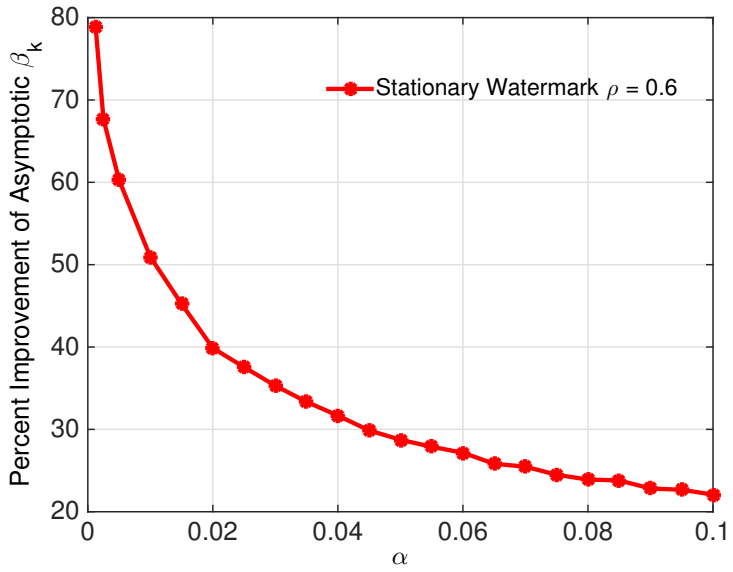


Figure 2.6: Percentage improvement in $\lim_{k\to\infty} \beta_k$ over the independent and identically distributed (IID) design versus α for a stationary watermarking scheme with $\rho = 0.6$ and $\Delta J = 10$.

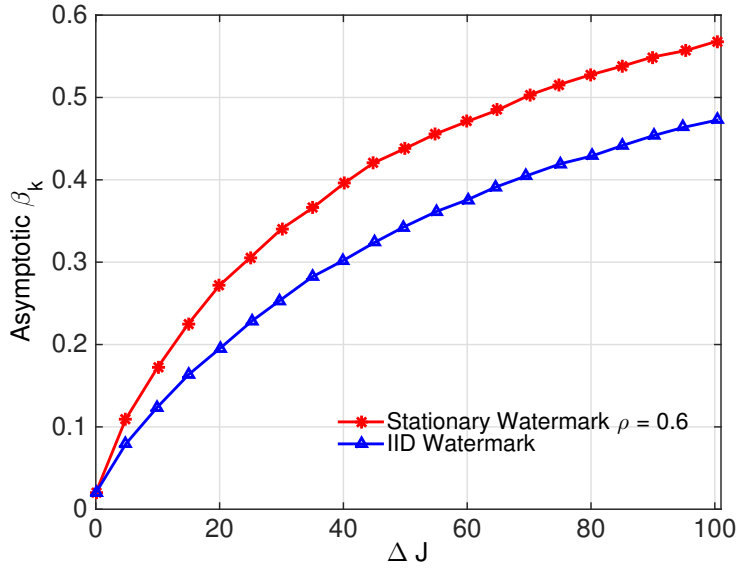


Figure 2.7: $\lim_{k\to\infty} \beta_k$ versus ΔJ for a stationary watermark with $\rho = 0.6$ and an independent and identically distributed (IID) watermark, $\alpha = 0.02$.

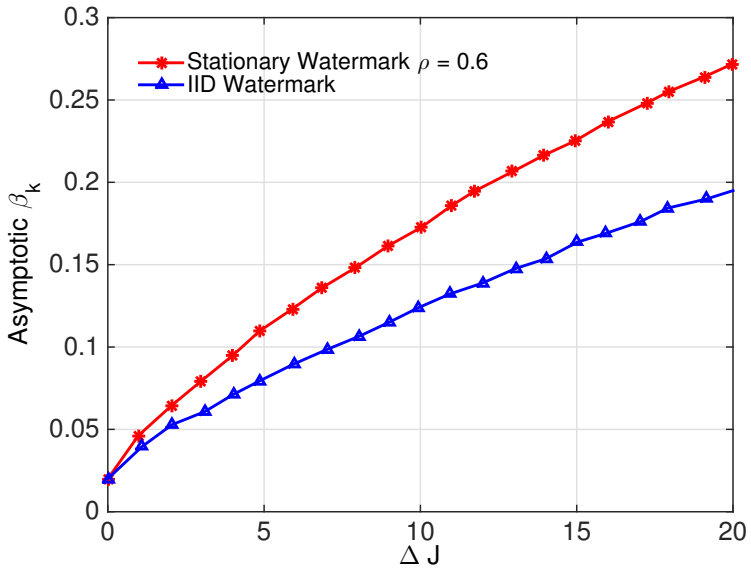


Figure 2.8: $\lim_{k\to\infty} \beta_k$ versus ΔJ for a stationary watermarks with $\rho = 0.6$ and an independent and identically distributed (IID) watermark, $\alpha = 0.02, \Delta J \leq 20$.

gathers measurements from $-50 \leq k \leq -1$ and replays these measurements from $0 \leq k \leq 49$. For all chosen designs, the probability of detection quickly rises to a maximum detection rate at $k = 0$ due to a mismatch between the expected and received measurements at the beginning of a replay attack. However, since $\mathcal{A}$ is stable, the detection rate quickly decreases back to false alarm rate without watermarking. Meanwhile, in the watermarking strategies β_k converges quickly. As a result, it is reasonable to design the watermark signal to optimize the asymptotic detection performance.

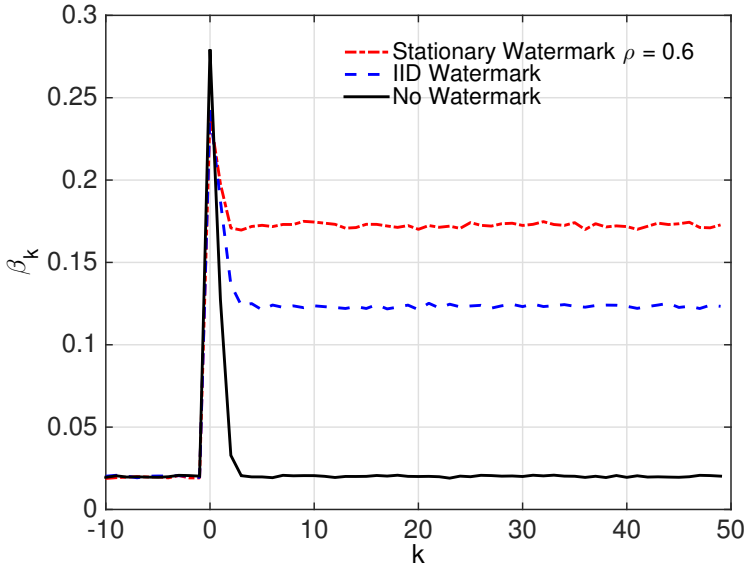


Figure 2.9: β_k versus time k for a stationary watermark with $\rho = 0.6$, an independent and identically distributed (IID) watermark, and no watermark. For watermarking schemes, $\Delta J = 10$, and $\alpha = 0.02$.

Finally, Fig 2.10 examines the relationship between the expected time of detection and the additional LQG cost ΔJ when $\alpha = 0.02$. In the absense of physical watermarking, which corresponds to $\Delta J = 0$, the expected time of detection is roughly given by $k = 34.3$. Watermarking strategies can significantly reduce the time of detection. For instance, for $\Delta J = 10$, the expected time of detection for the stationary watermark is $k = 5.82$ and the expected time of detection for the IID watermark is $k = 6.27$.

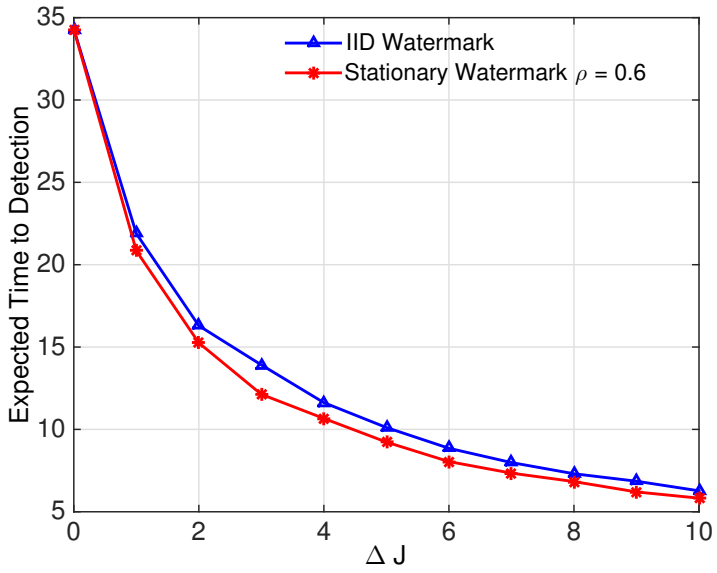


Figure 2.10: Expected time of detection versus ΔJ for a stationary watermark with $\rho = 0.6$, and an independent and identically distributed (IID) watermark, $\alpha = 0.02$.

2.2 Robust Physical Watermarking

In the prior section, we introduced physical watermarking as an active method for detection, illustrating its effectiveness against replay attacks. In this section, we aim to demonstrate that this mechanism for active detection can remain effective in other scenarios, even when considering attackers who retain model knowledge. The root of trust in the prior section was the watermark and in general the control input. In particular, it was assumed that the attacker did not have knowledge of the control input. We relax this assumption by instead assuming the attacker has access to a subset of control inputs while the defender is able to keep a known subset of inputs hidden from the attacker. Given this scenario, we aim to design a so called robust watermarking scheme.

2.2.1 System Description

The system description remains unchanged from the prior section. The dynamics are represented via a LTI control system (2.1) which is monitored by a suite of sensors (2.2). The designers aim to

minimize an LQG cost (2.3). This is done via a combination of a Kalman filter (2.4),(2.5),(2.6), and linear state feedback controller (2.7),(2.8),(2.9).

We again examine the binary detection problem of verifying the integrity of sensor measurements. For simplicity, we assume that any integrity attack begins at time 0. Here, we generically define the null hypothesis $\mathcal{H}_0$ and alternative hypothesis $\mathcal{H}_1$ as follows.

$$\begin{aligned} \mathcal{H}_0 : &\quad \text{The system is operating normally.} \\ \mathcal{H}_1 : &\quad \text{An attacker implements an attack strategy } \mathcal{Z}. \end{aligned}$$

A detector of the form

$$g(z_k) \underset{\mathcal{H}_0}{\overset{\mathcal{H}_1}{\gtrless}} \eta, \quad z_k \triangleq y_k - C\hat{x}_{k|k-1}, \tag{2.63}$$

is implemented where z_k is the current residue and $g : \mathbb{R}^m \to \mathbb{R}$ is a scalar valued function. The detector g leverages the fact that under normal operation

$$z_k \sim \mathcal{N}(0, \bar{P}), \quad \bar{P} \triangleq CPC^T + R, \tag{2.64}$$

to find faulty data. The probability of detection when the system is under attack β_k and the probability of false alarm α are defined respectively as

$$\beta_k \triangleq \Pr(g(z_k) > \eta | \mathcal{H}_1), \quad \alpha \triangleq \Pr(g(z_k) > \eta | \mathcal{H}_0). \tag{2.65}$$

As before, since z_k is stationary under normal operation, α is constant. While we consider the class of detectors which only consider the current residue, previous residues can also be considered. It can be shown, that the class of residue detectors are vulnerable to stealthy integrity attacks. Namely, we have the following result.

Theorem 2.7. *Suppose an attacker modifies sensor measurements to y_k^a so that the measurements y_k^a are statistically identical to measurements y_k gathered under normal operation. That is y_k^a is a zero-mean Gaussian process such that*

$$\mathrm{Cov}(y_{k_1}, y_{k_2}) = \mathrm{Cov}(y_{k_1}^a, y_{k_2}^a) \quad \forall\, k_1,\ k_2. \tag{2.66}$$

Furthermore suppose the matrix $(A + BL)(I - KC)$ *is stable. Then for all residue detectors* g *that are continuous in* z_k

$$\lim_{k\to\infty} \beta_k = \alpha. \tag{2.67}$$

The proof is essentially equivalent to that of Theorem 2.1 and is thus omitted.

The above theorem shows that for certain systems, if an attacker is able to generate statistically correct measurements y_k^a, no residue detector can asymptotically provide any information about whether an integrity attack has taken place. We showed that an attacker can generate statistically correct sensor measurements through a replay attack. If the system model is known to the attacker, y_k^a can be generated if the attacker simulates his own version of the system.

To counter such an adversary, for simplicity, we consider the design of an IID watermark $\Delta u_k \sim \mathcal{N}(0, \mathcal{J})$ (first considered in [15]) added on top of the optimal input u_k^*,

$$u_k = u_k^* + \Delta u_k. \tag{2.68}$$

2.2.2 Attack Model

In our attack model, we consider a near omniscient adversary. Such an adversary may be highly sophisticated such as in Stuxnet. On the other hand, malicious insiders may have significant access to system components and knowledge that are not publicly available. They can in turn leverage their knowledge and resources to design stealthy attacks [6]. We consider an attacker with the following capabilities.

1) The attacker can insert an external control input $B^a u_k^a$ starting at time $k = 0$ where u_k^a is a control input and B^a denotes the direction.

Remark 2.13. *Here, the attacker can take over a subset of control inputs as was done in Stuxnet so that* B^a *is contained in* B*. Alternatively, the attacker can inject his own control input. The goal of the attacker is to design the input so as to cause physical damage to the system. An alternative approach would be to perform a purely sensor based false data injection attack so that*

the system destabilizes by the actions of the system operator without the presence of an external input. However, such an attack can be limited in terms of the range of achievable states or the speed at which damage can be achieved.

2) The attacker knows the system model, $\mathcal{M} = \{A, B, C, K, L, Q, R, W, U, \mathcal{J}\}$.

Remark 2.14. *A sophisticated attacker could potentially perform system identification to obtain the system matrices. Alternatively, a malicious insider could obtain knowledge of the model.*

3) The attacker can arbitrarily modify all sensor measurements from y_k to y_k^a starting at time $k = 0$. Moreover, the attacker can read the true sensor measurements y_k for all k.

Remark 2.15. *Unlike replay attacks considered in the prior section as well as in [15],[30],[27], knowledge of true sensor measurements in real time can be leveraged by the attacker to improve the statistical properties of y_k^a.*

Remark 2.16. *In theory, knowledge of y_k, the injected control input $B^a u_k^a$, and the system model, would allow an adversary to subtract his influence and generate undetectable virtual outputs y_k^a. However, for such a scheme to be successful, the plant must be open loop stable. If A is unstable, any cancellation errors due to modeling discrepancies would grow exponentially. Moreover, if A is stable, the input u_k^a could cause the system to enter a nonlinear operating region, nullifying the attacker's knowledge of the model.*

4) The attacker can read a subset of control inputs for all k. That is u_k can, without loss of generality, be partitioned as $u_k^T = [{u_k^1}^T {u_k^2}^T]^T$ where $u_k^1 \in \mathbb{R}^{p_1}$ is known to the attacker while $u_k^2 \in \mathbb{R}^{p-p_1}$ remains secret.

Remark 2.17. *In previous work, all inputs were secret to the attacker. However, if the attacker is able to modify a subset of control inputs, it is likely that the attacker can read the intended inputs chosen by the system designer. The root of trust in this system is the set of secret inputs u_k^2. If the attacker has knowledge of all inputs as well as the model, he can generate virtual outputs by*

simulating the system. By assumption, the set of vulnerable inputs are known to both the system operator and adversary. Again, knowledge of a subset of control inputs can be used by the attacker to generate y_k^a. The attacker is summarized in Fig. 2.11

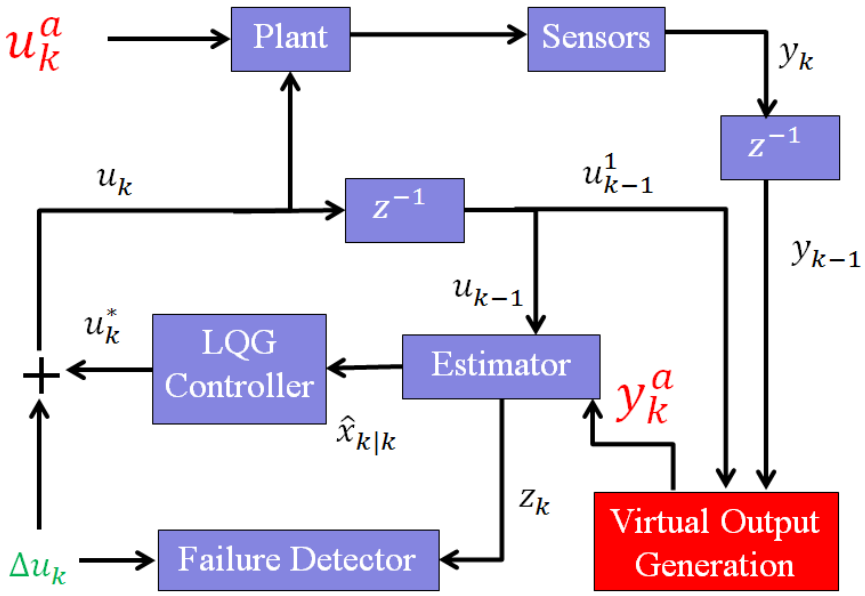


Figure 2.11: Diagram of System under Robust Attack

We can again characterize the attacker in terms of the three dimensional attack space. We will refer to this attack as the robust attack. Relative to the replay attacker, we observe that this adversary has full model knowledge, as well as the ability to read a subset of inputs. The attack space is illustrated in Fig. 2.12.

Before we continue, we introduce notation where the superscript ^{1} denotes parameters associated

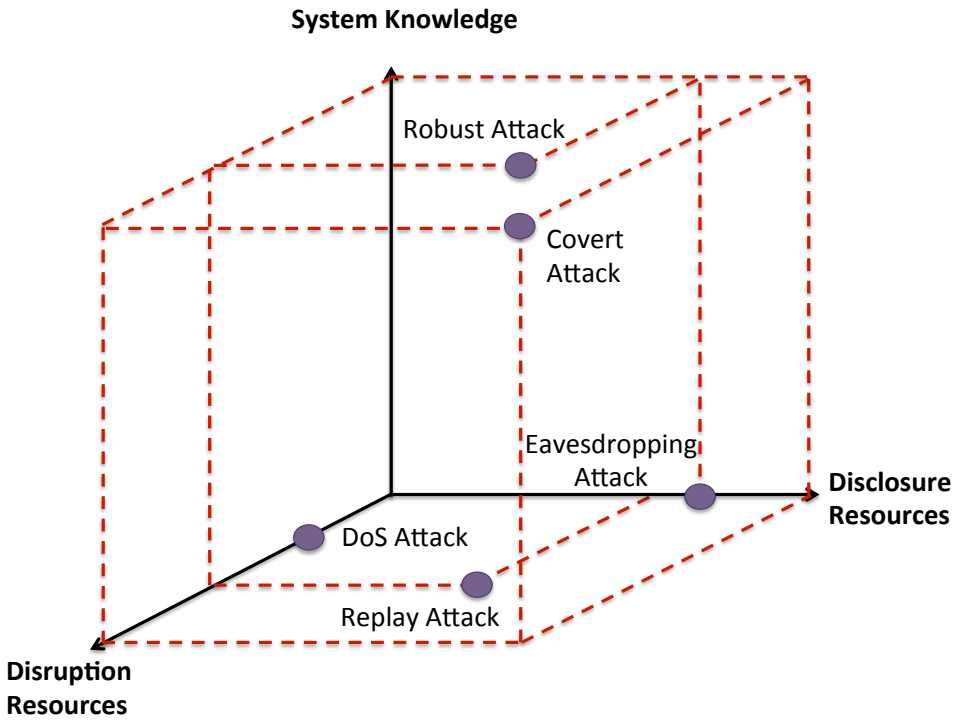


Figure 2.12: Cyber-Physical Attack Space with Robust Attack

with known inputs and the superscript ^{2} denotes parameters associated with unknown inputs.

$$Bu_k = [B^1\ B^2]\begin{bmatrix} u_k^1 \\ u_k^2 \end{bmatrix} = B^1u_k^1 + B^2u_k^2, u_k^1 = L^1\hat{x}_k + \Delta u_k^1,\ u_k^2 = L^2\hat{x}_k + \Delta u_k^2, \tag{2.69}$$

$$\Delta u_k^1 \sim \mathcal{N}(0, \mathcal{J}_1), \quad \Delta u_k^2 \sim \mathcal{N}(0, \mathcal{J}_2), \quad \mathcal{J}_1 \geq \epsilon I, \quad \epsilon > 0, \quad \mathcal{J}_2 \geq 0, \tag{2.70}$$

$$U = \begin{bmatrix} U^1 & U^{12} \\ U^{21} & U^2 \end{bmatrix}, \quad L = \begin{bmatrix} L^1 \\ L^2 \end{bmatrix}.$$

Here, we design Δu_k^1 and Δu_k^2 to be independent using the rationale that the system operator would not want to provide any information about Δu_k^2 in the watermark Δu_k^1 if u_k^1 is vulnerable. Note that $\mathcal{J}_1 \geq \epsilon I$ for some $\epsilon > 0$ to be chosen by the designer. We remark that this assumption allows us to use theoretical results regarding the discrete algebraic Riccati equation (DARE) included later

in the section and can be relaxed by choosing ϵ to be close to 0. As in the previous section, the dynamics under attack can be characterized as

$$x_{k+1} = Ax_k + Bu_k + B^a u_k^a + w_k,\ y_k = Cx_k + v_k, \tag{2.71}$$

$$\hat{x}_{k+1|k} = A\hat{x}_{k|k} + Bu_k,\ \hat{x}_{k|k} = \hat{x}_{k|k-1} + K\left(y_k^a - C\hat{x}_{k|k-1}\right), \tag{2.72}$$

$$u_k = L\hat{x}_{k|k} + \Delta u_k,\ z_k = y_k^a - C\hat{x}_{k|k-1}. \tag{2.73}$$

2.2.3 Attack Strategy

We would like to determine how the attacker should generate virtual outputs as illustrated in Fig. 2.11. To fool a detector $g(z_k)$, the adversary would like to generate measurements y_k^a so that residue under attack $y_k^a - C\hat{x}_{k|k-1}$ has statistical properties approaching the residue under normal operation which has distribution $\mathcal{N}(0, \bar{P})$ (or equivalently $y_k \sim \mathcal{N}_0(C\hat{x}_{k|k-1}, \bar{P})$) under normal operation. In the following theorem, we assume the attacker wishes to minimize the distance between distributions of the residue under attack and normal operation using the available information.

Theorem 2.8. *Denote the information available to the adversary at time k as $\mathcal{I}_k^a$. Suppose the attacker generates virtual measurements y_k^a such that*

$$y_k^a \sim \mathcal{N}_1(\mu_k^a, \Sigma^a). \tag{2.74}$$

The expected KL divergence $\mathbb{E}\left[D_{KL}\left(\mathcal{N}_1||\mathcal{N}_0\right)|\mathcal{I}_k^a\right]$ is minimized by

$$\mu_k^a = C\mathbb{E}[\hat{x}_{k|k-1}|\mathcal{I}_k^a],\ \ \Sigma^a = \bar{P}. \tag{2.75}$$

Thus, under attack $z_k \sim \mathcal{N}_1(C\mathbb{E}[\hat{x}_{k|k-1}|\mathcal{I}_k^a] - C\hat{x}_{k|k-1}, \bar{P})$.

Proof. Letting $\nu_k^a \triangleq \mu_k^a - C\hat{x}_{k|k-1}$, the expected KL divergence of probability distributions $y_k \sim \mathcal{N}_0(C\hat{x}_{k|k-1}, \bar{P})$ and $y_k^a \sim \mathcal{N}_1(\mu_k^a, \Sigma^a)$ or $\mathbb{E}[D_{KL}\left(\mathcal{N}_1||\mathcal{N}_0\right)|\mathcal{I}_k^a]$ is given by

$$\mathbb{E}[D_{KL}\left(\mathcal{N}_1||\mathcal{N}_0\right)|\mathcal{I}_k^a] = \frac{1}{2}\operatorname{tr}(\Sigma^a\bar{P}^{-1}) + \frac{1}{2}\left(\mathbb{E}\left[\nu_k^{aT}\bar{P}^{-1}\nu_k^a|\mathcal{I}_k^a\right] - \log\det(\Sigma^a\bar{P}^{-1}) - m\right). \tag{2.76}$$

We would like to obtain the minimizing Σ^a and μ_k^a, which can be obtained by solving two optimization problems.

$$\Sigma_{opt}^a = \arg\min_{\Sigma^a} \left(\mathrm{tr}(\Sigma^a \bar{P}^{-1}) - \log\det(\Sigma^a \bar{P}^{-1}) \right). \tag{2.77}$$

Since the objective is convex in Σ^a, we can differentiate with respect to Σ_{opt}^a and equate to 0, obtaining

$$-{\Sigma_{opt}^a}^{-1} + \bar{P}^{-1} = 0 \implies \Sigma_{opt}^a = \bar{P}. \tag{2.78}$$

Similarly, the minimizing μ_k^a can be obtained by solving

$$\mu_{k\,opt}^a = \arg\min_{\mu_k^a} \mathbb{E}\left[\nu_k^{aT} \bar{P}^{-1} \nu_k^a | \mathcal{I}_k^a \right], \tag{2.79}$$

where $\mathbb{E}\left[\nu_k^{aT} \bar{P}^{-1} \nu_k^a | \mathcal{I}_k^a \right]$ is given by

$$\int_{\hat{x}_{k|k-1}} \nu_k^{aT} \bar{P}^{-1} \nu_k^a f(\hat{x}_{k|k-1} | \mathcal{I}_k^a) d\hat{x}_{k|k-1}.$$

Differentiating with respect to μ_k^a, we obtain

$$\int_{\hat{x}_{k|k-1}} \bar{P}^{-1}(\mu_{k\,opt}^a - C\hat{x}_{k|k-1}) f(\hat{x}_{k|k-1} | \mathcal{I}_k^a) d\hat{x}_{k|k-1} = 0.$$

Solving for μ_k^a, we have $\mu_{k\,opt}^a = C\mathbb{E}\left[\hat{x}_{k|k-1} | \mathcal{I}_k^a \right]$. □

Here, the KL divergence is used as a heuristic to represent the distance between distributions. From Theorem 2.8, the attacker should generate stealthy virtual inputs y_k^a as follows.

Virtual Output Generation

1. Calculate $C\mathbb{E}[\hat{x}_{k|k-1} | \mathcal{I}_k^a]$.
2. Generate IID noise $\zeta_k \sim \mathcal{N}(0, \bar{P})$.
3. Compute $y_k^a = C\mathbb{E}[\hat{x}_{k|k-1} | \mathcal{I}_k^a] + \zeta_k$.

Thus, in an optimal solution, an attacker computes a best approximation of the output a defender would expect to see and then adds noise of the appropriate distribution, in this case the distribution of the residue. For the remainder of the section, we determine how the attacker should use all available information to compute $C\mathbb{E}[\hat{x}_{k|k-1}|\mathcal{I}_k^a]$. In order to generate y_{k+1}^a, the attacker at time $k+1$ has knowledge of the outputs y_j, and control inputs u_j^1 up to time k. That is,

$$\mathcal{I}_{k+1}^a = \{\mathcal{M}, y_k, y_k^a, u_k^a, u_k^1, y_{k-1}, y_{k-1}^a, u_{k-1}^a, u_{k-1}^1, \ldots\}. \tag{2.80}$$

Remark 2.18. *We have assumed that at time k, to generate, y_k^a, the attacker does not have the ability to incorporate the real time y_k into his estimate. This perhaps might be a real time constraint for an attacker who does not wish to introduce suspicious delays into the system by processing real time sensor measurements.*

To obtain a conditional estimate of $\hat{x}_{k+1|k}$, using $\mathcal{I}_{k+1}^a$, we formulate a new model from the attacker's perspective. Suppose an adversary has information $\mathcal{I}_{k+1}^a$. Furthermore define,

$$y_k^u \triangleq u_k^1 - L^1 K y_k^a. \tag{2.81}$$

Given y_k^a, which is known by the attacker, y_k^u is an invertible function of u_k^1. Thus, $\mathcal{I}_{k+1}^a$ can be rewritten as

$$\mathcal{I}_{k+1}^a = \{\mathcal{M}, y_k, y_k^a, u_k^a, y_k^u, y_{k-1}, y_{k-1}^a, u_{k-1}^a, y_{k-1}^u, \ldots\}. \tag{2.82}$$

Lemma 2.1. *For $k \geq 0$, the attacker's observations can be formulated as the outputs of a state space model as follows.*

$$\begin{bmatrix} \hat{x}_{k+1|k} \\ x_{k+1} \end{bmatrix} = \mathcal{A} \begin{bmatrix} \hat{x}_{k|k-1} \\ x_k \end{bmatrix} + \mathcal{B} \begin{bmatrix} y_k^a \\ y_k^u + L^1 K y_k^a \\ u_k^a \end{bmatrix} + \mathcal{W}_k, \quad \begin{bmatrix} y_k^u \\ y_k \end{bmatrix} = \mathcal{C} \begin{bmatrix} \hat{x}_{k|k-1} \\ x_k \end{bmatrix} + \mathcal{V}_k, \tag{2.83}$$

where

$$\mathcal{A} \triangleq \begin{bmatrix} (A+B^2L^2)(I-KC) & 0 \\ B^2L^2(I-KC) & A \end{bmatrix}, \mathcal{B} \triangleq \begin{bmatrix} (A+B^2L^2)K & B^1 & 0 \\ B^2L^2K & B^1 & B^a \end{bmatrix},$$

$$\mathcal{W}_k \triangleq \begin{bmatrix} B^2\Delta u_k^2 \\ B^2\Delta u_k^2 + w_k \end{bmatrix} \quad \mathcal{V}_k \triangleq \begin{bmatrix} \Delta u_k^1 \\ v_k \end{bmatrix}, \ \mathcal{W}_k \sim \mathcal{N}(0, \mathcal{Q}), \ \mathcal{V}_k \sim \mathcal{N}(0, \mathcal{R}), \mathcal{W}_k, \mathcal{V}_k \textit{ IID}.$$

$$\mathcal{Q} \triangleq \begin{bmatrix} B^2\mathcal{J}^2B^{2^T} & B^2\mathcal{J}^2B^{2^T} \\ B^2\mathcal{J}^2B^{2^T} & B^2\mathcal{J}^2B^{2^T} + Q \end{bmatrix}, \mathcal{R} \triangleq \begin{bmatrix} \mathcal{J}^1 & 0 \\ 0 & R \end{bmatrix}, \ \mathcal{C} \triangleq \begin{bmatrix} L^1(I-KC) & 0 \\ 0 & C \end{bmatrix}$$

Proof. From (2.4), for $k \geq 0$, when the attacker inserts virtual outputs and external inputs we have

$$\hat{x}_{k+1|k} = A(I-KC)\hat{x}_{k|k-1} + Bu_k + AKy_k^a. \tag{2.84}$$

From (2.69), and (2.4), we have

$$Bu_k = B^1u_k^1 + B^2\left(L^2(I-KC)\hat{x}_{k|k-1} + L^2Ky_k^a + \Delta u_k^2\right), \tag{2.85}$$

so that the first state equation immediately follows. The second state equation is trivially obtained from the dynamic equation (2.1), (2.85), and the attacker's external input $B^au_k^a$. From (2.69) and (2.4),

$$u_k^1 = L^1(I-KC)\hat{x}_{k|k-1} + L^1Ky_k^a + \Delta u_k^1, \tag{2.86}$$

thus arriving at the first output equation. The second output equation is identical to (2.2). Finally, the noise distributions are easily derived from (2.70), and the process and sensor noise statistics. □

We remark that for $k < 0$, the same state equations hold for the attacker, except that the external input $u_k^a = 0$ and the virtual output y_k^a is simply the true output y_k. Therefore, before executing his attack, the attacker can still observe the model up to time $k = 0$ to obtain the best possible estimate of the state. Secondly, from Remark 2.16, we note that a subset of outputs may not be useful to the attacker due to instability or nonlinearities in the plant. The attacker can ignore a subset of sensors by removing rows from C when defining $\mathcal{C}$. Before deriving a filter to obtain an optimal estimate $\mathbb{E}[\hat{x}_{k|k-1}|\mathcal{I}_k^a]$, we make the following assumptions.

Assumption 2.2.1. *$(A+BL)(I-KC)$ is stable. From [27], the stability of $(A+BL)(I-KC)$ is a standard assumption in watermarking algorithms since otherwise detecting the adversary is trivial. It can be easily shown that the stability $(A+BL)(I-KC)$ implies that $(\mathcal{A},\mathcal{C})$ is detectable.*

Assumption 2.2.2. *$\mathcal{A}$ has no eigenvalues on the unit circle.*

Theorem 2.9. *Suppose the adversary starts observing the system at time $k=-N$, where $N>0$. Assume at $k=-N$*

$$f\left(\begin{bmatrix}\hat{x}_{k|k-1}\\ x_k\end{bmatrix}\mid \mathcal{I}_k^a\right)\sim\mathcal{N}\left(\begin{bmatrix}\bar{\hat{x}}\\ \bar{x}\end{bmatrix},\Sigma\right) \tag{2.87}$$

where $\Sigma>0$.

Define

$$\tilde{x}_{k|k-1}\triangleq\mathbb{E}[\hat{x}_{k|k-1}|\mathcal{I}_k^a],\ \tilde{x}_k\triangleq\mathbb{E}[x_k|\mathcal{I}_k^a]. \tag{2.88}$$

Then, $\tilde{x}_{k|k-1}$ and $\tilde{x}_k$ satisfies the following recursive filter.

$$\text{For } k=-N:\quad \begin{bmatrix}\tilde{x}_{k|k-1}\\ \tilde{x}_k\end{bmatrix}=\begin{bmatrix}\bar{\hat{x}}\\ \bar{x}\end{bmatrix},\quad \mathcal{P}_k=\Sigma>0, \tag{2.89}$$

$$\text{For } 0>k\geq -N:\quad \begin{bmatrix}\tilde{x}_{k+1|k}\\ \tilde{x}_{k+1}\end{bmatrix}=\mathcal{A}(I-\mathcal{K}_k\mathcal{C})\begin{bmatrix}\tilde{x}_{k|k-1}\\ \tilde{x}_k\end{bmatrix}+\mathcal{A}\mathcal{K}_k\begin{bmatrix}y_k^u\\ y_k\end{bmatrix}+\mathcal{B}\begin{bmatrix}y_k\\ u_k^1\\ 0\end{bmatrix} \tag{2.90}$$

where

$$\mathcal{K}_k=\mathcal{P}_k\mathcal{C}^T(\mathcal{C}\mathcal{P}_k\mathcal{C}^T+\mathcal{R})^{-1}, \tag{2.91}$$

and the covariance satisfies

$$\mathcal{P}_{k+1}=\mathcal{A}\mathcal{P}_k\mathcal{A}^T+\mathcal{Q}-\mathcal{A}\mathcal{P}_k\mathcal{C}^T(\mathcal{C}\mathcal{P}_k\mathcal{C}^T+\mathcal{R})^{-1}\mathcal{C}\mathcal{P}_k\mathcal{A}^T. \tag{2.92}$$

Assume $N \to \infty$. Define $\mathcal{P} \triangleq \lim_{j\to\infty} \mathcal{P}_j$, where $\mathcal{P}_j$ is recursively defined according to (2.92). Then, for $k \geq 0$, $\tilde{x}_{k|k-1}$ and $\tilde{x}_k$ satisfy the following recursive filter

$$\begin{bmatrix} \tilde{x}_{k+1|k} \\ \tilde{x}_{k+1} \end{bmatrix} = \mathcal{A}(I - \mathcal{K}\mathcal{C}) \begin{bmatrix} \tilde{x}_{k|k-1} \\ \tilde{x}_k \end{bmatrix} + \mathcal{A}\mathcal{K} \begin{bmatrix} y_k^u \\ y_k \end{bmatrix} + \mathcal{B} \begin{bmatrix} y_k^a \\ u_k^1 \\ u_k^a \end{bmatrix}, \tag{2.93}$$

where

$$\mathcal{K} = \mathcal{P}\mathcal{C}^T(\mathcal{C}\mathcal{P}\mathcal{C}^T + \mathcal{R})^{-1}. \tag{2.94}$$

Additionally, $\mathcal{A}(I - \mathcal{K}\mathcal{C})$ is Schur stable.

Proof. For $k < 0$, the proof follows from the definition of the standard Kalman filter, [31] and is thus not reported. Observe from assumption 2.2.1, $(\mathcal{A}, \mathcal{C})$ is detectable. Moreover, from assumption 2.2.2, $(\mathcal{A}, \mathcal{Q}^{\frac{1}{2}})$ has no uncontrollable eigenvalues on the unit circle since $\mathcal{A}$ has no eigenvalues on the unit circle. This combined with the fact that $\mathcal{R} > 0$ and $\Sigma > 0$ implies that $\mathcal{P}_k$ converges to the unique stabilizing solution X of the riccati equation (2.95), [32].

$$X = \mathcal{A}X\mathcal{A}^T + \mathcal{Q} - \mathcal{A}X\mathcal{C}^T(\mathcal{C}X\mathcal{C}^T + \mathcal{R})^{-1}\mathcal{C}X\mathcal{A}^T. \tag{2.95}$$

This implies $\mathcal{A}(I - \mathcal{K}\mathcal{C})$ is Schur stable. The proof for $k \geq 0$ again follows from the definition of the standard Kalman filter. □

2.2.4 Attack Detection

In this section, we propose a Neyman Pearson detector to determine whether an attack has occurred. To begin, we would like to characterize the distribution of the stealthy y_k^a generated by the attacker from the defender's perspective. Unlike the attacker, the defender has full knowledge of the state estimate $\hat{x}_{k|k-1}$ and watermarks $\Delta u_k^1, \Delta u_k^2$. However, the defender does not have access to the true y_k or u_k^a. As such, the defender can not directly calculate the attacker's state estimate $\tilde{x}_{k|k-1}$. Nonetheless, it can characterize the distribution of $\tilde{x}_{k|k-1}$ and thus y_k^a in general.

We let $\mathcal{I}_k$ represent the **reliable** information available to the system operator under attack. Here, we will ignore all sensor information as that may be corrupted by the attacker. Thus, we say

$$\mathcal{I}_k = \{\mathcal{M}, \hat{x}_{k|k-1}, u_{k-1}, \Delta u^1_{k-1}, \Delta u^2_{k-1}, \hat{x}_{k-1|k-2}, u_{k-2}, \Delta u^1_{k-2}, \Delta u^2_{k-2}, \cdots\} \tag{2.96}$$

Lemma 2.2. *Assume at $k = -N$,*

$$f\left(\begin{bmatrix}\hat{x}_{k|k-1} - \bar{\hat{x}} \\ x_k - \bar{x}\end{bmatrix} | \mathcal{I}_k\right) \sim \mathcal{N}\left(\begin{bmatrix}0\\0\end{bmatrix}, \bar{\Sigma}\right). \tag{2.97}$$

Let $N \to \infty$. For $k \geq 0$, we have

$$f(y^a_k | \mathcal{I}_k) \sim \mathcal{N}_1\left(C(\hat{x}_{k|k-1} - \epsilon_{k|k-1}), \mathcal{P}_y + \bar{P}\right), \tag{2.98}$$

where $\epsilon_{k|k-1}$ satisfies the recursive filter $\epsilon_{-N|-N-1} = 0,\ \epsilon_{-N} = 0$. For $k < 0$

$$\begin{bmatrix}\epsilon_{k+1|k} \\ \epsilon_{k+1}\end{bmatrix} = \mathcal{A}\left(I - \mathcal{K}_k\mathcal{C}\right)\begin{bmatrix}\epsilon_{k|k-1} \\ \epsilon_k\end{bmatrix} - \mathcal{A}\mathcal{K}_k\begin{bmatrix}\Delta u^1_k \\ 0\end{bmatrix} + \begin{bmatrix}B^2 \\ B^2\end{bmatrix}\Delta u^2_k. \tag{2.99}$$

For $k \geq 0$,

$$\begin{bmatrix}\epsilon_{k+1|k} \\ \epsilon_{k+1}\end{bmatrix} = \mathcal{A}\left(I - \mathcal{K}\mathcal{C}\right)\begin{bmatrix}\epsilon_{k|k-1} \\ \epsilon_k\end{bmatrix} - \mathcal{A}\mathcal{K}\begin{bmatrix}\Delta u^1_k \\ 0\end{bmatrix} + \begin{bmatrix}B^2 \\ B^2\end{bmatrix}\Delta u^2_k. \tag{2.100}$$

Moreover, $\mathcal{P}_y$ is defined as

$$\mathcal{P}_y \triangleq \begin{bmatrix}C & 0\end{bmatrix}\mathcal{P}_\epsilon\left(\begin{bmatrix}C & 0\end{bmatrix}\right)^T, \tag{2.101}$$

and $\mathcal{P}_\epsilon$ satisfies the following Lyapunov equation,

$$\mathcal{P}_\epsilon = \mathcal{A}\left(I - \mathcal{K}\mathcal{C}\right)\mathcal{P}_\epsilon\left(\mathcal{A}\left(I - \mathcal{K}\mathcal{C}\right)\right)^T + \mathcal{U}, \tag{2.102}$$

where

$$\mathcal{U} = \mathcal{A}\mathcal{K}\begin{bmatrix}0 & 0\\ 0 & R\end{bmatrix}\mathcal{K}^T\mathcal{A}^T + \begin{bmatrix}0 & 0\\ 0 & Q\end{bmatrix}. \tag{2.103}$$

Proof. To begin, we define

$$\tilde{e}_{k|k-1} \triangleq \hat{x}_{k|k-1} - \tilde{x}_{k|k-1},\ \tilde{e}_k \triangleq x_k - \tilde{x}_k. \tag{2.104}$$

From (2.83) and (2.90), the error dynamics of the attacker's estimation filter during attack for $-N < k < 0$ are

$$\begin{bmatrix} \tilde{e}_{k+1|k} \\ \tilde{e}_{k+1} \end{bmatrix} = \mathcal{A}\left(I - \mathcal{K}_k\mathcal{C}\right)\begin{bmatrix} \tilde{e}_{k|k-1} \\ \tilde{e}_k \end{bmatrix} - \mathcal{A}\mathcal{K}_k\mathcal{V}_k + \mathcal{W}_k. \tag{2.105}$$

Rearranging terms we have

$$\begin{bmatrix} \tilde{e}_{k+1|k} \\ \tilde{e}_{k+1} \end{bmatrix} = \mathcal{A}\left(I - \mathcal{K}_k\mathcal{C}\right)\begin{bmatrix} \tilde{e}_{k|k-1} \\ \tilde{e}_k \end{bmatrix} - \mathcal{A}\mathcal{K}_k\begin{bmatrix} 0 \\ v_k \end{bmatrix} + \begin{bmatrix} 0 \\ w_k \end{bmatrix} - \mathcal{A}\mathcal{K}_k\begin{bmatrix} \Delta u_k^1 \\ 0 \end{bmatrix} + \begin{bmatrix} B^2 \\ B^2 \end{bmatrix}\Delta u_k^2. \tag{2.106}$$

As $N \to \infty$, for $k \geq 0$, we have,

$$\begin{bmatrix} \tilde{e}_{k+1|k} \\ \tilde{e}_{k+1} \end{bmatrix} = \mathcal{A}\left(I - \mathcal{K}\mathcal{C}\right)\begin{bmatrix} \tilde{e}_{k|k-1} \\ \tilde{e}_k \end{bmatrix} - \mathcal{A}\mathcal{K}\begin{bmatrix} 0 \\ v_k \end{bmatrix} + \begin{bmatrix} 0 \\ w_k \end{bmatrix} - \mathcal{A}\mathcal{K}\begin{bmatrix} \Delta u_k^1 \\ 0 \end{bmatrix} + \begin{bmatrix} B^2 \\ B^2 \end{bmatrix}\Delta u_k^2. \tag{2.107}$$

Since the states $\tilde{e}_{k|k-1}$ and $\tilde{e}_k$ initially have a normal distribution given $\mathcal{I}_k$ and the system is linear with IID Gaussian noise, for each k, $\tilde{e}_{k|k-1}$ and $\tilde{e}_k$ has a normal distribution given $\mathcal{I}_k$. Let

$$\epsilon_{k|k-1} \triangleq \mathbb{E}[\tilde{e}_{k|k-1}|\mathcal{I}_k] = \hat{x}_{k|k-1} - \mathbb{E}[\tilde{x}_{k|k-1}|\mathcal{I}_k], \quad \epsilon_k \triangleq \mathbb{E}[\tilde{e}_k|\mathcal{I}_k]. \tag{2.108}$$

Taking the expected value of (2.106) and (2.107), we obtain (2.99) and (2.100). Noting that y_k^a has expected value $C\mathbb{E}[\tilde{x}_{k|k-1}|\mathcal{I}_k]$, we see that

$$\mathbb{E}[y_k^a|\mathcal{I}_k] = C(\hat{x}_{k|k-1} - \epsilon_{k|k-1}). \tag{2.109}$$

Next observe that (2.107) is an unobserved dynamical system from the defender's perspective. From the convergence of the gain $\mathcal{K}_k$ as $N \to \infty$, and the stability $\mathcal{A}(I - \mathcal{K}\mathcal{C})$, the covariance of $\begin{bmatrix} \tilde{e}_{k|k-1}^T & \tilde{e}_k^T \end{bmatrix}^T$ for $k \geq 0$ simply satisfies the Lyapunov equation (2.102) and is thus $\mathcal{P}_\epsilon$. From the attack strategy, to compute y_k^a, the attacker calculates $C\tilde{x}_{k|k-1} + \zeta_k$ where $\zeta_k \sim \mathcal{N}(0, \bar{P})$. From here, the covariance of y_k^a is simply $C\,\mathrm{Cov}(\tilde{x}_{k|k-1})C^T + \bar{P} = \mathcal{P}_y + \bar{P}$. Since, $\tilde{e}_{k|k-1}$ has a normal distribution given $\mathcal{I}_k$, y_k^a has a normal distribution given $\mathcal{I}_k$ and the result holds. □

Remark 2.19. *We remark that in practice the defender will be unaware of the time $-N$ an attacker begins observing a system. However, due to the stability of $\mathcal{A}(I - \mathcal{KC})$, the effect of the chosen $-N$ on the distribution of y_k^a asymptotically vanishes.*

Using the results of the previous theorem, we can characterize the distribution of our residue when the system is under attack and when the system is under normal operation. Namely, we can redefine our null and alternative hypotheses as follows.

$$\mathcal{H}_0 : z_k \sim \mathcal{N}(0, \bar{P}), \quad \mathcal{H}_1 : z_k \sim \mathcal{N}(-C\epsilon_{k|k-1}, \mathcal{P}_y + \bar{P}).$$

In this case, the optimal detector which maximizes the probability of detection β_k for a given probability of false alarm α is a Neyman Pearson threshold detector (see [33]).

Theorem 2.10. *The most powerful test for the hypothesis test $\mathcal{H}_1$ vs. $\mathcal{H}_0$ is $g_{NP}(z_k) \underset{\mathcal{H}_0}{\overset{\mathcal{H}_1}{\gtrless}} \eta$, where $g_{NP}(z_k)$ is given by*

$$z_k^T \bar{P}^{-1} z_k - (z_k + C\epsilon_{k|k-1})^T \left(\mathcal{P}_y + \bar{P}\right)^{-1} (z_k + C\epsilon_{k|k-1}).$$

In general, we would like to maximize our detection rate using the degrees of freedom we have in the design of our watermark. However, directly maximizing the probability of detection is difficult because it involves integrating a Gaussian function. In the next subsection, we propose a relaxed optimization problem to design a robust watermark.

2.2.5 Watermark Design

In this section, we attempt to maximize the probability we can detect an attack through the design of our watermark, namely the covariances $\mathcal{J}_1$ and $\mathcal{J}_2$. Qualitatively, large covariances would increase the attacker's uncertainty about the watermark, making statistically sound y_k^a's difficult to generate. However, large watermarks also increase the cost to the system (2.3). As such, we would like to bound the additional cost created by the watermark. As discussed in the prior section, the optimal LQG cost of the system without the watermark J^* is given by

$$J^* = \text{tr}(SQ) + \text{tr}((A^T SA + W - S)(P - KCP)). \tag{2.110}$$

With the watermark, the cost J from [27] is given by $J = J^* + \Delta J$ where

$$\begin{aligned}\Delta J &= \text{tr}((B^T S B + U)\mathcal{J}), \\ &= \text{tr}((B^{1^T} S B^1 + U^1)\mathcal{J}^1 + (B^{2^T} S B^2 + U^2)\mathcal{J}^2).\end{aligned} \tag{2.111}$$

Because it is difficult to directly maximize the probability of detection, we would like to maximize the distance between distributions of our residues so that they are easier to distinguish. To obtain a concave metric, we select the expected KL divergence between $z_k \sim \mathcal{N}_3(C(\tilde{x}_{k|k-1} - \hat{x}_{k|k-1}), \bar{P})$, the residue generated under attack, and $z_k \sim \mathcal{N}_2(0, \bar{P})$ the residue generated under normal operation given the attacker's information. From Theorem 2.8 and (2.76) the expected KL divergence is given by

$$\mathbb{E}[D_{KL}(\mathcal{N}_3||\mathcal{N}_2)|\mathcal{I}_k^a] = \frac{1}{2}\,\text{tr}\left(\mathcal{P}_{\hat{x}} C^T \bar{P}^{-1} C\right), \tag{2.112}$$

where

$$\mathcal{P}_{\hat{x}} = \begin{bmatrix} I & 0 \end{bmatrix} \mathcal{P} \begin{bmatrix} I \\ 0 \end{bmatrix} \tag{2.113}$$

Thus, we arrive at the following optimization problem to design the covariances of our watermark.

Problem 1

$$\begin{aligned} \underset{\mathcal{J}_1, \mathcal{J}_2, \mathcal{P}}{\text{maximize}} \quad & \text{tr}\left(\mathcal{P}_{\hat{x}} C^T \bar{P}^{-1} C\right) \\ \text{subject to} \quad & \mathcal{P}, \mathcal{J}_2 \geq 0, \quad \mathcal{J}_1 \geq \epsilon I \\ & \text{tr}((B^{1^T} S B^1 + U^1)\mathcal{J}^1 + (B^{2^T} S B^2 + U^2)\mathcal{J}^2) \leq \delta \\ & \mathcal{P} = \mathcal{A}\mathcal{P}\mathcal{A}^T + \mathcal{Q} - \mathcal{A}\mathcal{P}\mathcal{C}^T(\mathcal{C}\mathcal{P}\mathcal{C}^T + \mathcal{R})^{-1}\mathcal{C}\mathcal{P}\mathcal{A}^T \\ & \mathcal{P}_{\hat{x}} = \begin{bmatrix} I & 0 \end{bmatrix} \mathcal{P} \begin{bmatrix} I \\ 0 \end{bmatrix}, \ \mathcal{P} \text{ is the stabilizing solution of the riccati equation.} \end{aligned}$$

In the above formulation, we are maximizing a linear function in $\mathcal{P}$ with a cost constraint convex in $\mathcal{J}_1, \mathcal{J}_2$. However, the Riccati constraint is not convex and it is not entirely obvious how to enforce

$\mathcal{P}$ to be the stabilizing solution. As a result, we consider the following essentially equivalent convex optimization problem.

Problem 2

$$\begin{aligned}
\underset{\mathcal{J}_1,\mathcal{J}_2,\mathcal{P}}{\text{maximize}} \quad & \operatorname{tr}\left(\mathcal{P}_{\hat{x}} C^T \bar{P}^{-1} C\right) \\
\text{subject to} \quad & \mathcal{P}, \mathcal{J}_2 \geq 0, \quad \mathcal{J}_1 \geq \epsilon I \\
& \operatorname{tr}((B^{1^T} S B^1 + U^1)\mathcal{J}^1 + (B^{2^T} S B^2 + U^2)\mathcal{J}^2) \leq \delta \\
& \begin{bmatrix} \mathcal{A}\mathcal{P}\mathcal{A}^T + \mathcal{Q} - \mathcal{P} & \mathcal{A}\mathcal{P}\mathcal{C}^T \\ \mathcal{C}\mathcal{P}\mathcal{A}^T & \mathcal{C}\mathcal{P}\mathcal{C}^T + \mathcal{R} \end{bmatrix} \geq 0, \mathcal{P}_{\hat{x}} = \begin{bmatrix} I & 0 \end{bmatrix} \mathcal{P} \begin{bmatrix} I \\ 0 \end{bmatrix}.
\end{aligned}$$

We now have the following result.

Theorem 2.11. *Let $(\mathcal{J}^1_{2_{opt}}, \mathcal{J}^2_{2_{opt}})$ be a maximizing solution to Problem 2. Then, $(\mathcal{J}^1_{2_{opt}}, \mathcal{J}^2_{2_{opt}})$ is also a maximizing solution to problem 1.*

Proof. In Problem 2, the semidefinite constraint under the condition that $\mathcal{C}\mathcal{P}\mathcal{C}^T + \mathcal{R} > 0$ is equivalent to

$$\mathcal{P} \leq \mathcal{A}\mathcal{P}\mathcal{A}^T + \mathcal{Q} - \mathcal{A}\mathcal{P}\mathcal{C}^T(\mathcal{C}\mathcal{P}\mathcal{C}^T + \mathcal{R})^{-1}\mathcal{C}\mathcal{P}\mathcal{A}^T, \tag{2.114}$$

by Schur's complement condition for positive definiteness. Let $(\mathcal{J}^1_{2_{opt}}, \mathcal{J}^2_{2_{opt}}, \mathcal{P}_{2_{opt}})$ be an optimal solution for Problem 2 and $(\mathcal{J}^1_{1_{opt}}, \mathcal{J}^2_{1_{opt}}, \mathcal{P}_{1_{opt}})$ be an optimal solution for problem 1. From (2.114), the feasible set of solutions for Problem 1 are a subset of the feasible solutions for Problem 2. Letting $\mathcal{P}_{\hat{x}_1} = \mathcal{P}_{\hat{x}}(\mathcal{J}^1_{1_{opt}}, \mathcal{J}^2_{1_{opt}}, \mathcal{P}_{1_{opt}})$ and $\mathcal{P}_{\hat{x}_2} = \mathcal{P}_{\hat{x}}(\mathcal{J}^1_{2_{opt}}, \mathcal{J}^2_{2_{opt}}, \mathcal{P}_{2_{opt}})$, we thus have

$$\operatorname{tr}\left(\mathcal{P}_{\hat{x}_2} C^T \bar{P}^{-1} C\right) \geq \operatorname{tr}\left(\mathcal{P}_{\hat{x}_1} C^T \bar{P}^{-1} C\right). \tag{2.115}$$

From Theorem 13.1.1 in [34], for the set of positive semidefinite $\mathcal{P}$ satisfying (2.114), we have $\mathcal{P}_s \geq \mathcal{P}$ where $\mathcal{P}_s$ is the stabilizing solution of the discrete algebraic riccati equation with watermark covariances $\mathcal{J}^1_{2_{opt}}, \mathcal{J}^2_{2_{opt}}$. Since the objective function is monotone increasing in $\mathcal{P}$ this implies that

$$\operatorname{tr}\left(\mathcal{P}_{\hat{x}}(\mathcal{J}^1_{2_{opt}}, \mathcal{J}^2_{2_{opt}}, \mathcal{P}_s) C^T \bar{P}^{-1} C\right) \geq \operatorname{tr}\left(\mathcal{P}_{\hat{x}_2} C^T \bar{P}^{-1} C\right). \tag{2.116}$$

However, $(\mathcal{J}^1_{2_{opt}}, \mathcal{J}^2_{2_{opt}}, \mathcal{P}_s)$ lies in the feasible set of problem 1. As a result, from (2.115) and (2.116),

$$\text{tr}\left(\mathcal{P}_{\hat{x}_2} C^T \bar{P}^{-1} C\right) = \text{tr}\left(\mathcal{P}_{\hat{x}_1} C^T \bar{P}^{-1} C\right). \tag{2.117}$$

Therefore, $(\mathcal{J}^1_{2_{opt}}, \mathcal{J}^2_{2_{opt}}, \mathcal{P}_s)$ is a solution to problem 1 and the result holds. □

Note, since we consider IID watermarks, it is clear that the watermarking sequence is not a sinusoid like the previous section. An interesting future problem is to determine whether a stationary robust watermarking design generated from a hidden Markov model will induce an optimal watermark that is a sinusoid.

We conclude this section by noting that the defender can use this optimization problem to select which inputs he wishes to secure from the attacker. That is, he can optimize his choice of B^1 and B^2 subject to some constraints on the number of inputs he wishes to keep secret or the identities of inputs which he can keep secret. The optimization problem however becomes combinatorial in nature. Nonetheless, for a small number of inputs the problem remains feasible. Moreover, the problem only needs to be solved once prior to the system deployment.

2.2.6 Numerical Example

We consider a randomly generated system with $n = 10$ states, $p = 8$ inputs and $m = 7$ sensors. The matrices A,B, and C are uniform sparsely generated matrices with density 0.3. Moreover Q, R, U, W were each chosen to be the identity. The optimal cost for the system is $J^* = 25.7$. The matrix $(A + BL)(I - KC)$ is stable, thus motivating the use of a watermark.

We consider four separate scenarios in our system. In three scenarios, we utilize the watermarking design scheme proposed in this section and seen in Problem 2. In these scenarios we vary the number of inputs the attacker can see from 1 input, to 4 inputs, to 7 inputs. The more inputs the attacker can see, the better he can estimate the defender's state estimate. Moreover, in our fourth scenario, we consider the case where the attacker knows only 1 input, but the defender uses the

watermarking scheme seen in [27]. In this case, the watermarks Δu_k^1 and Δu_k^2 are correlated. As such, the attacker can estimate u_k^2.

In Fig. 2.13 and Fig. 2.14, we plot the asymptotic probability of detection as a function of the probability of false alarm, where we consider small values of α in Fig. 2.14. Here $\Delta J = 5$, meaning that the additional cost is roughly 20% of the optimal cost. It can be seen that the proposed approach offers increased security over the approach in [27], when the inputs are compromised. Moreover, in this example, knowledge of a single input allows the attacker to fool a detector with the replay watermarking design [27]. The probability of detection is roughly equal to the probability of false alarm and thus the detector asymptotically provides little to no information about whether an attack has taken place. In Fig. 2.15, we plot the asymptotic probability of detection as a function of the additional cost for fixed $\alpha = 0.1$. For the proposed watermarking scheme, increasing the magnitude of the watermark, and thereby the cost, improves the probability of detection. However, for the previous design, [27] the additional cost does not aid in detection.

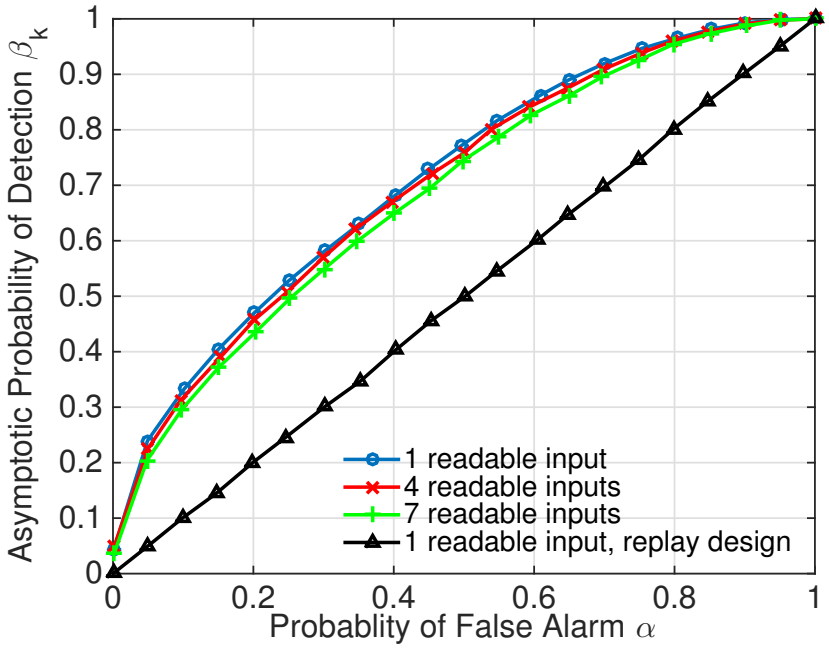


Figure 2.13: Robust Watermark, Asymptotic probability of detection β_k vs probability of false alarm α, $\Delta J = 5$

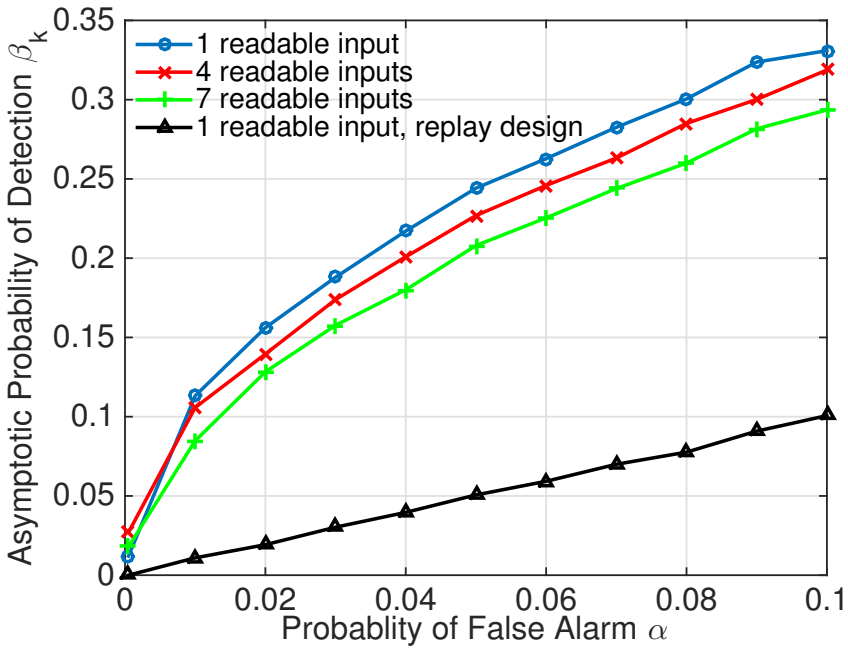


Figure 2.14: Robust Watermark, Asymptotic probability of detection β_k vs probability of false alarm α for small α, $\Delta J = 5$

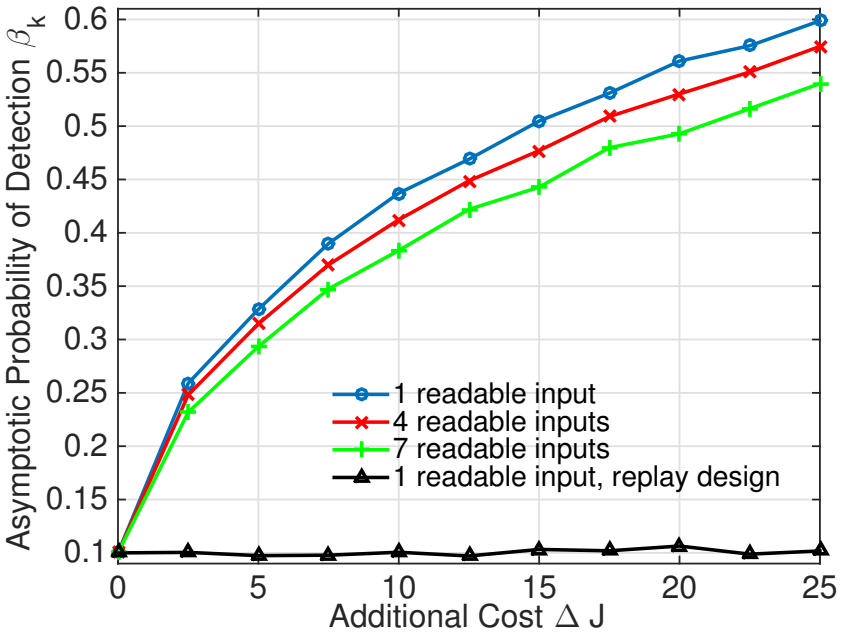


Figure 2.15: Robust Watermark, Asymptotic probability of detection β_k vs additional cost ΔJ, $\alpha = 0.1$

Chapter 3

Environmental Watermarks

In the previous chapter we demonstrated the effectiveness of physical watermarking in detecting several classes of integrity attacks. The main mechanism of detection in the use of physical watermarks is a source of randomness that a defender knows, in that case an additive element to the control input, which the attacker does not. In this chapter, we make the observation that such sources of randomness may not have to be introduced by the defender, but might simply be a product of the environment. As an example, a defender might have side information about the process or sensor noise in a system, which can in turn be leveraged to detect an attacker who does not have this same information. In this chapter, we will specifically examine how packet drops at the control input can serve as an environmental watermark. In section 3.1 we will introduce the idea of a packet drop watermark, which can occur environmentally or as a result of intended action by the defender. In section 3.2 we will extend this work to consider a joint Gaussian and packet drop watermark. The results in this chapter are largely based on [35] and [36].

3.1 A Packet Drop Watermark

In this section, we consider how packet drops can serve as an environmental or naturally occurring watermark that allows us to actively detect malicious adversaries. Packet drops occur naturally

in the context of networked control systems. In particular, both command and measurement channels could be subjected to packet drops due to, e.g., imperfections at the wireless and/or wired communication networks [37, 38]. Packet drops at the command and measurement channels change the system dynamics in a specific form, see e.g. [10, 9]. In this section, we view the packet drops as a means to create watermarked dynamics and we explore the possibility to authenticate the system via intentional packet drop injections.

We assume there exists independent and identically distributed packet drops at the channel to the actuators with certain probability. This already occurs naturally and can be intentionally introduced by a defender to enhance security. Such a mechanism is easy to implement using, e.g., switches and pulses and they are applicable for a wide range of applications. We will next evaluate the benefits of packet drops in terms of detecting stealthy attackers with high probability.

3.1.1 System Description

As in the previous chapter, we model the system using discrete time linear time invariant (LTI) dynamics. However, here we model packet drops at the control input.

$$x_{k+1} = Ax_k + \eta_k Bu_k + w_k, \tag{3.1}$$

$$y_k = Cx_k + v_k. \tag{3.2}$$

Again, $x_k \in \mathbb{R}^n$ is the state vector at time k, $u_k \in \mathbb{R}^p$ is the control input at time k, and $y_k \in \mathbb{R}^m$ denotes sensor measurements taken at time k. In the model, $w_k \sim \mathcal{N}(0, Q)$ is IID process noise and $v_k \sim \mathcal{N}(0, R)$ is IID measurement noise. We assume that (A, C) is detectable and $R > 0$. Moreover, (A, B) and $(A, Q^{\frac{1}{2}})$ are stabilizable.

We now consider $\eta_k \in \{0, 1\}$ which is an independent identically distributed (IID) packet drop process generated at the controller and known at the actuator and the estimator. Here, $\eta_k = 0$ indicates a packet drop and $\Pr(\eta_k = 0) = p_d$ is the packet drop probability. If the packet drops are not introduced intentionally by the defender, but occur naturally due to the environment, we would

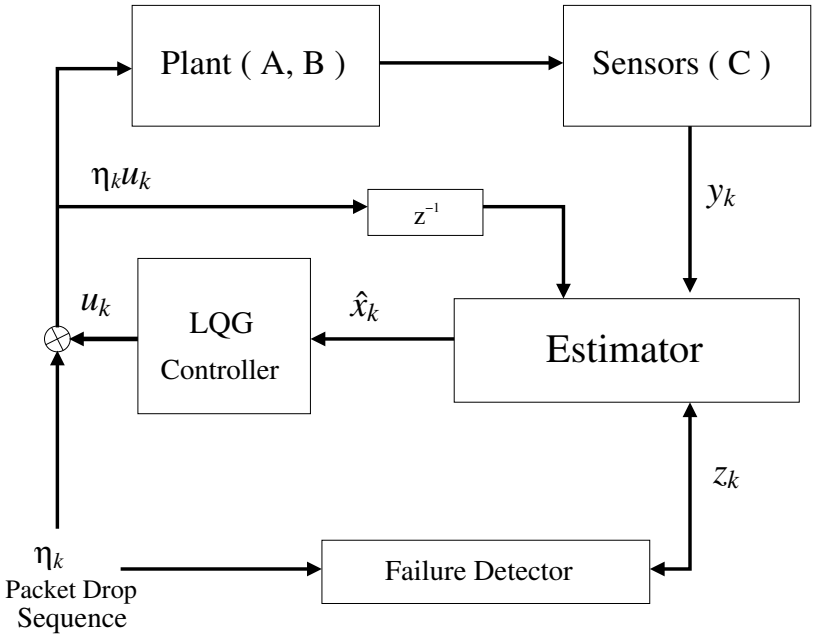


Figure 3.1: System model under normal operation. When a replay attack occurs, the attacker replaces the output y_k with its time lagged version. The plant input may also be compromised.

assume the CPS utilizes a TCP like protocol, where the defender receives acknowledgements when a control packet is successfully delivered to the plant. An illustrative diagram is found in Fig. 3.1

3.1.2 LQG Control with Packet Drops

Let us assume that the following information set $\mathcal{I}_k = \{\mathcal{M}, y_{-\infty:k}, u_{-\infty:k-1}, \eta_{-\infty:k-1}\}$ is available to the defender's estimator at time k where $\mathcal{M} = \{A, B, C, Q, R, p_d\}$. This information is leveraged to obtain an estimate $\hat{x}_{k|k}$ and generate an input u_k. As in the prior chapter, we consider LQG cost optimization:

$$J = \lim_{N\to\infty} \mathbb{E}\left[\frac{1}{2N+1}\sum_{k=-N}^{N}\left(x_k^T W x_k + \eta_k u_k^T U u_k\right)\right] \tag{3.3}$$

where U and W matrices are positive definite and the optimization is performed over all inputs u_k that are measurable with respect to the information set $\mathcal{I}_k$. Note that the separation principle holds [9] and the optimal estimator and controller can be designed separately. A Kalman filter is used to obtain minimum mean squared error estimates $\hat{x}_{k|k} = \mathbb{E}[x_k|\mathcal{I}_k]$. The innovation or residual

$z_k = y_k - CA\hat{x}_{k-1|k-1} - \eta_{k-1}CBu_{k-1}$ is used to recursively update the state estimate as follows:

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + Kz_k,\ \hat{x}_{k|k-1} = A\hat{x}_{k-1|k-1} + \eta_{k-1}Bu_{k-1}, \tag{3.4}$$

where K is the stationary Kalman filter gain due to (A, C, Q, R):

$$K = PC^T(CPC^T + R)^{-1}, \tag{3.5}$$

$$P = APA^T + Q - APC^T(CPC^T + R)^{-1}CPA^T, \tag{3.6}$$

and $\hat{x}_{0|-1}$ is the initial apriori Kalman state estimate.

The optimal control is in the following form $u_k^* = L_k\hat{x}_{k|k}$ where $L_k = -(B^TS_{k+1}B + U)^{-1}B^TS_{k+1}A$ and

$$S_k = A^TS_{k+1}A + W - (1-p_d)A^TS_{k+1}B(B^TS_{k+1}B + U)^{-1}B^TS_{k+1}A.$$

We note that L_k converges to $L_{(b)} = -(B^TS_{(b)}B + U)^{-1}B^TS_{(b)}A$ where $S_{(b)}$ satisfies the Riccati equation:

$$S_{(b)} = A^TS_{(b)}A + W - (1-p_d)A^TS_{(b)}B(B^TS_{(b)}B + U)^{-1}B^TS_{(b)}A.$$

We assume that p_d is sufficiently small so that (3.7) has a solution. The long term average LQG cost due to the packet drops is (c.f. [9]) given as follows:

Lemma 3.1. *The optimal cost J is*

$$J = J_{(b)} = tr(S_{(b)}Q) + tr[(A^TS_{(b)}A + W - S_{(b)})(P - KCP)].$$

Proof. From equation (27) in [9], we have the optimal finite horizon cost, J_N^*, found as follows:

$$J_N^* = q_{-N} + \sum_{k=-N}^{N} \text{tr}(S_{k+1}Q) + \sum_{k=-N}^{N} \text{tr}(A^TS_{k+1}A + W - S_k)P_{k|k},$$

where q_{-N} is a bounded constant (specified in [9]) and

$$P_{k|k} = P_k - P_kC^T(CP_kC^T + R)^{-1}CP_k \tag{3.7}$$

Here, P_k denotes the apriori error covariance. As $N \to \infty$, $P_{k|k} \to P - KCP$ and $S_k \to S_{(b)}$. Thus, $\frac{1}{2N+1}J_N^* \to \text{tr}(S_{(b)}Q) + \text{tr}[(A^TS_{(b)}A + W - S_{(b)})(P - KCP)]$. □

It is worthwhile to note that J can be computed in closed form when packet drops occur only in the control channel. This is not possible in the general setting of [9] with sensor and control packet drops. We also note that the dependence of J on p_d is due to $S_{(b)}$. In the sequel, we assume that the system has been running for a long time (i.e. from $k = -\infty$) so that the Kalman and state feedback gains have converged to K and $L_{(b)}$, respectively.

3.1.3 Packet Drops as a Watermark

We now analyze the role of packet drops as a potential physical watermark. In a scenario where the control packets are dropped by following an IID Bernoulli sequence η_k as in (3.1), the resulting dynamics have strong dependence on the realization of the drop sequence. This dependence offers an advantage to be used for attack detection in the same spirit as the Gaussian physical watermark. The packet drop sequence, if known to the defender and kept secret from an attacker, acts as a new type of secret nonce that can be used in active detection.

We next consider packet drop injections in the context of replay attack detection. For ease of presentation, the replay attack is repeated below as follows:

1. The attacker records a sequence of sensor measurements from time $-T$ to time -1, where T is a large enough number to ensure that the attacker can replay the sequence later for an extended period of time.

2. Starting at time 0 to time $T-1$, the attacker modifies the sensor signals to y_k^a, which is the same as the measurements recorded by the attacker at time $k-T$. In other words,

$$y_k^a = y_{k-T},\ 0 \leq k \leq T-1.$$

3. Starting at time 0, the attacker injects an external control input $B^a u_k^a$, where $u_k^a \in \mathbb{R}^{p_a}$ is the control input and $B^a \in \mathbb{R}^{n \times p_a}$ denotes its direction.

The dynamics of the system

$$\hat{x}_{k|k-1}^a \triangleq \hat{x}_{k-T|k-T-1},\ z_k^a = z_{k-T},\ \eta_k^a = \eta_{k-T},\ 0 \leq k \leq T-1, \tag{3.8}$$

are defined above. During the replay ($0 \leq k \leq T-1$), the system dynamics changes to

$$x_{k+1} = Ax_k + \eta_k Bu_k + B^a u_k^a + w_k,\ y_k = Cx_k + v_k, \tag{3.9}$$

$$\hat{x}_{k+1|k} = A\hat{x}_{k|k} + \eta_k Bu_k,\ \hat{x}_{k|k} = \hat{x}_{k|k-1} + K\left(y_k^a - C\hat{x}_{k|k-1}\right), \tag{3.10}$$

$$u_k = L_{(b)}\hat{x}_{k|k},\ z_k = y_k^a - CA\hat{x}_{k-1|k-1} - \eta_{k-1}CBu_{k-1}. \tag{3.11}$$

Recall a replay attack may or may not be effective depending on the defender's control strategy. From Theorem 2.1, it is reported that replay attacks are asymptotically stealthy ($\lim_{k\to\infty}\beta_k - \alpha = 0$) in an LQG setting without drops provided that the matrix $\mathcal{A} \triangleq (A+BL_{(b)})(I-KC)$ is Schur stable. On the other hand, if $\mathcal{A}$ has a spectral radius greater than 1, then replay attacks are asymptotically detectable with an exponentially growing detection statistic. We consider the use of packet drop injections when $(A + BL_{(b)})(I - KC)$ is stable.

In particular, we consider residue detector performance under replay attack. Consider the residue z_k and the delayed version z_k^a during a replay attack with packet drops where $k < T$. We start by noting that

$$\begin{aligned} z_k = z_k^a &- C\mathcal{A}_k(\eta_{0:k-1})\hat{x}_{0|-1} + C\mathcal{A}_k(\eta_{0:k-1}^a)\hat{x}_{0|-1}^a \\ &- C\sum_{i=1}^{k}\left(\mathcal{A}_{k-i}(\eta_{i:k-1})(A+\eta_{i-1}BL_{(b)}) - \mathcal{A}_{k-i}(\eta_{i:k-1}^a)(A+\eta_{i-1}^aBL_{(b)})\right)Ky_{i-1}^a, \end{aligned} \tag{3.12}$$

For any $\ell_1 \leq \ell_2$, we define

$$\mathcal{A}_{\ell_2-\ell_1}(\eta_{\ell_1+1}^{\ell_2}) = \Pi_{j=\ell_1+1}^{\ell_2}(A+\eta_j BL_{(b)})(I-KC), \tag{3.13}$$

where $\mathcal{A}_0 = I$ and $\eta_{\ell_1+1:\ell_2}$ denotes the sequence $(\eta_{\ell_1+1}\dots\eta_{\ell_2})$. For $k \leq T$, we see in (3.12), $\{\eta_k\}$ and $\{\eta_k^a\}$ are two binary drop sequences independent from each other and IID across k. We note that even when $\mathcal{A}_k(\eta_{0:k-1})$ vanishes, the additive term $\nu_k \triangleq C\sum_{i=1}^{k}(\mathcal{A}_{k-i}(\eta_{i:k-1})(A+\eta_{i-1}BL_{(b)}) - \mathcal{A}_{k-i}(\eta_{i:k-1}^a)(A+\eta_{i-1}^aBL_{(b)}))Ky_{i-1}^a$ renders the residue z_k different than the residue z_k^a. For example, we can show that if $||(A+BL_{(b)})||^{1-p_d}||A||^{p_d}||(I-KC)|| < 1$ where $||.||$ denotes the matrix norm, then $\mathcal{A}_k(\eta_0^{k-1})$ vanishes in probability. However, the additive term ν_k does not

vanish and creates a difference in the distributions of z_k and z_k^a. This additive term has a similar effect to that of the additive watermark in the previous chapter and can be leveraged to detect replay attacks. As an example, one can characterize explicit or approximate distributions of the additive term and analyze detection performance. Also note that when $p_d = 0$ or $p_d = 1$ or (possibly) the packet drop sequence is periodic, the effect of the additive term is lost since $\mathcal{A}_{\ell_2-\ell_1}(\eta_{\ell_1+1:\ell_2})$ is equivalent to $\mathcal{A}_{\ell_2-\ell_1}(\eta^a_{\ell_1+1:\ell_2})$. In these cases, the asymptotic stealthiness condition described in Theorem 2.1 could be adapted to the current setting. In the next subsection, we provide real life examples and extensive numerical results to determine the effects of packet drop injection watermarking on both detection performance and overall cost.

3.1.4 Numerical Examples

In this section we evaluate the performance of physical watermarking via packet drop injections on two systems. We first consider replay attacks in the quadruple tank process [39]. Then, we examine a microgrid example [30].

Quadruple Tank Process

In the quadruple tank process, the desired system goal is to control the water level of two tanks by leveraging two input pumps. Two sensors are used to measure the water heights of two tanks. The chosen sample period is 1 second. We use an LQG controller with weighting matrices determined using suggestions made in [40]. When examining the quadruple tank process, the optimal state feedback matrix $L_{(b)}$ is dependent on the probability of drop p_d.

A passive detector that recognizes the difference between normal and malicious operation must be selected. As in the previous chapter, we assume the defender constructs algorithms which leverage his/her information $\mathcal{I}_k$ to make a decision, whether the system operates normally $\mathcal{H}_0$ or under attack $\mathcal{H}_1$. In a threshold based detector, this can be formulated as

$$g_k(\mathcal{I}_k) \underset{\mathcal{H}_0}{\overset{\mathcal{H}_1}{\gtrless}} \tau_k. \tag{3.14}$$

Noting the difference in the distribution of the residue under attack and normal operation, we select a χ^2 detector.

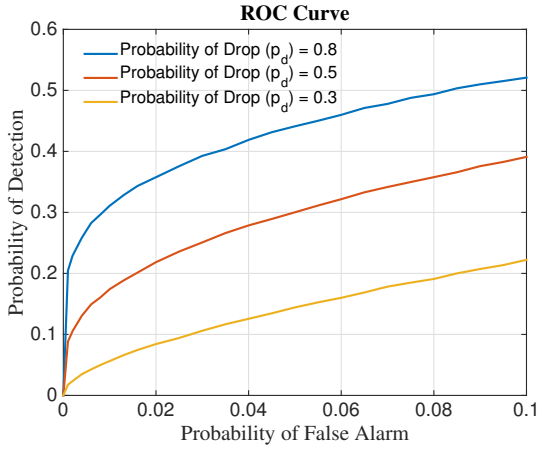
$$g_k(\mathcal{I}_k) = \sum_{t=k-WS+1}^{k} z_t^T (CPC^T + R)^{-1} z_t \tag{3.15}$$

Under normal operation $z_t^T (CPC^T + R)^{-1} z_t$ should follow a χ^2 distribution with m degrees of freedom. The χ^2 detector attempts to exploit this fact by testing to see if the innovations follow the correct distribution. It is easy to see that large residues, indicating a discrepancy between measured and expected behavior, create alarms, while smaller residues, which indicate good agreement between measured and expected behavior, are indicative of normal operation.

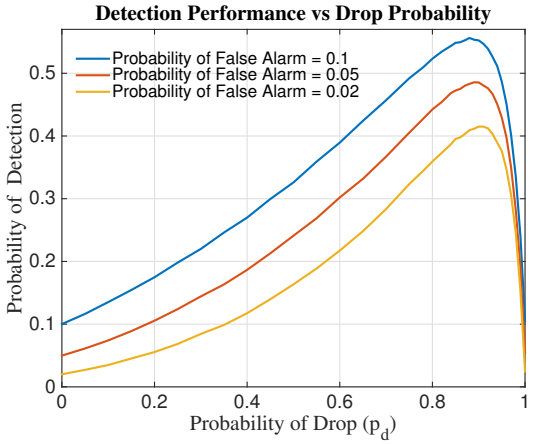
Note unlike our previous detectors, here we are allowing the possibility for a larger window size. A larger window allows the defender to use more information, which can aid the quality of detection. However, this can come at the cost of time to detection as typically a larger delay is seen before an attack can significantly impact a detection statistic. In this system, we take the window size WS to be 10.

In Fig. 3.2 we examine security and performance trade-offs through relationships between the probability of false alarm, the probability of detection, and the packet drop rate. Results were averaged over 1500 trials where each trial consists of a run with 1000 time steps. In Fig. 3.2(a), we plot several ROC curves examining the probability of detection as a function of the probability of false alarm for different packet drop rates. In Fig. 3.2(b), we plot the probability of detection as a function of the drop rate for different false alarm probabilities ranging from 0.02 to 0.1. Note that detection performance peaks before the drop rate equals one. This can be understood in the extreme case where $p_d = 1$. Here, the system is operating in an open loop without control. Thus, when using a stable estimator, a replay attack will always be asymptotically stealthy.

In Fig. 3.3, we further characterize the tradeoff between security and control performance by mapping the probability of drop to the increased LQG cost (as a percentage of the optimal LQG cost when $p_d = 0$). In Fig. 3.3(a), we observe the relationship between control performance and drop probability over the domain of p_d. In Fig. 3.3(b), we examine this relationship over a smaller



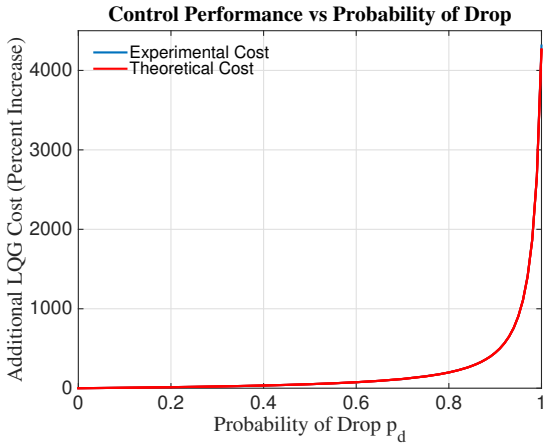
(a) Detection vs False Alarm



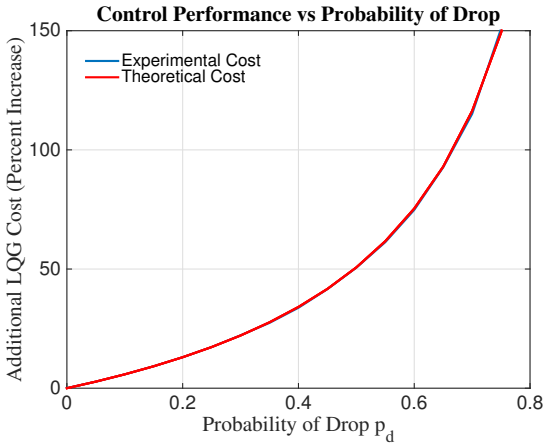
(b) Detection vs Drop Rate

Figure 3.2: Probability of Detection as a Function of Probability of Drop and Probability of False Alarm, Quadruple Tank

domain where the cost increase is restricted to be less than 150% of the optimal cost. Both the empirical cost, obtained by averaging results over 4,500 trials, and the theoretical cost are shown. We observe that they closely agree.



(a) LQG Cost vs Drop Rate: Full

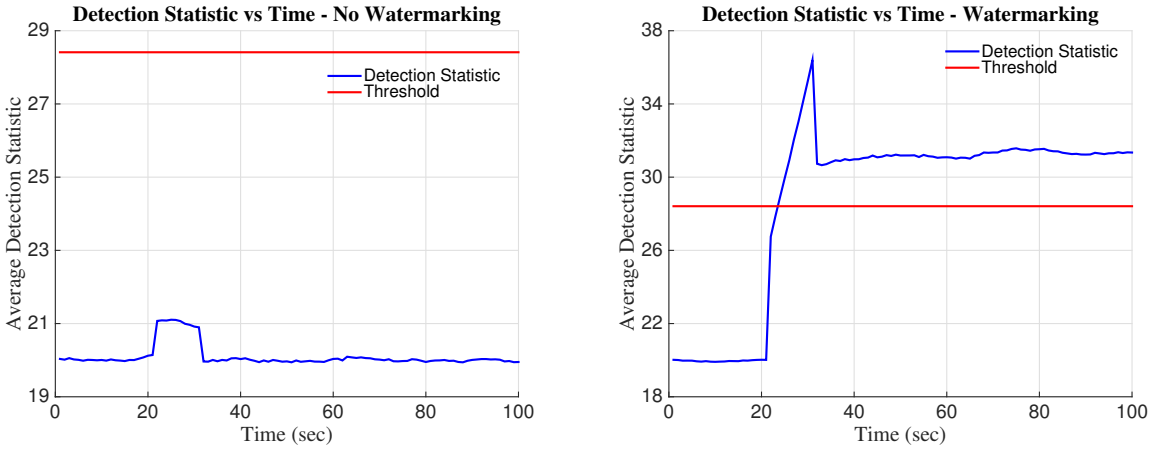


(b) LQG Cost vs Drop Rate: Partial

Figure 3.3: Percent Increase in LQG Cost as a Function of Drop Probability, Quadruple Tank

In Fig. 3.4, we plot our χ^2 detection statistic (with window size 10) averaged over 10,000 trials during a replay attack as a function of time for a system without packet drop injections (Fig. 3.4(a)) and a system with packet drop injections (Fig. 3.4(b)). Replay attacks commence at time 20. The

probability of false alarm in Fig. 3.4 is fixed to be 0.1 and $p_d = 0.7$. The noticeable temporary bumps in detection performance seen in both Fig. 3.4(a) and Fig. 3.4(b) are likely due to initial state mismatches between the true and replayed systems. An intelligent attacker can choose to delay the start of a replay attack until the true and replayed states closely match.



(a) No Watermarking

(b) Watermarking, $p_d = 0.7$

Figure 3.4: χ^2 Detection Statistic vs Time, Packet Drops in Quadruple Tank

Microgrid

We now investigate a microgrid example borrowed from [30], using an alternative watermarking design. Here, there are 5 loads and frequency control by a mechanical speed governor is used to address small imbalances (roughly 1 percent) between load and demand. The frequency should be kept close to constant near 60 Hz. If the demand in a system far exceeds the generation, resulting in a measured drop in frequency, loads are shed to account for the imbalance. We use the linear generator model found in [41, p. 386, Fig. 11.8], see also [30]. ΔP_c, a control input which moves a steam valve in the generator, is used for watermarking. Additionally, we use $\Delta\omega$ to denote a change in angular frequency.

In the attack model, the attacker has the ability to manipulate the system's frequency sensors. The goal is to make the operator believe the frequency in the system is dropping. The defender in

response sheds loads one at a time to address perceived imbalances. The attacker, once a third load is shed, relinquishes control on the frequency sensor and this way the attacker forces the operator to supply power to only two loads.

As a response, we assume the defender inserts a watermark at ΔP_c. As opposed to the packet drop watermark considered in this section, we evaluate a similar zero-mean Bernoulli pulse watermark. In particular, we have

$$\Delta P_c(k) = \eta_k M(-1)^k. \tag{3.16}$$

where M is the magnitude of the pulse and η_k is an IID Bernoulli random variable where P($\eta_k = 0$) $= p_d$. Observe that a χ^2 detector is ineffective against the proposed attack because it will send an alarm in both the case that an attacker modifies a frequency sensor as well as the case that a real drop in frequency has occurred. As a result, we consider the correlation based detectors used in [30]. Here, a virtual model of the system with input ΔP_c is simulated by the defender. The response $\Delta\hat{\omega}_k$ is multiplied by the true frequency $\Delta\omega$ to obtain a correlation detector statistic g_k. Under normal operation,

$$\mathbb{E}[\Delta\hat{\omega}_k \Delta\omega_k] = \mathbb{E}[g_k] = \sigma'^2 > 0. \tag{3.17}$$

Under a replay attack $\mathbb{E}[\Delta\hat{\omega}_k \Delta\omega_k] = 0$. Unlike the χ^2 detector, a higher detection statistic indicates normal operation.

We simulate the microgrid over 70 seconds. Control inputs are modified every 0.1 seconds. The amplitude M controls the variance of the watermark, $\mathbb{E}[\Delta P_c^2(k)]$. A correlation detector with window of length 10 seconds is used. We consider two scenarios. We first assume the sensor is not under attack, but the frequency in the system is dropping. The average frequency profile considered is given by Fig. 3.5. Secondly, an attacker replays the same profile (with noise independent of the watermark) from time 10 sec to 57.5 sec to force the defender to incorrectly shed loads.

In Fig. 3.6, we plot several ROC curves averaged over 1500 trials. The probability of detection is computed over the region where g_k has reached a steady state (20 to 57.5 seconds). Three

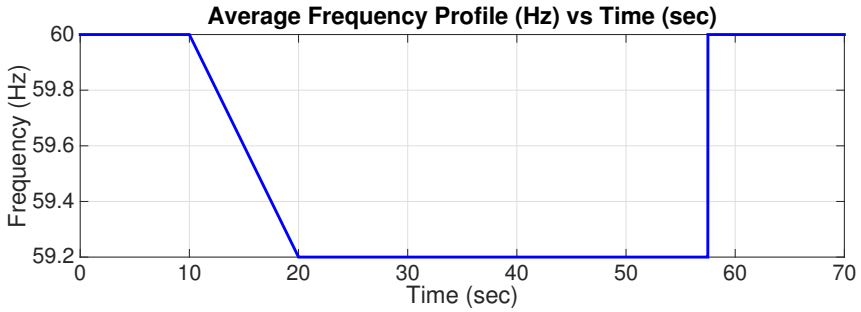


Figure 3.5: Average Frequency Profile during Fault and Attack

different watermark variances $\mathbb{E}[\Delta P_c^2(k)]$ and p_d's are evaluated where we observe that increasing $\mathbb{E}[\Delta P_c^2(k)]$ improves detection performance.

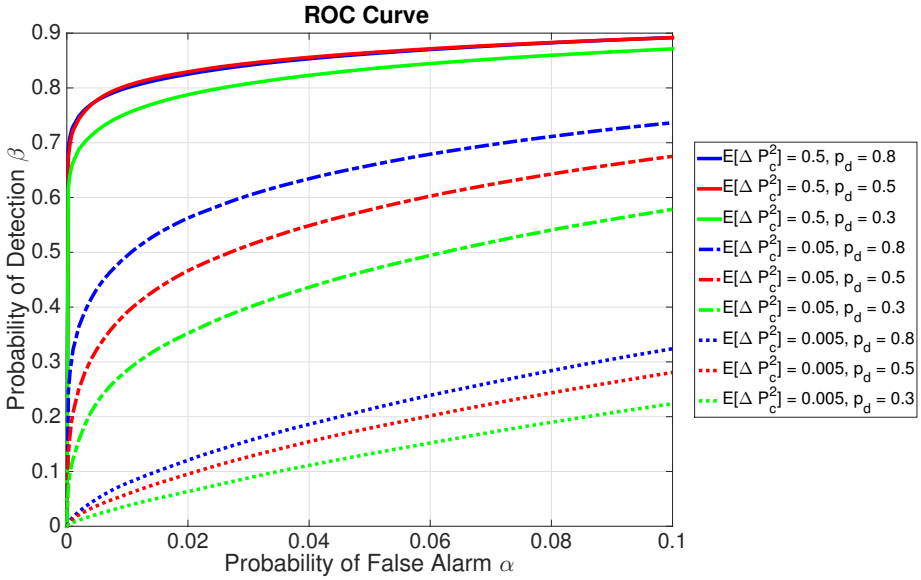


Figure 3.6: Probability of Detection vs. Probability of False Alarm: Microgrid Bernoulli Watermark

In Fig. 3.7, we observe the detection statistics used by the correlation detector under system fault and replay attack scenarios as a function of time, averaged over 1500 trials. In this setting, the variance of the watermark is set to 0.5. Since the replayed profile is independent of the pulse watermark under a replay attack the correlation drops to 0. Detection delays occur due to the chosen 10 second detector window.

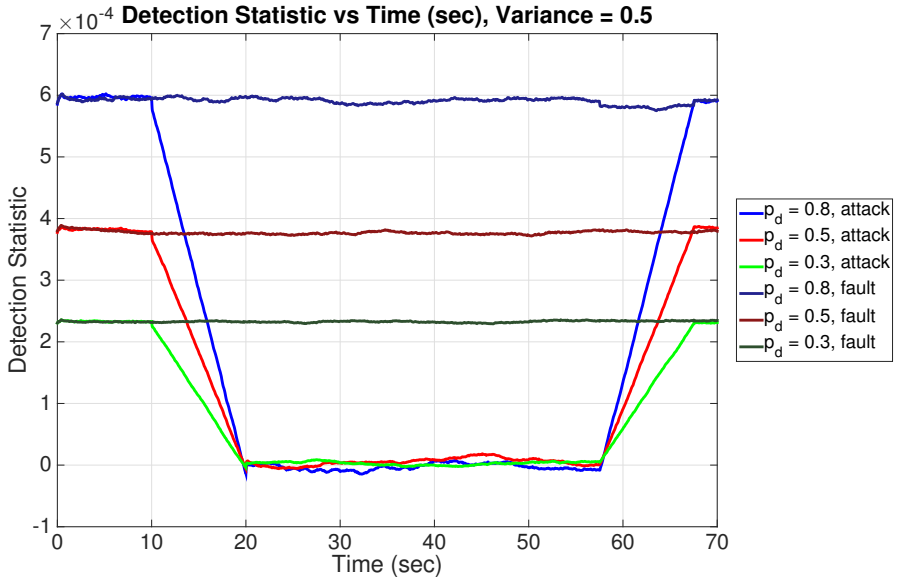


Figure 3.7: Detection Statistic During Fault and Attack: Microgrid Bernoulli Watermark

As an additional measure of the watermark's affect on system performance, we consider the mean absolute deviation of the measured frequency from the average frequency profile with watermarking. Note that in the absence of watermarking, the mean deviation is 0.0252 Hz for the simulation setting. Introducing larger, more random, watermarks to improve security of course increases the frequency deviation in the system.

Table 3.1: Mean Abs. Deviation from Avg. Freq. Profile (Hz)

$\mathbb{E}[\Delta P_c^2]$	$p_d = 0.3$	$p_d = 0.5$	$p_d = 0.8$
0.005	0.0254	0.0256	0.0258
0.05	0.0275	0.0288	0.0307
0.5	0.0429	0.0513	0.0604

3.2 A Joint Gaussian and Packet Drop Watermark

We observed that naturally occurring phenomena, specifically packet drops at the control input can act as a watermark and enable active detection. In this section, we consider the joint design of Gaussian and packet drop watermarks. This work simultaneously considers two scenarios. First, the defender is able to design Gaussian watermarks while also accounting for realistic network uncertainties. Secondly, the defender can introduce a hybrid watermarking scheme that combines both packet drops and Gaussian watermarks for the goal of maximizing detection performance. This section will investigate the design of 1) an input with IID Gaussian watermark, multiplied by a Markovian drop process at the control input 2) an input with a stationary Gaussian watermark, multiplied by an IID drop process at the control input.

3.2.1 System Description

We consider the same system dynamics (3.1), (3.2) as in the previous section, with the same assumption on (A, B, C, Q, R). Moreover, we consider the same LQG cost (3.3) which a defender aims to minimize. For clarity, in this section we differentiate between the control input the defender computes, u_k, and the control input the plant receives, which we define as $u_{k,c}$. We have

$$u_{k,c} \triangleq \eta_k u_k. \tag{3.18}$$

A Kalman filter can be used to perform optimal state estimation (in the minimum mean squared error sense) (3.4)

Once more, if drops occur naturally in the system, we assume an acknowledgement is delivered when a control input is successfully delivered. We consider both IID and Markovian Bernoulli drop sequences. In the IID case, $\Pr(\eta_k = 1) = 1 - p_d$. Assume the system has been running for a long time and p_d is chosen so the system can have finite cost J. Then, given an information set

$\mathcal{F}_k \triangleq \{y_{-\infty:k}, \eta_{-\infty:k-1}, u_{-\infty:k-1}\}$, the optimal control strategy has control input $u_k = u_k^b$ where

$$u_k^b = L_{(b)}\hat{x}_{k|k},\ L_{(b)} = -(B^T S_{(b)} B + U)^{-1} B^T S_{(b)} A,$$
$$S_{(b)} = A^T S_{(b)} A + W - (1-p_d) A^T S_{(b)} B (B^T S_{(b)} B + U)^{-1} B^T S_{(b)} A.$$

In addition, as shown in the previous section, $J = J_{(b)}$ for this strategy where $J_{(b)}$ is

$$J_{(b)} = \text{tr}\left(S_{(b)}Q + (A^T S_{(b)} A + W - S_{(b)})(P - KCP)\right). \tag{3.19}$$

In the Markovian case, considered in [42],we assume packet drops follow a Markovian process.

$$\begin{bmatrix} \Pr(\eta_{k+1}=0|\eta_k=0) & \Pr(\eta_{k+1}=1|\eta_k=0) \\ \Pr(\eta_{k+1}=0|\eta_k=1) & \Pr(\eta_{k+1}=1|\eta_k=1) \end{bmatrix} = \begin{bmatrix} 1-\alpha & \alpha \\ \beta & 1-\beta \end{bmatrix} \tag{3.20}$$

Here, we assume $0 < \alpha \leq 1$, $0 < \beta \leq 1$ so that η_k is irreducible. Moreover, we assume η_k is stationary, which can be obtained by letting its initial distribution be $\Pr(\eta_{-\infty} = 0) = \frac{\beta}{\alpha+\beta}$. Finally, we assume that α and β are selected (or given) so that the system can have finite cost J. The optimal control strategy at time k given $\mathcal{F}_k$ generates input u_k^m

$$u_k^m = L_{(m)}\hat{x}_{k|k},\ L_{(m)} = -(B^T R_{(m)} B + U)^{-1} B^T R_{(m)} A,$$
$$R_{(m)} = A^T(\beta S_{(m)} + (1-\beta)R_{(m)})A + W - (1-\beta)A^T R_{(m)} B (B^T R_{(m)} B + U)^{-1} B^T R_{(m)} A,$$
$$S_{(m)} = A^T((1-\alpha)S_{(m)} + \alpha R_{(m)})A + W - \alpha A^T R_{(m)} B (B^T R_{(m)} B + U)^{-1} B^T R_{(m)} A,$$

where $L_{(m)}, R_{(m)}, S_{(m)}$ are parameters which converged to their steady state values. The resulting cost of control is

$$J_{(m)} = \frac{\text{tr}(\beta S_{(m)} Q + \alpha R_{(m)} Q)}{\alpha+\beta} + \frac{\text{tr}((A^T((1-\alpha)S_{(m)} + \alpha R_{(m)})A + W - S_{(m)})(P - KCP))}{\alpha+\beta}.$$

Note, we preserve the notation defined in [42] where α and β are use to define the Markovian drop process. This should not be confused with notation defining the probability of false alarm and the probability of detection.

Remark 3.1. *The prior strategies are optimal when the defender only has knowledge of the observed drop sequence $\eta_{-\infty:k-1}$. However, if the drop sequence is intentionally introduced to improve watermarking/detection performance by using a pseudo random number generator (PRNG), the defender knows future values of η_k. The design of a controller that uses this information is left for future work.*

A Joint Bernoulli Gaussian Physical Watermark

We now aim to intelligently combine the Gaussian watermarks with a Bernoulli drop process at the input. Such a design accomplishes two goals: 1) to expand the analysis of physical watermarking to a more realistic network setting with packet drops and 2) to potentially improve performance by considering a more general joint Bernoulli-Gaussian watermark. The joint design allows us to mix environmental watermarks that may occur naturally within the confines of a system and intentional physical watermarks in order to attain better detection performance.

We consider two main joint designs.

Watermark 1: IID Gaussian Input + Markovian Drops

$$u_{k,c} = \eta_k(u_k^m + \Delta u_k). \tag{3.21}$$

$\{\eta_k\}$ is a Markovian Bernoulli process and $\Delta u_k \sim \mathcal{N}(0, \mathcal{J})$ is an IID Gaussian watermark [15]. We assume Δu_k is independent of other stochastic processes in the system.

Watermark 2: Stationary Gaussian Input + IID Drops

$$u_{k,c} = \eta_k(u_k^b + \Delta u_k). \tag{3.22}$$

In this case, $\{\eta_k\}$ is an IID Bernoulli process. The Gaussian input Δu_k is assumed to be a stationary process generated by a hidden Markov model (HMM) as considered in section 2.1

$$\xi_{k+1} = A_\omega \xi_k + \psi_k, \quad \Delta u_k = C_h \xi_k. \tag{3.23}$$

ξ_k is the hidden state of the HMM, A_ω has spectral radius $\rho(A_\omega) \leq \bar{\rho} \leq 1$, and $\psi_k \sim \mathcal{N}(0, \Psi)$ is IID Gaussian noise. For stationarity, $\text{Cov}(\xi_0) = A_\omega \text{Cov}(\xi_0) A_\omega^T + \Psi$. Δu_k is independent of other stochastic processes in the system.

Recall that $\bar{\rho}$, the maximum allowable spectral radius, is a design parameter for the defender. We observe a larger $\bar{\rho}$ improves expected detection performance. However, a larger $\bar{\rho}$ means a larger correlation between watermarks and this could facilitate the prediction of future watermarks if the attacker guesses an initial Gaussian input Δu_k. A system diagram is found in Fig. 3.8

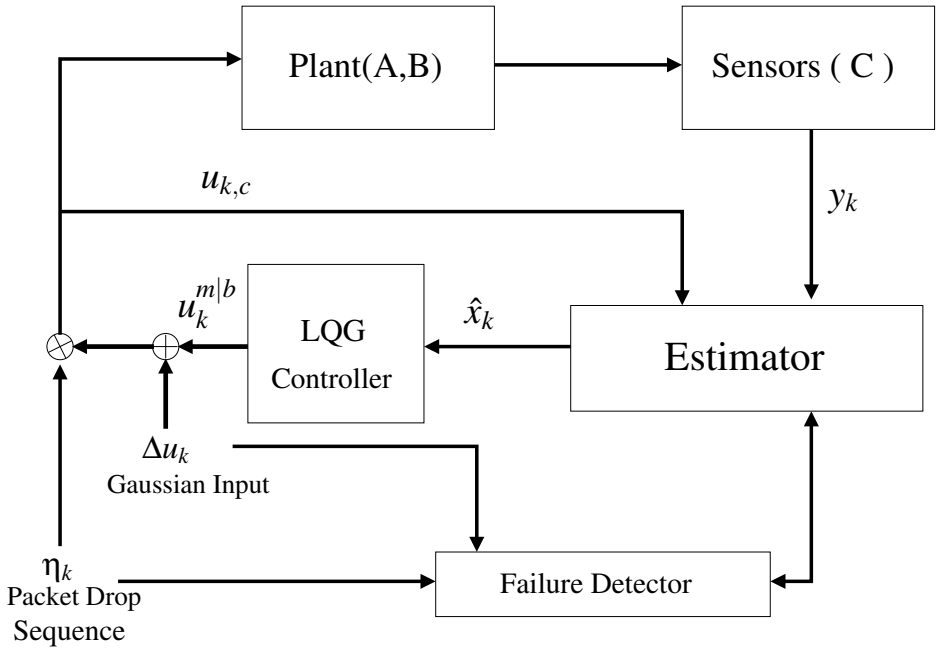


Figure 3.8: System Model with a Joint Packet Drop and Gaussian Watermark

3.2.2 Attack Model

In this section we describe a model of our adversary in terms of knowledge, capabilities, and potential strategies. We will show the attack is a generalization of a replay attack. In particular, it can reflect a replay adversary or a model aware attacker who constructs simulations.

Attacker Capabilities

Without loss of generality, we assume an attack begins at time $k = 0$. We make the following assumptions.

1. The attacker can modify all measurements y_k, $k \geq 0$. The falsified outputs at time k are denoted by y_k^v.

2. The attacker inserts an input $B^a u_k^a$ into the system.

3. The attacker is unable to read the true control inputs $u_{k,c}$. As a result, he is unaware of the drop sequence $\{\eta_k\}$ and the Gaussian watermark $\{\Delta u_k\}$.

The system under attack is given by

$$x_{k+1} = Ax_k + Bu_{k,c} + B^a u_k^a + w_k, \tag{3.24}$$

$$\hat{x}_{k+1|k+1} = (I - KC)(A\hat{x}_{k|k} + Bu_{k,c}) + Ky_{k+1}^v. \tag{3.25}$$

Attack Strategy

The attacker generates y_k^v through a virtual system:

$$x_{k+1}^v = Ax_k^v + \eta_k^v B(L_{m|b}\hat{x}_{k|k}^v + \Delta u_k^v) + w_k^v, \quad y_k^v = Cx_k^v + v_k^v. \tag{3.26}$$

$$\hat{x}_{k+1|k+1}^v = (I - KC)(A + \eta_k^v BL_{m|b})\hat{x}_{k|k}^v + Ky_{k+1}^v + \eta_k^v(I - KC)B\Delta u_k^v, \tag{3.27}$$

In the case of Watermark 1, $L_{m|b} = L_{(m)}$, η_k^v follows a Markovian process (3.20) with parameters α and β and $\Delta u_k^v \sim \mathcal{N}(0, \mathcal{J})$ is an IID Gaussian process. In the case of Watermark 2, $L_{m|b} = L_{(b)}$, η_k^v is an IID Bernoulli process with drop probability p_d and Δu_k^v is a stationary Gaussian process which satisfies (3.23). Additionally, $v_k^v \sim \mathcal{N}(0, R)$ and $w_k^v \sim \mathcal{N}(0, Q)$ are IID processes. Finally, we assume the stochastic processes $\{\eta_k^v, \Delta u_k^v, w_k^v, v_k^v\}$ are independent of the real system's stochastic parameters $\{\eta_k, \Delta u_k, w_k, v_k\}$.

The previous attack strategy can be generated (approximately) by the replay attack where the attacker records a long sequence of outputs $y_{=1:-T}$ and, starting at time 0, replaces y_k with $y_k^v = y_{k-T}$ for $0 \leq k \leq T-1$. Attackers who do not have precise knowledge of the model may engage in replay attacks, which only require access to the outputs. Alternatively, this attack strategy can be constructed by an adversary who is familiar with the model, for instance a malicious insider. In this case, the attacker simulates a virtual copy of the system dynamics to fool a bad data detector. We refer to such an attack as a simulation attack. The placement of the simulation attack on the attack space is given in Fig. 3.9.

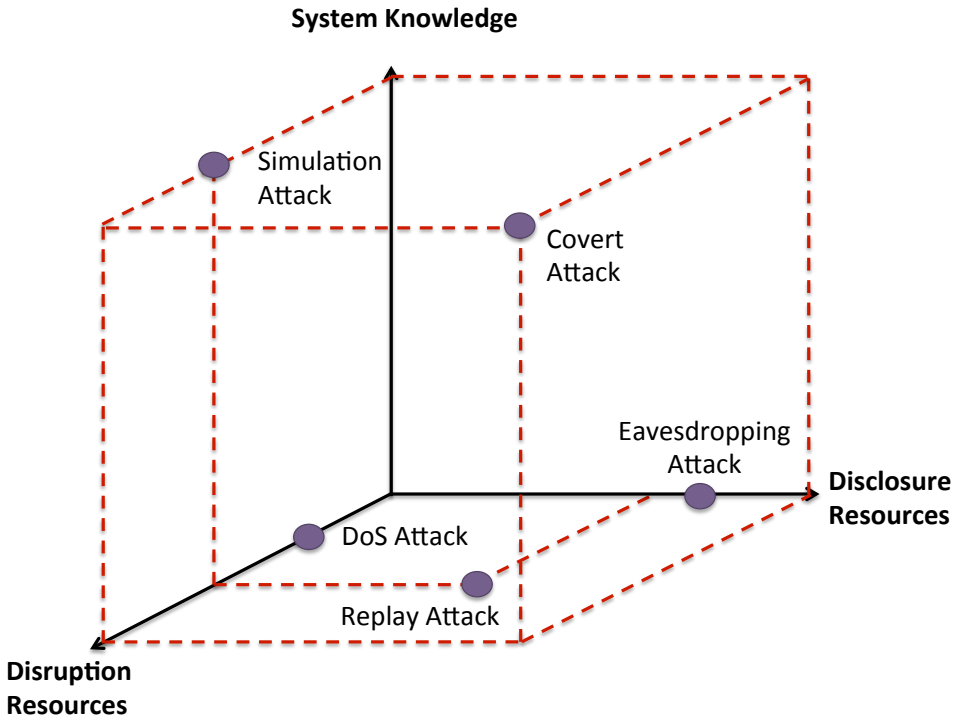


Figure 3.9: Cyber-Physical Attack Space with Simulation Attack

A model aware attacker could also potentially pursue an additive attack, for instance a false data injection attack [43] or a zero dynamics attack [44, 19]. In these attacks, the adversary injects

an additive bias into the system which preserves the watermark and allows the attacker to remain stealthy. However, there are scenarios where additive attacks on sensor measurements are not feasible. As an example, suppose the defender uses public key cryptography, where a public key is used to encrypt the measurements while a private key is used to decrypt the associated cipher text. An attacker could send his own virtual measurements encrypted with the public key. If the encryption is not malleable, such an attack could not leverage information in the true measurement as that would require access to the defender's private key to learn y_k. In this case, additive attacks constructed by replacing a true output packet with a virtual packet would be infeasible. By assumption, an additive networked-based attack on the defender's control input is also impossible because the adversary is unable to read the defender's input.

We argue that alternative attack strategies which manipulate all sensors y_k in a setting with public key cryptography also fail due to the fact that the resulting attack sequence $\{y_k^v\}$ is independent of the watermarks $\{\Delta u_k, \eta_k\}$. Specifically, an attacker who is unable to read the inputs or outputs will have no information about the watermarks. As a result, the outputs he can construct will fail to fool the correlation detector, which we propose in the next subsection.

3.2.3 A Correlation Detector

In the previous sections we have briefly introduced several possible passive detectors including the Neyman Pearson detector as well as the χ^2 detector. In this section, we will now closely examine the design of a correlation detector. We will see that when combined with active detection, the correlation detector allows us to detect classes of attacks, and potentially distinguish certain faults from attacks.

In the correlation detector, (considered in [27]), the defender computes a virtual output y'_k, which explicitly characterizes the effect of watermarks on y_k.

$$x'_{k+1} = Ax'_k + \eta_k B(L_{m|b}\hat{x}'_{k|k} + \Delta u_k), \quad y'_k = Cx'_k, \tag{3.28}$$

$$\hat{x}'_{k+1|k+1} = (I - KC)(A + \eta_k BL_{m|b})\hat{x}'_{k|k} + Ky'_{k+1} + \eta_k(I - KC)B\Delta u_k \tag{3.29}$$

where with some abuse of notation $x'_{-\infty} = 0, \hat{x}'_{-\infty|-\infty} = 0$. We can simplify (3.28) and (3.29) to obtain

$$x'_{k+1} = (A + \eta_k B L_{m|b}) x'_k + \eta_k B \Delta u_k, \quad y'_k = C x'_k. \tag{3.30}$$

This virtual process created by the defender is driven entirely by the sequence of Bernoulli-Gaussian watermarks $\{\Delta u_k, \eta_k\}$. Thus, if we were to multiply the true outputs y_k with the defender's virtual outputs y'_k we would expect a positive correlation. However, if an attacker introduces measurements y^v_k, which are driven by an independent sequence of watermarks, the expected correlation drops to 0. This motivates consideration of the detection statistic $y_k^T y'_k$, where a large statistic is indicative of normal behavior while a small statistic indicates malicious behavior. Observe due to the random real time selection of watermarks, $\|y'_k\|_2$ may be close to 0, impacting detector performance since the correlation will likely also approach 0 even under normal operation. As a result, we propose an event triggered detector:

$$\begin{aligned} &\text{If } \|y'_k\|_2^2 \geq \mu \qquad \textit{Perform Detection} \\ &\kappa = \kappa + 1, \;\; t_\kappa = k \\ &\sum_{j=\kappa - WS+1}^{\kappa} g_j \underset{\mathcal{H}_1}{\overset{\mathcal{H}_0}{\gtrless}} \tau, \quad g_\kappa = y_{t_\kappa}^T y'_{t_\kappa}. \end{aligned} \tag{3.31}$$

The null hypothesis $\mathcal{H}_0$ is that the system is operating without malicious behavior while the alternative hypothesis $\mathcal{H}_1$ is that the system is under attack. WS is the size of the detector's window. A detection event is triggered if $\|y'_k\|_2^2$ is greater than some user defined threshold μ, preventing false alarms from being raised when y'_k is small, while sacrificing time to detection. This tradeoff can be addressed by tuning μ. Note that κ corresponds to the time index of the event triggered correlation detector and increases at instants when a new detection statistic is computed. Identifying attacks on an individual sensor i can be done by focusing on the correlation between individual measurements. An appropriate statistic g^i_κ would be $y^i_{t_\kappa} {y^i_{t_\kappa}}'$ where $y^i_{t_\kappa}$ is the ith entry of y_{t_κ}.

Remark 3.2. *A detector with an adaptive threshold could address issues of small y'_k. However,*

such a detector is more prone to misses, mistaking an attack for noise. Incorporation and analysis of such a detector is left for future work.

Remark 3.3. *An adversary that can not read $\{u_k\}, \{y_k\}$ can not take advantage of instances when detection does not occur, because such instances are entirely dependent on the realization of previous watermarks. An attacker who is forced to act independently of the real time watermarking sequence cannot determine if a detection has been triggered.*

We now verify that the expected correlation is 0, if the outputs y_k^v are generated independently of the watermarks.

Theorem 3.1. *If y_k^v and $\{\Delta u_k, \eta_k\}$ are independent, then*

$$\mathbb{E}\left[y_k^{vT} y_k' \middle| \|y_k'\|_2^2 \geq \mu \right] = 0.$$

Proof. Observe that y_k' can be written as a linear function of the Gaussian watermarks Δu_k so that

$$y_k' = \sum_{j=-\infty}^{k-1} G_j(\eta_{j:k-1}) \Delta u_j, \tag{3.32}$$

where G_j is some linear gain, determined by the sequence of Bernoulli drops $\eta_{j:k-1}$. Thus, we have

$$\begin{aligned} \mathbb{E}[y_k^{vT} y_k'] &= \mathbb{E}\left[y_k^{vT} \sum_{j=-\infty}^{k-1} G_j(\eta_{j:k-1}) \Delta u_j \middle| \|y_k'\|_2^2 \geq \mu \right] \\ &= \sum_{j=-\infty}^{k-1} \mathbb{E}\left[y_k^v\right]^T \mathbb{E}\left[G_j(\eta_{j:k-1}) \Delta u_j \middle| \|y_k'\|_2^2 \geq \mu \right] = 0. \end{aligned}$$

□

The proposed detector can often differentiate between faulty and malicious scenarios. During a fault, we expect to see the effect of the embedded watermarks in the output and it could be measured through correlation. Alternatively, residue based detectors such as the χ^2 detector ($g_\kappa = -z_{t_\kappa}^T (CPC^T + R)^{-1} z_{t_\kappa}$), which measures the difference between measured and expected behavior, will likely raise an alarm during faulty behavior and malicious behavior. Both detectors

can be used in tandem. A χ^2 detector can raise alarms in the case of faulty or malicious behavior, while a correlation detector can distinguish these events. In this section, we focus on the correlation detector.

3.2.4 Markovian - IID Gaussian Watermark

We consider the design of a watermark consisting of an IID Gaussian input and Markovian drops. This requires the evaluation of a detection and performance trade-off. We wish to maximize the correlation of y_k and y'_k to distinguish the system under attack from normal operation. However, we also need to ensure the system meets an adequate level of performance. We do this by considering the cost $\bar{J}$, starting at $k = 0$.

$$\bar{J} = \lim_{N\to\infty} \frac{1}{N}\mathbb{E}\left[\sum_{k=0}^{N-1} x_k^T W x_k + u_{k,c}^T U u_{k,c}\right] \tag{3.33}$$

As such, we design the parameters $\alpha, \beta, \mathcal{J}$ by solving the following optimization problem

$$\begin{aligned} \underset{\alpha,\beta,\mathcal{J}}{\text{maximize}} \quad & \lim_{k\to\infty} \mathbb{E}[y_k^T y'_k|\mathcal{H}_0] \\ \text{subject to} \quad & \bar{J} \leq \delta,\ 0 < \alpha, \beta \leq 1. \end{aligned} \tag{3.34}$$

To begin with, we use [42, Theorem 3] to analytically compute the cost $\bar{J}$ as follows.

Theorem 3.2. *Suppose α and β are chosen so that the system has finite cost $J_{(m)}$ in the absence of a Gaussian watermark. The LQG cost $\bar{J}$ of the control system (3.1), (3.2) with IID Gaussian and Markovian watermark (3.21) is:*

$$\bar{J} = J_{(m)}(\alpha, \beta) + \frac{\alpha}{\alpha + \beta} tr\left((B^T R_{(m)} B + U)\mathcal{J}\right). \tag{3.35}$$

Proof. Consider the cost to go in a finite horizon, $V_k(x_k) \triangleq \sum_{j=k}^{N} \mathbb{E}\left[x_j^T W x_j + u_{j,c}^T U u_{j,c}|\mathcal{F}_k\right]$, and let $u_{N,c} = 0$. Similar to, [42], it can be shown that

$$V_k(x_k) = \begin{cases} \mathbb{E}[x_k^T S_k x_k|\mathcal{F}_k] + c_k & (\eta_{k-1} = 0) \\ \mathbb{E}[x_k^T R_k x_k|\mathcal{F}_k] + d_k & (\eta_{k-1} = 1) \end{cases}, \tag{3.36}$$

where $c_N = d_N = 0,\ R_N, S_N = W,\ \tilde{P} = P - KCP, F = A + BL_{(m)}$ and

$$\begin{aligned}
R_k &= W + \beta A^T S_{k+1} A + (1-\beta) F^T R_{k+1} F + (1-\beta) L_{(m)}^T U L_{(m)}, \\
S_k &= W + (1-\alpha) A^T S_{k+1} A + \alpha F^T R_{k+1} F + \alpha L_{(m)}^T U L_{(m)}, \\
c_k &= -\alpha \text{tr}((F^T R_{k+1} F - A^T R_{k+1} A + L_{(m)}^T U L_{(m)})(\tilde{P})) + (1-\alpha)[\text{tr}(S_{k+1} Q) + c_{k+1}] \\
&+ \alpha[\text{tr}(R_{k+1} Q) + d_{k+1} + \text{tr}((B^T R_{k+1} B + U)\mathcal{J})], \qquad (3.37) \\
d_k &= -(1-\beta)\text{tr}((F^T R_{k+1} F - A^T R_{k+1} A + L_{(m)}^T U L_{(m)})\tilde{P}) + \beta[\text{tr}(S_{k+1} Q) + c_{k+1}] \\
&+ (1-\beta)[\text{tr}(R_{k+1} Q) + d_{k+1} + \text{tr}((B^T R_{k+1} B + U)\mathcal{J})]. \qquad (3.38)
\end{aligned}$$

Let $\bar{J}_N = \mathbb{E}\left[\sum_{k=0}^N x_k^T W x_k + u_{k,c}^T U u_{k,c}\right] = \mathbb{E}[V_0(x_0)]$. We find that

$$\bar{J}_N = \text{Pr}(\eta_{-1} = 0)\left(\mathbb{E}[x_0^T S_0 x_0 | \eta_{-1} = 0] + c_0\right) + \text{Pr}(\eta_{-1} = 1)\left(\mathbb{E}[x_0^T R_0 x_0 | \eta_{-1} = 1] + d_0\right).$$

Leveraging the fact that $\{\eta_k\}$ is stationary with $\text{Pr}(\eta_k = 0) = \frac{\beta}{\alpha+\beta}$ as well as (3.37) and (3.38), we obtain

$$\begin{aligned}
\bar{J}_N = \frac{1}{\alpha+\beta}\sum_{k=0}^{N-1}\Bigg(&-\alpha\text{tr}((F^T R_{k+1} F - A^T R_{k+1} A + L_{(m)}^T U L_{(m)})\tilde{P}) + \text{tr}((\beta S_{k+1} + \alpha R_{k+1})Q) \\
&+ \alpha\text{tr}((B^T R_{k+1} B + U)\mathcal{J})\Bigg) + \frac{\beta\mathbb{E}[x_0^T S_0 x_0 | \eta_{-1} = 0] + \alpha\mathbb{E}[x_0^T R_0 x_0 | \eta_{-1} = 1]}{\alpha+\beta}.
\end{aligned}$$

It can be shown (in a similar manner to the proof of Theorem 3.3) that the last term is bounded. Note $\bar{J} = \lim_{N\to\infty} \frac{1}{N}\bar{J}_{N-1}$. Moreover, from [42][Theorem 3, Lemma 4], $\{S_k\}, \{R_k\}$ converge to $S_{(m)}, R_{(m)}$, respectively. This proves the desired result. □

We now compute the expected correlation without attacks.

Theorem 3.3. *Suppose α and β are chosen so the resulting system has finite cost $J_{(m)}$ [42][Theorem 3] in the absence of a Gaussian watermark. Then, for the control system (3.1),(3.2) with IID Gaussian and Markovian watermark (3.21), we have*

$$\lim_{k\to\infty} \mathbb{E}[y_k^T y_k' | \mathcal{H}_0] = \frac{\text{tr}(C(\alpha X_1 + \beta X_0)C^T)}{\alpha+\beta}, \qquad (3.39)$$

where

$$X_0 = A((1-\alpha)X_0 + \alpha X_1)A^T, \tag{3.40}$$
$$X_1 = (A + BL_{(m)})(\beta X_0 + (1-\beta)X_1)(A + BL_{(m)})^T + B\mathcal{J}B^T$$

Proof. We begin with the Lemma below.

Lemma 3.2. $\forall\ M \in \mathbb{R}^{2n\times n}$, $\lim_{k\to\infty} \mathcal{L}_0^k(M) = 0$ *where,*

$$\mathcal{L}_0 \begin{pmatrix} X \\ Y \end{pmatrix} = \begin{bmatrix} A((1-\alpha)X + \alpha Y)A^T \\ (A + BL_{(m)})(\beta X + (1-\beta)Y)(A + BL_{(m)})^T \end{bmatrix}.$$

The proof is in the appendix. The closed loop dynamics are

$$x_{k+1} = (A + \eta_k BL_{(m)})x_k - \eta_k BL_{(m)}e_k + w_k + \eta_k B\Delta u_k$$
$$e_{k+1} = (A - KCA)e_k + (I - KC)w_k - Kv_{k+1},$$

where $e_k = x_k - \hat{x}_{k|k}$. From (3.30), when $\eta_k = 1$. we obtain

$$\begin{aligned}
&\mathbb{E}[x'_{k+1}x_{k+1}^T|\eta_k = 1] \\
&= F\mathbb{E}[x'_k x_k^T|\eta_k = 1]F^T - B\mathbb{E}[\Delta u_k e_k^T|\eta_k = 1](BL_{(m)})^T \\
&- F\left(\mathbb{E}[x'_k e_k^T|\eta_k = 1]L_{(m)}^T B^T - \mathbb{E}[x'_k w_k^T|\eta_k = 1]\right) + F\mathbb{E}[x'_k \Delta u_k^T|\eta_k = 1]B^T \\
&+ B\mathbb{E}[\Delta u_k x_k^T|\eta_k = 1]F^T + B\left(\mathbb{E}[\Delta u_k w_k^T|\eta_k = 1] + \mathbb{E}[\Delta u_k \Delta u_k^T|\eta_k = 1]B^T\right),
\end{aligned}$$

where $F = (A + BL_{(m)})$ and we implicitly condition on $\mathcal{H}_0$. x'_k is independent of $\Delta u_k, w_k, e_k$ and Δu_k is independent of x_k, w_k, e_k. Thus,

$$\mathbb{E}[x'_{k+1}x_{k+1}^T|\eta_k = 1] = (A + BL_{(m)})\mathbb{E}[x'_k x_k^T|\eta_k = 1](A + BL_{(m)})^T + B\mathcal{J}B^T. \tag{3.41}$$

Next, since the Markov process is stationary and x_k, x'_k and η_k are conditionally independent given η_{k-1}, we observe

$$\begin{aligned}
\mathbb{E}[x'_k x_k^T|\eta_k = 1] &= \mathbf{Pr}(\eta_{k-1} = 1|\eta_k = 1)\mathbb{E}[x'_k x_k^T|\eta_k = 1, \eta_{k-1} = 1] \\
&+ \mathbf{Pr}(\eta_{k-1} = 0|\eta_k = 1)\mathbb{E}[x'_k x_k^T|\eta_k = 1, \eta_{k-1} = 0], \\
&= (1-\beta)\mathbb{E}[x'_k x_k^T|\eta_{k-1} = 1] + \beta\mathbb{E}[x'_k x_k^T|\eta_{k-1} = 0].
\end{aligned} \tag{3.42}$$

It can be similarly shown that

$$\mathbb{E}[x'_{k+1}x_{k+1}^T|\eta_k = 0] = A\mathbb{E}[x'_k x_k^T|\eta_k = 0]A^T. \tag{3.43}$$

$$\mathbb{E}[x'_k x_k^T|\eta_k = 0] = \alpha\mathbb{E}[x'_k x_k^T|\eta_{k-1} = 1] + (1-\alpha)\mathbb{E}[x'_k x_k^T|\eta_{k-1} = 0]. \tag{3.44}$$

Letting $X_{k,j} = \mathbb{E}[x'_k x_k^T|\eta_{k-1} = j]$ we have

$$\begin{pmatrix} X_{k+1,0} \\ X_{k+1,1} \end{pmatrix} = \mathcal{L}_0 \begin{pmatrix} X_{k,0} \\ X_{k,1} \end{pmatrix} + \begin{bmatrix} 0 \\ B\mathcal{J}B^T \end{bmatrix}. \tag{3.45}$$

Since $\mathcal{L}_0$ is stable, $\lim_{k\to\infty}\mathbb{E}[x'_k x_k^T|\eta_{k-1} = 0]$ and $\lim_{k\to\infty}\mathbb{E}[x'_k x_k^T|\eta_{k-1} = 1]$ are obtained by solving a fixed point equation which has a unique solution X_0 and X_1. (3.40) immediately follows from (3.45). Next, we find that

$$\lim_{k\to\infty}\mathbb{E}[x'_k x_k^T] = \mathrm{Pr}(\eta_{k-1} = 1)X_1 + \mathrm{Pr}(\eta_{k-1} = 0)X_0 = \frac{\alpha X_1 + \beta X_0}{\alpha + \beta}, \tag{3.46}$$

Finally, to the conclude the proof, we observe that

$$\mathbb{E}[y_k^T y'_k] = \mathrm{tr}\left(\mathbb{E}[(y'_k y_k^T)]\right) = \mathrm{tr}\left(C\mathbb{E}[x'_k x_k^T]C^T\right). \tag{3.47}$$

□

Thus, the watermark design problem (3.34) is given by

$$\begin{aligned} \underset{\alpha,\beta,\mathcal{J}}{\text{maximize}} \quad & \frac{\mathrm{tr}(C(\alpha X_1 + \beta X_0)C^T)}{\alpha+\beta} \\ \text{subject to} \quad & \begin{pmatrix} X_0 \\ X_1 \end{pmatrix} = \mathcal{L}_0 \begin{pmatrix} X_0 \\ X_1 \end{pmatrix} + \begin{bmatrix} 0 \\ B\mathcal{J}B^T \end{bmatrix}, \\ & J_{(m)}(\alpha,\beta) + \mathrm{tr}((B^T R_{(m)} B + U)\mathcal{J}) \le \delta, \\ & 0 < \alpha, \beta \le 1. \end{aligned}$$

For fixed α and β, the problem is an efficiently solvable semidefinite program. However, to optimize over α and β, we have to solve multiple instances of the problem over a finite 2

dimensional space. Ideally a designer will sample the space sufficiently. Note, not all (α, β) in $(0,\ 1] \times (0,\ 1]$ are feasible as some selections of α and β lead to unbounded cost. Likewise, naturally occurring drops will constrain α and β. For instance, if we add an artificial Markovian drop process on top of a naturally occurring IID drop process with drop probability p_d, we know that $\alpha \leq (1 - p_d), (1 - \beta) \leq (1 - p_d)$.

Remark 3.4. *The optimal design of Watermark 1 requires solving multiple instances of a convex optimization problem with parameters varying over a bounded 2 dimensional space. This will also be true for Watermark 2. A formulation that considers a stationary Gaussian input with a Markovian drop process is nontrivial. Even if analysis can be performed, optimal design will likely require searching over 3 dimensions. This more complicated case is left for future work.*

3.2.5 IID Bernoulli - Stationary Gaussian Watermark

We now investigate a watermark consisting of stationary Gaussian noise generated by a HMM (3.23) and an IID Bernoulli drop process at the control input with drop probability equal to p_d. Again, we design a watermark to address a performance and security trade-off. We wish to solve:

$$\begin{aligned} \underset{p_d, A_\omega, C_h, \Psi}{\text{maximize}} \quad & \lim_{k\to\infty} \mathbb{E}[y_k^T y_k' | \mathcal{H}_0] \\ \text{subject to} \quad & \bar{J} \leq \delta, \ \ \rho(A_\omega) \leq \bar{\rho}, \\ & 0 \leq p_d \leq 1. \end{aligned} \tag{3.48}$$

Rather than optimizing over the parameters of the HMM, we instead optimize over the autocovariance functions $\Gamma(d) \triangleq \mathbb{E}[\Delta u_k \Delta u_{k+d}^T]$. As in section 2.1, for tractable analysis we replace the constraint $\rho(A_\omega) \leq \bar{\rho}$ with the following related assumption (identical to assumption 2.1.2)

Assumption 3.2.1. *Let $\Gamma(d)$ be an autocovariance function for a Gaussian process generated by an HMM (A_ω, C_h, Ψ). $(A_\omega, C_h, \Psi, \bar{\rho})$ is feasible only if $\tilde{\Gamma}(d) \triangleq \bar{\rho}^{-|d|}\Gamma(d)$ is a autocovariance function of a stationary Gaussian process.*

Remark 3.5. *Recall when $\bar{\rho} = 1$, assumption 3.2.1, introduces no relaxation. In fact, the resulting formulation optimizes all stationary Gaussian processes in general. However, in the case $\bar{\rho} = 1$, we will prove that the resulting Gaussian process $\{\Delta u_k\}$ is entirely deterministic except for the initial watermark. A lower parameter $\bar{\rho}$ reduces average performance, but prevents an attacker who learns or guesses the current hidden state from adequately predicting future watermarks.*

We arrive at a relaxed formulation to (3.48) below.

Theorem 3.4. *Consider the control system (3.1),(3.2) with IID Bernoulli and stationary Gaussian watermark (3.23). Suppose p_d is chosen so that the system has finite cost $J_{(b)}$ [42][Theorem 3] in the absence of a Gaussian watermark. An equivalent formulation to (3.48) after replacing the constraint $\rho(A_\omega) \leq \bar{\rho}$ with Assumption 3.2.1 is given by*

$$\begin{aligned} \underset{\omega, H, p_d}{\textit{maximize}} \quad & \textit{tr}(CF_2(\omega, H, p_d)C^T) \\ \textit{subject to} \quad & J_{(b)}(p_d) + F_1(\omega, H, p_d) \leq \delta, \\ & 0 \leq p_d \leq 1, \;\; 0 \leq \omega \leq 0.5, \\ & H \in \mathbb{C}^{p\times p}, \;\; H \geq 0. \end{aligned} \tag{3.49}$$

where

$$\begin{aligned} F_2(\omega, H, p_d) &= 2\,\mathfrak{Re}\left(2\textit{sym}\left[L_1(M_2 H B^T)\right] + L_1(BHB^T)\right) \\ F_1(\omega, H, p_d) &= \textit{tr}(U\Theta) + \textit{tr}((W + \bar{p}_d L_{(b)}^T U L_{(b)})F_2), \\ \Theta(\omega, H, p_d) &= 2\,\mathfrak{Re}\left(2\textit{sym}\left[\bar{p}_d M_1 H\right] + \bar{p}_d H\right), \\ M_2 &= \bar{p}_d \bar{\rho} s (A + BL_{(b)})\left[I - s\bar{\rho}(A + \bar{p}_d B L_{(b)})\right]^{-1} B, \\ M_1 &= \bar{p}_d \bar{\rho} s L_{(b)}\left[I - s\bar{\rho}(A + \bar{p}_d B L_{(b)})\right]^{-1} B, \\ L_1(X) &= \bar{p}_d\left((A + BL_{(b)})L_1(X)(A + BL_{(b)})^T + X\right) + p_d A L_1(X) A^T, \\ \textit{sym}(X) &= \frac{X + X^T}{2}, \quad s = \exp(2\pi j\omega), \quad \bar{p}_d = 1 - p_d. \end{aligned}$$

There is also an optimal solution (H_*, ω_*, p_{d*}) such that $H_* = hh^H$ where h^H denotes the conjugate transpose or adjoint of $h \in \mathbb{C}^p$. Letting $\mathfrak{Re}$ and $\mathfrak{Im}$ be the real and imaginary parts of a

matrix/vector, respectively, an optimal A_ω, C_h, Ψ *is*

$$A_\omega = \bar{\rho} \begin{bmatrix} \cos(2\pi\omega_*) & -\sin(2\pi\omega_*) \\ \sin(2\pi\omega_*) & \cos(2\pi\omega_*) \end{bmatrix},$$
$$C_h = \sqrt{2} \begin{bmatrix} \mathfrak{Re}(h) & \mathfrak{Im}(h) \end{bmatrix}, \quad \Psi = (1-\bar{\rho}^2)I. \tag{3.50}$$

The proof is similar in nature to the proof Theorem 2.6. A sketch is found in the appendix. For fixed p_d and ω, the proposed problem is an efficiently solvable semidefinite program. To approximate a global maximum, we solve the problem repeatedly over the space $0 \leq \omega \leq 0.5$ and $0 \leq p_d \leq 1$. For sufficiently large p_d, the cost $\bar{J}$ becomes infinite in open loop unstable systems [9], limiting the feasible space. We can account for natural packet drops in the system as before. For instance, if the input is dropped naturally with probability p'_d, we have $p'_d \leq p_d \leq 1$. Once more, the optimal watermark is a noisy sinusoid. As mentioned in the previous chapter, the linearity of the objective and constraints with respect to the autocovariance function in the frequency domain results in a single frequency being optimal.

Remark 3.6. *An optimal watermark for a given* $p_d \neq p_{d*}$ *may have better detection performance than the globally optimal watermark. Future work aims to use objective functions that better highlight the relative performance of watermarks.*

Remark 3.7. *In general, introducing intentional drops may or may not improve detection performance for a given LQG cost. Future work aims to specifically characterize systems where a jointly designed watermark can outperform a purely Gaussian watermark.*

Remark 3.8. *While packet drops at the sensor measurements are not modeled in this chapter, our framework could be extended to address this behavior without significantly changing the formulations of the proposed optimization problems. The main effect of packet drops at the sensor side is a time varying Kalman gain. The objective function and increase in cost* $\bar{J}$ *due to the Gaussian portion of the watermark are not affected by time variations in the Kalman gain in both*

watermarking settings. Both $J_{(m)}$ and $J_{(b)}$ can be empirically evaluated for fixed (α, β) and p_d, respectively, to account for packet drops at the sensor measurements.

3.2.6 Numerical Examples

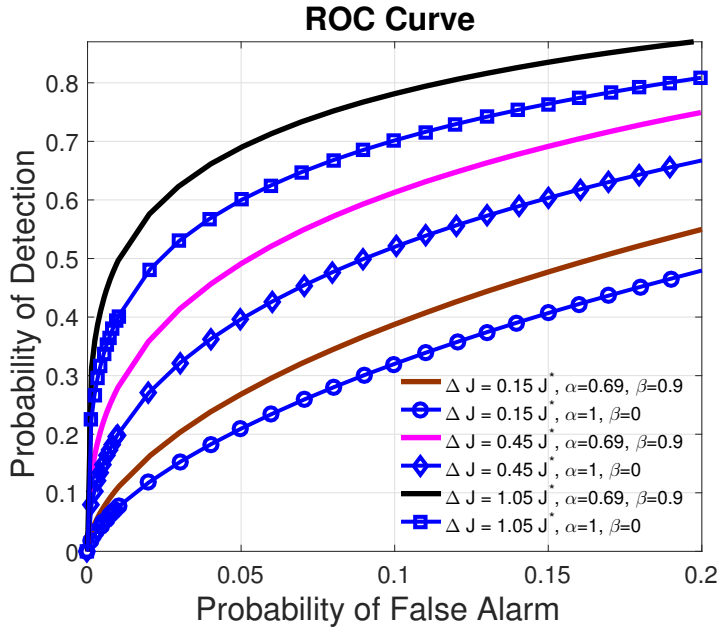
In this section, we illustrate the performance of the proposed watermarking designs through extensive numerical results. We tested our watermark designs in various randomly generated systems and, unless otherwise stated, averaged results over 1500 trials. Replay attacks are considered.

In Fig. 3.10, we utilize the first watermark, which has a Markovian drop process defined by parameters (α, β) and an IID Gaussian watermark. The watermark is tested on a randomly generated open loop stable system with 5 states, 4 inputs, and 2 outputs. We plot the receiver operating characteristic (ROC) curve for both the proposed correlation detector and a χ^2 detector. The χ^2 detector serves as a benchmark, having been previously used for attack detection [15, 27, 45] in watermarked systems. The threshold μ is chosen to be a constant multiple of $\lim_{k\to\infty} E[y_k^T y_k']$. The ROC curves are collected at multiple different costs $\Delta J = 1.05J^*$, $\Delta J = 0.45J^*$ and $\Delta J = 0.15J^*$. Here, ΔJ represent the increase in the cost $\bar{J}$ relative to optimal cost J^* without drops or a Gaussian watermark. We compare a system with drops $(\alpha = 0.69, \beta = 0.9)$ to a system without drops $(\alpha = 1, \beta = 0)$. The proposed detector outperforms the χ^2 detector in all cases and packet drops improve the ROC curve for both detectors. The improvement appears to be higher for moderately valued ΔJ before saturating. In Fig. 3.11, we plot the expected time to detection for both detectors in a system with the Markovian watermark. The packet drop process introduces an additional delay in the time to detection though this additional time is less significant as ΔJ is increased.

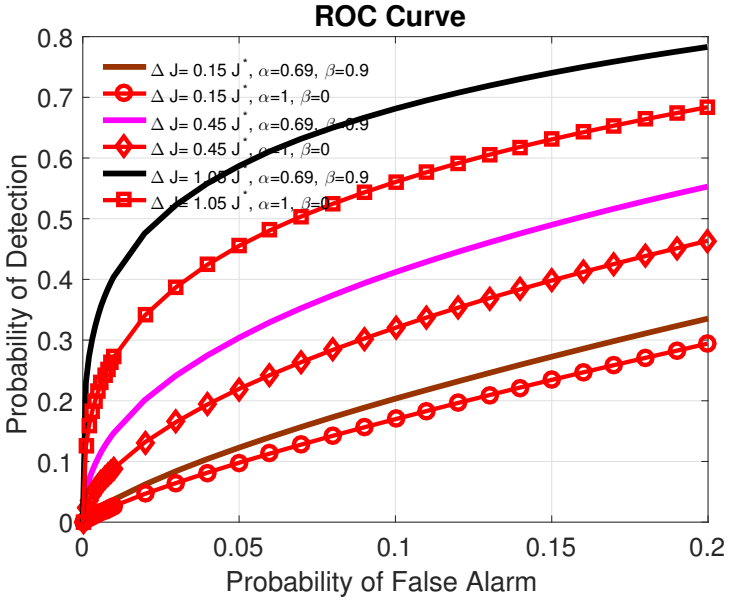
In Fig. 3.12, we introduce the second watermark, which has IID drops (with probability of drop p_d) and a stationary Gaussian watermark. The watermark is added to a randomly generated open loop stable system with 6 states, 5 inputs, and 5 outputs. We plot ROC curves generated by both the correlation detector and χ^2 detector for a system with drops $(p_d = 0.6)$ and a system without

drops ($p_d = 0$), at various costs of control $\Delta J = 0.95J^*$, $\Delta J = 0.45J^*$ and $\Delta J = 0.15J^*$. Time to detection plots are provided in Fig. 3.13. The results and patterns observed here are similar to the results seen in the system with the first watermark.

In Figs. 3.14 and 3.15, we plot χ^2 detector and correlation detector statistics (averaged over 500 trials) during a fault in the system. The fault introduced (at time 210) is a constant additive bias added to a subset of sensors (i.e. due to disturbances/sensor drift). While the χ^2 detector raises an alarm, the correlation detector does not since the watermark is preserved in the system. This motivates the use of both the correlation and χ^2 detector to distinguish faults from attacks. If both detectors raise an alarm, indicating the watermark is absent in the outputs, we consider a likely attack scenario. If only the χ^2 detector raises an alarm, we expect that the watermark is preserved while the dynamics are inconsistent with modeling. As such, we anticipate a fault.

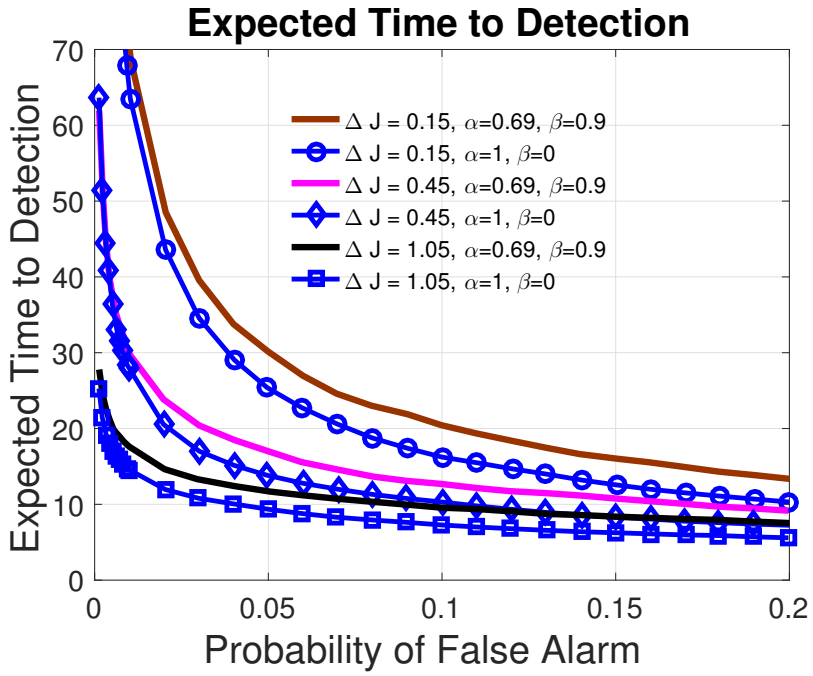


(a) Correlation Detector

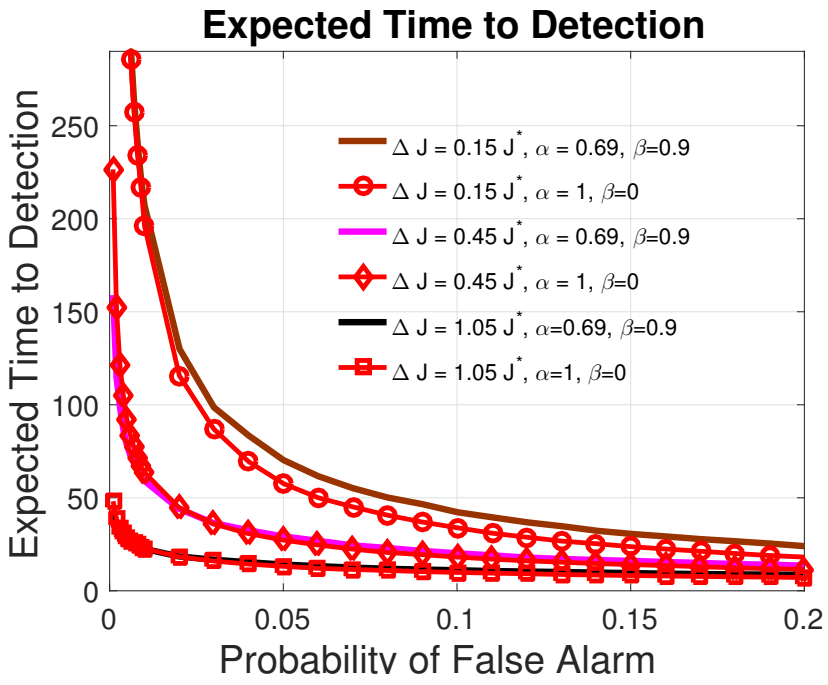


(b) χ^2 Detector

Figure 3.10: Detection probability versus false alarm rate for χ^2 and correlation detectors for a system using Markovian Bernoulli and IID Gaussian Watermark.

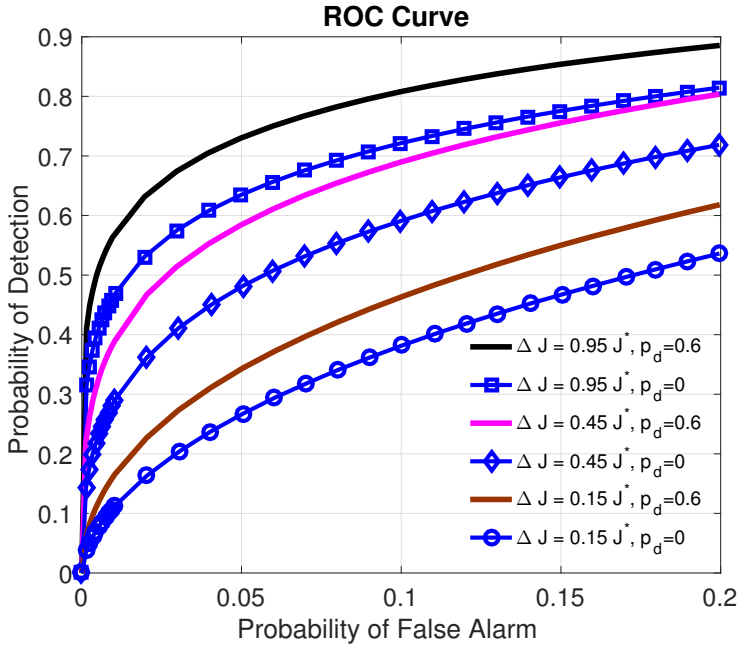


(a) Correlation Detector

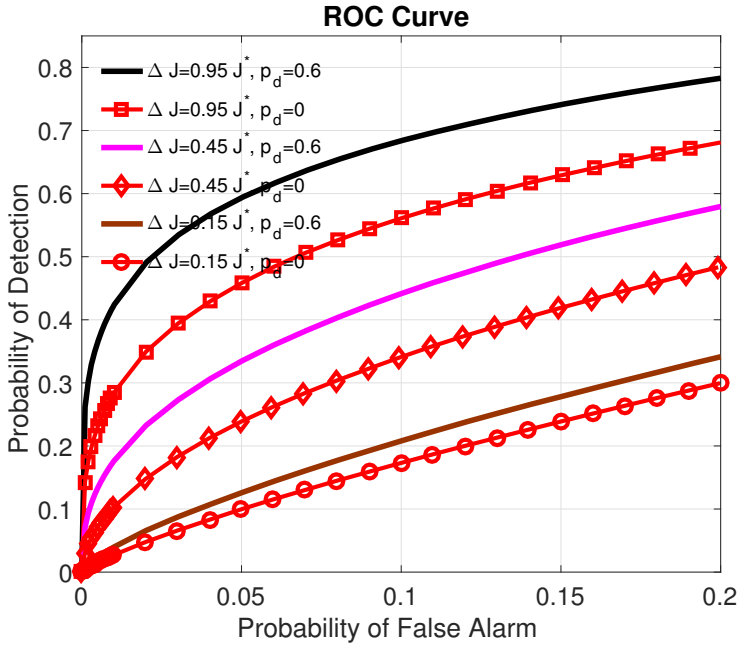


(b) χ^2 Detector

Figure 3.11: Expected time to detection for χ^2 and correlation detectors for a system using Markovian Bernoulli and IID Gaussian Watermark.

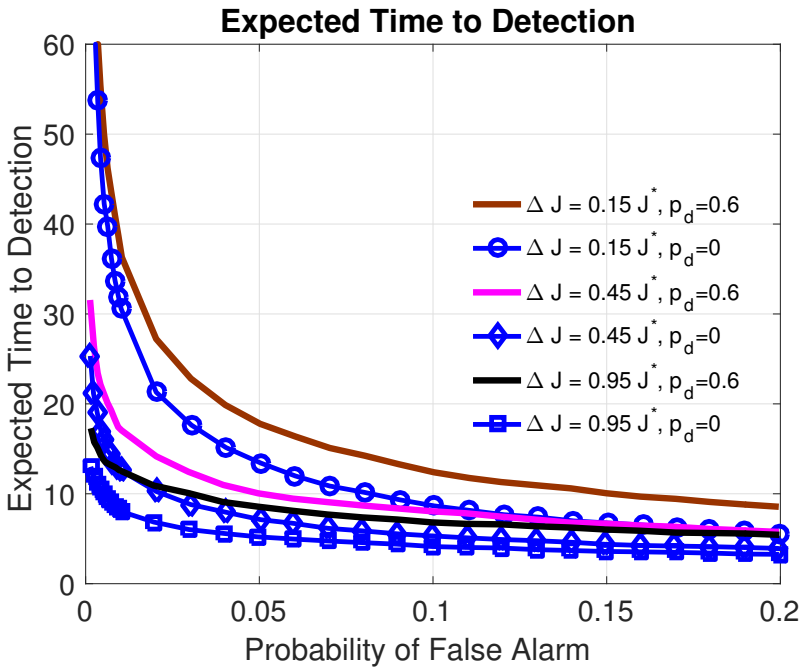


(a) Correlation Detector

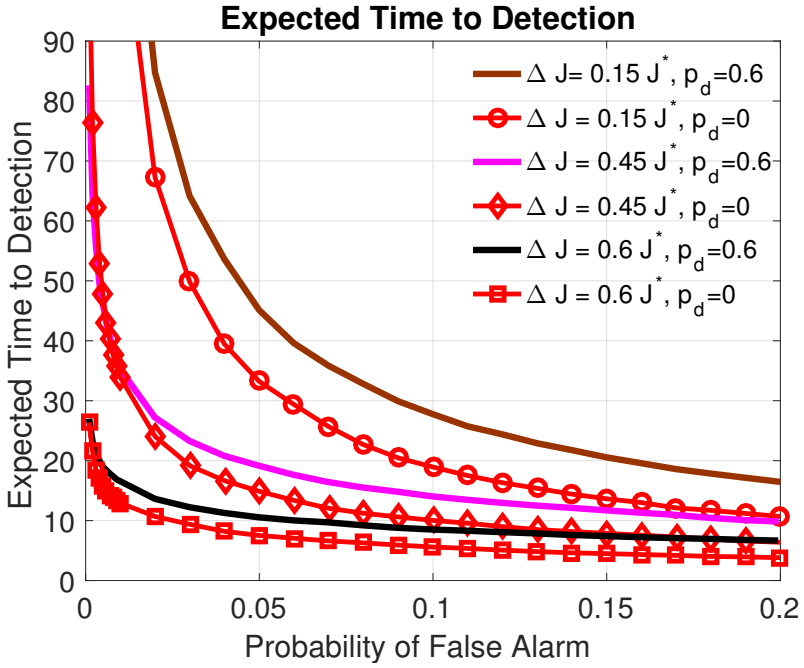


(b) χ^2 Detector

Figure 3.12: Detection probability versus false alarm rate for χ^2 and correlation detectors for a system using IID Bernoulli and Stationary Gaussian Watermark.

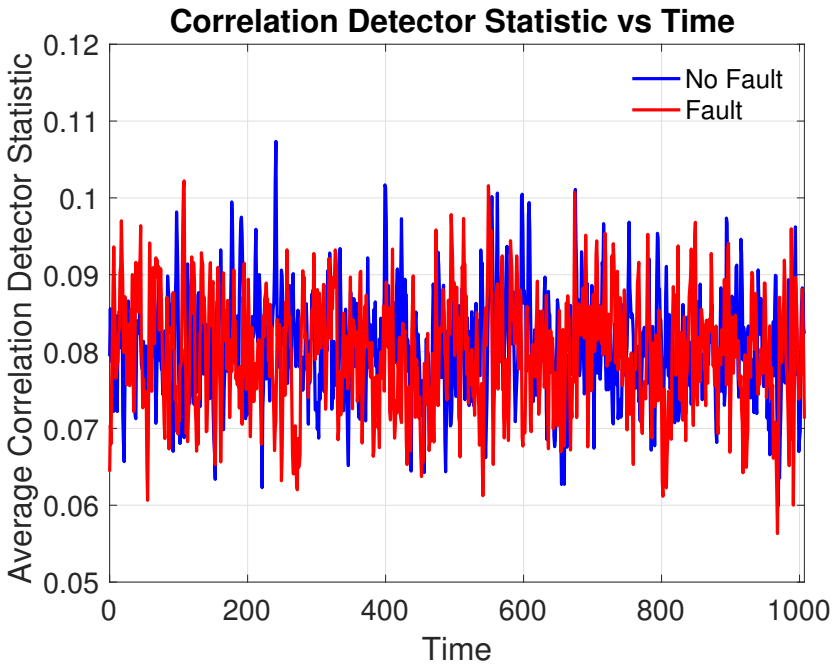


(a) Correlation Detector

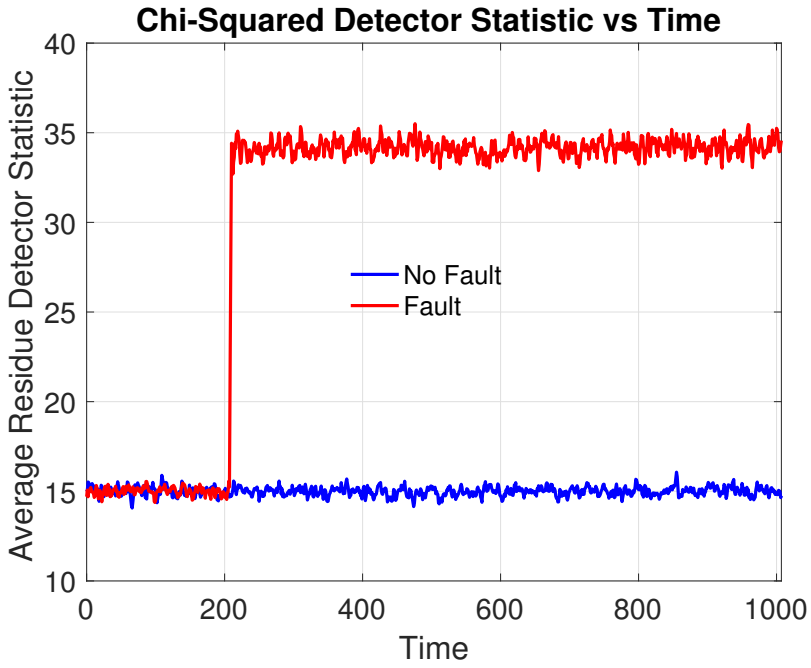


(b) χ^2 Detector

Figure 3.13: Expected time to detection for χ^2 and correlation detectors for a system using IID Bernoulli and Stationary Gaussian Watermark.

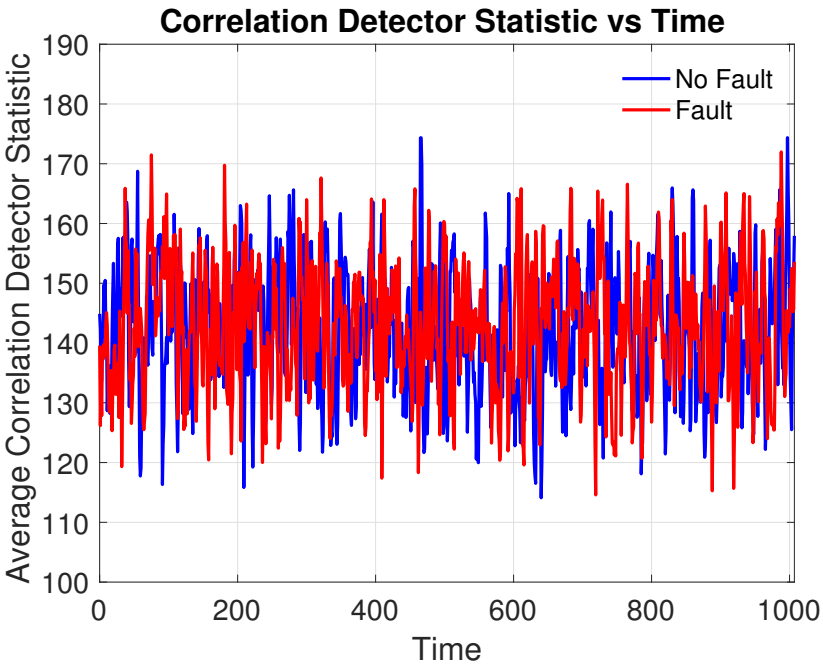


(a) Correlation Detector

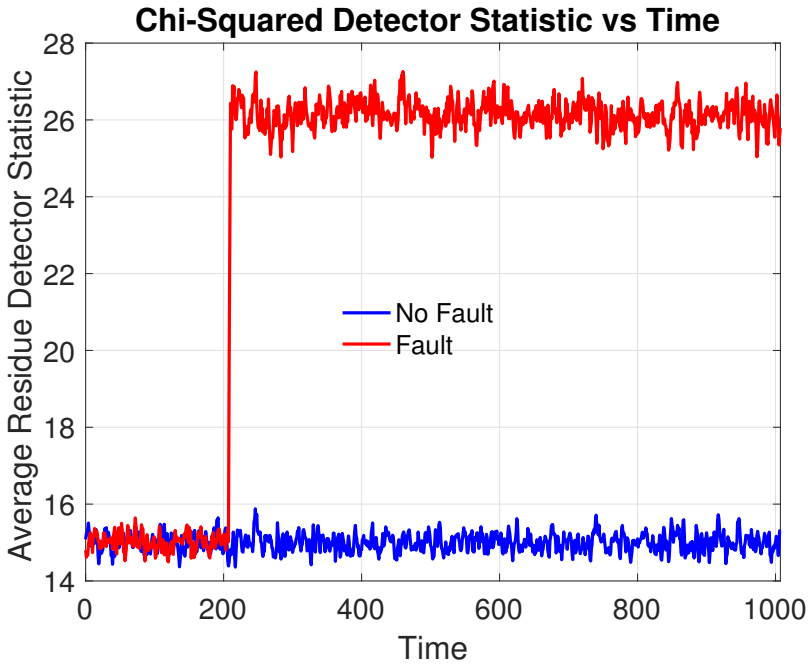


(b) χ^2 Detector

Figure 3.14: Average correlation detector and χ^2 detector statistics under a fault at the sensor output for a system using Markovian Bernoulli and IID Gaussian Watermark.



(a) Correlation Detector



(b) χ^2 Detector

Figure 3.15: Average correlation detector and χ^2 detector statistics under a fault at the sensor output for a system using IID Bernoulli and Stationary Gaussian Watermark.

Chapter 4

Moving Target Approach

In the previous chapters, we investigated how physical watermarking can be use to detect classes of stealthy attacks. In particular, we demonstrated physical watermarking as being particularly effective against replay attacks as well as some model aware attackers. However, model knowledge, when combined with channel access can lead to extremely powerful, yet stealthy attackers. The primary example is a covert attack, where an adversary is able to completely take control of a system, while using model knowledge to hide his impact. This chapter will examine how removing knowledge of the model allows us to actively detect attackers. Specifically, we introduce the moving target approach. In the moving target, the defender introduce time varying perturbations to the plant in order to limit an attacker's understanding of the system dynamics. By limiting an attacker's understanding of the system, the defender prevents an attacker from carrying out stealthy attacks, thus enabling active detection. This chapter introduces two methods for constructing a moving target. In section 4.1, we examine the addition of an authenticating subsystem. This subsystem, which can take the form of some external hardware will be affected by the dynamics of the true system. The time varying perturbations of the authenticating subsystem will be leveraged to actively detect an attacker. In section 4.2, we consider a plant with multiple discrete modes of operation (i.e. a hybrid system). We consider the design of a switched linear system to enable not only the detection, but also the isolation of malicious attackers. The results in this chapter are

largely based on [46] and [47].

4.1 The Authenticating Subsystem Approach

In this subsection, we consider the first method for a moving target, the authenticating subsystem approach. This approach is meant to counter an attacker with significant disclosure and disruption resources. In particular, we consider a strong adversary who can read and modify all input and sensor channels. If an attacker has knowledge of the system dynamics he or she can arbitrarily and stealthily perturb a system using a covert attack [21]. To prevent such a scenario, the defender has to keep the adversary unaware of the full system model. This can be challenging for several reasons. The dynamics of the system may be well known for instance by physical laws. Alternatively, an attacker can use his disclosure resource to learn the model through passive observations. We describe our approach to deal with this problem in this section.

4.1.1 System Description

As in the prior section our cyber-physical system can be modeled as a discrete time control system where

$$x_{k+1} = Ax_k + Bu_k + w_k, \tag{4.1}$$

$$y_k = Cx_k + v_k. \tag{4.2}$$

Here $x_k \in \mathbb{R}^n$ is the state vector at time k and $u_k \in \mathbb{R}^p$ is a collection of control inputs. A suite of sensors are used to monitor the state. Here $y_k \in \mathbb{R}^m$ is a vector of sensor measurements taken at time k. w_k is the independent and identically distributed (IID) process noise with probability distribution given by $\mathcal{N}(0, Q)$ where $Q \geq 0$. Meanwhile, v_k is the IID measurement noise with distribution given by $v_k \sim \mathcal{N}(0, R)$ where $R > 0$. We assume that (A, C) is detectable. Additionally, (A, B) and $(A, Q^{\frac{1}{2}})$ are assumed to be stabilizable.

As in prior chapters, a bad data detector can be utilized to determine whether a malicious attack is occurring.

$$g_k(\mathcal{I}_k) \underset{\mathcal{H}_0}{\overset{\mathcal{H}_1}{\gtrless}} \tau_k. \tag{4.3}$$

Here, $\mathcal{I}_k$ is the information available to the defender. The null hypothesis $\mathcal{H}_0$ is that the system is operating normally while the alternate hypothesis $\mathcal{H}_1$ is that the system is under attack. The probability of detection β_k and false alarm α_k are

$$\beta_k = \Pr\left(g_k\left(\mathcal{I}_k\right) > \eta_k | \mathcal{H}_1\right), \ \alpha_k = \Pr\left(g_k\left(\mathcal{I}_k\right) > \eta_k | \mathcal{H}_0\right). \tag{4.4}$$

Regardless of the chosen passive detector, an attacker with knowledge of the input to output model as well as the ability to manipulate sensor measurements and control inputs, can in theory generate undetectable attacks.

For instance, assume at time 0 an adversary simply subtract the influence he inserts through the control inputs from the system outputs as follows

$$x_{k+1} = Ax_k + B(u_k + u_k^a) + w_k, \tag{4.5}$$

$$y_k = Cx_k + v_k + d_k^a, \tag{4.6}$$

where d_k^a is given by

$$x_{k+1}^a = Ax_k^a + Bu_k^a, \ \ x_0^a = 0 \tag{4.7}$$

$$d_k^a = -Cx_k^a. \tag{4.8}$$

From the linearity of the system, we observe that $y_k = y_k'$ where y_k' is given by

$$x_{k+1}' = Ax_k' + Bu_k + w_k, \ \ x_0' = x_0 \tag{4.9}$$

$$y_k' = Cx_k' + v_k. \tag{4.10}$$

In this case, the attacker has zero net effect on the outputs and as a result is perfectly stealthy. Moreover, the attacker can cause significant damage by perturbing the system arbitrarily along the

control subspace of (A, B). We observe that watermarking techniques are ineffective against this attacker as the realization of the defender's control strategy is independent of the effectiveness of the attack. Here, we argue that knowledge of the system model is the predominant tool that allows an attacker to remain hidden. As such, we design a technique to limit the attacker's knowledge of the model, which we refer to as the moving target.

4.1.2 Modeling the Moving Target

We propose introducing extraneous states which are causally affected by the ordinary states of the system. The extraneous states are part of an authenticating subsystem, which has linear time-varying dynamics, known to the system operator and hidden from the adversary. The dynamics are designed so that an attacker who impacts the original system will necessarily impact the authenticating sybsystem. Moreover, the time varying dynamics ideally act as a moving target, changing fast enough so the adversary does not have adequate opportunity to identify the extraneous system. While essentially an attempt to prevent covert attacks, the moving target by removing an attacker's model knowledge, can also defend against weaker zero dynamics and false data injection attacks.

Mathematically, we introduce an authenticating subsystem with time varying dynamics on top of the original system as follows:

$$\begin{bmatrix} \tilde{x}_{k+1} \\ x_{k+1} \end{bmatrix} = \mathcal{A}_k \begin{bmatrix} \tilde{x}_k \\ x_k \end{bmatrix} + \mathcal{B}_k u_k + \begin{bmatrix} \tilde{w}_k \\ w_k \end{bmatrix}, \quad \mathcal{A}_k \triangleq \begin{bmatrix} A_{1,k} & A_{2,k} \\ 0 & A \end{bmatrix}, \quad \mathcal{B}_k \triangleq \begin{bmatrix} B_k \\ B \end{bmatrix}. \tag{4.11}$$

Moreover, we introduce additional sensors $\tilde{y}_k \in \mathbb{R}^{\tilde{m}}$ to measure the extraneous states.

$$\begin{bmatrix} \tilde{y}_k \\ y_k \end{bmatrix} = \mathcal{C}_k \begin{bmatrix} \tilde{x}_k \\ x_k \end{bmatrix} + \begin{bmatrix} \tilde{v}_k \\ v_k \end{bmatrix}, \quad \mathcal{C}_k \triangleq \begin{bmatrix} C_k & 0 \\ 0 & C \end{bmatrix}. \tag{4.12}$$

The matrices are taken as IID random variables which are independent of the sensor and process noise processes with distribution

$$A_{1,k}, A_{2,k}, B_k, C_{k+1} \sim f_{A_{1,k},A_{2,k},B_k,C_{k+1}}(A_1, A_2, B, C). \tag{4.13}$$

Furthermore, we also assume that

$$\begin{bmatrix} \tilde{w}_k \\ w_k \end{bmatrix} \sim \mathcal{N}(0, \mathcal{Q}), \quad \begin{bmatrix} \tilde{v}_k \\ v_k \end{bmatrix} \sim \mathcal{N}(0, \mathcal{R}), \tag{4.14}$$

where

$$\mathcal{Q} = \begin{bmatrix} \tilde{Q} & \tilde{Q}_{12} \\ \tilde{Q}_{12}^T & Q \end{bmatrix} \geq 0, \quad \mathcal{R} = \begin{bmatrix} \tilde{R} & \tilde{R}_{12} \\ \tilde{R}_{12}^T & R \end{bmatrix} > 0. \tag{4.15}$$

Since the moving target system is linear and the noises remain Gaussian, we can use a Kalman filter to still perform state estimation.

$$\begin{aligned} &\begin{bmatrix} \hat{\tilde{x}}_{k+1|k} \\ \hat{x}_{k+1|k} \end{bmatrix} = \mathcal{A} \begin{bmatrix} \hat{\tilde{x}}_{k|k} \\ \hat{x}_{k|k} \end{bmatrix} + \mathcal{B}_k u_k, \quad \begin{bmatrix} \hat{\tilde{x}}_{k|k} \\ \hat{x}_{k|k} \end{bmatrix} = (I - \mathcal{K}_k \mathcal{C}_k) \begin{bmatrix} \hat{\tilde{x}}_{k|k-1} \\ \hat{x}_{k|k-1} \end{bmatrix} + \mathcal{K}_k \begin{bmatrix} \tilde{y}_k \\ y_k \end{bmatrix}, \\ &z_k = \begin{bmatrix} \tilde{y}_k \\ y_k \end{bmatrix} - \mathcal{C}_k \begin{bmatrix} \hat{\tilde{x}}_{k|k-1} \\ \hat{x}_{k|k-1} \end{bmatrix}, \quad \mathcal{K}_k = \mathcal{P}_k \mathcal{C}_k^T (\mathcal{C}_k \mathcal{P}_k \mathcal{C}_k^T + \mathcal{R})^{-1}, \\ &\mathcal{P}_{k+1} = \mathcal{A}_k \mathcal{P}_k \mathcal{A}_k^T + \mathcal{Q} - \mathcal{A}_k \mathcal{P}_k \mathcal{C}_k^T (\mathcal{C}_k \mathcal{P}_k \mathcal{C}_k^T + \mathcal{R})^{-1} \mathcal{C}_k \mathcal{P}_k \mathcal{A}_k^T \end{aligned} \tag{4.16}$$

Here, $\mathcal{K}_k$ is the Kalman gain, $\mathcal{P}_k$ is the apriori state estimation error covariance, $\hat{\tilde{x}}_{k+1|k}, \hat{x}_{k+1|k}$ is the apriori state estimates and $\hat{\tilde{x}}_{k|k}, \hat{x}_{k|k}$ are the aposteriori state estimates. Given this, a χ^2 detector can be used for passive detection. Recall, in a χ^2 detector,

$$g_k(\mathcal{I}_k) = \sum_{t=k-WS+1}^{k} z_t^T (CPC^T + R)^{-1} z_t. \tag{4.17}$$

Under normal operation $z_t^T (CPC^T + R)^{-1} z_t$ should follow a χ^2 distribution with m degrees of freedom. The χ^2 detector attempts to exploit this fact by testing to see if the innovations follow the correct distribution. It is easy to see that large residues, indicating a discrepancy between measured and expected behavior create alarms, while smaller residues which indicate good agreement between measure and expected behavior are indicative of normal operation.

Remark 4.1. *While the system introduced above involves IID matrices $A_{1,k}, A_{2,k}, B_k, C_{k+1}$, the moving target design can still be effective in other scenarios. For instance, the dynamics need not*

be linear as long as the defender can accurately model the system. Moreover, the system parameters can evolve at multiple time scales. In this case, the longer the target remains in place, the easier it is for the adversary to identify the system.

Remark 4.2. *The defender must be able to introduce extraneous states with time-varying dynamics correlated to the original state of the system. The extraneous states are application dependent and are to be decided by the system operator. Nonetheless, the system operator can leverage extra products of the system, for instance the heat dissipated by a reaction or process. The dynamics can be made time-varying by changing conditions at the plant. Alternatively, the defender can introduce dynamics into the system. For instance, the defender can introduce RLC circuits which measure the states. Time varying dynamics can be incorporated by including variable resistors or capacitors. By varying the components of the circuit according to an IID distribution at each time step, the defender can generate IID system matrices.*

Remark 4.3. *Unlike physical watermarking the moving target approach does not need to result in a suboptimal control performance. Specifically, if we assume the defender does not care about controlling the extra states, then no online performance has to be sacrificed. The cost of the moving target approach is likely primarily developmental. In particular, a defender may have to expend financial resources along with man hours to design, build, or purchase hardware which can be used to generate an appropriate authenticating subsystem.*

In the above formulation we assume that the defender is aware of the real time system matrices although they are random. In general, this information should not be sent over the network since doing so amounts to the existence of a secure communication channel. The secure communication channel could be leveraged to detect an attack without considering a moving target approach. Alternatively, we can generate pseudo random system matrices using a pseudo random number generator (PRNG). In this case, the seed of the PRNG is known to the defender and kept hidden from the attacker. The moving target approach is illustrated in Fig. 4.1

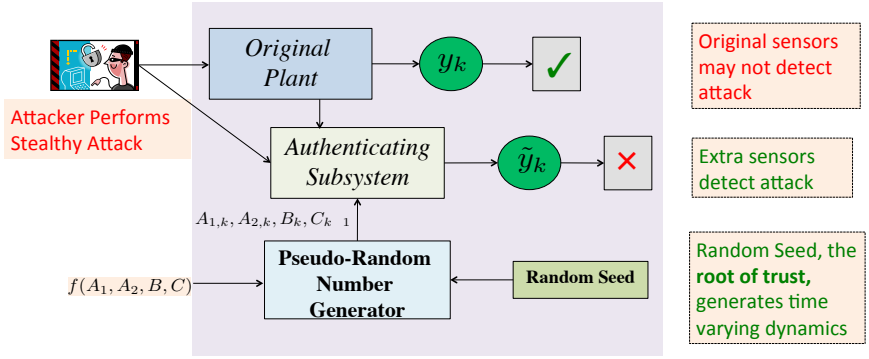


Figure 4.1: Moving Target for Active Detection in Cyber-Physical Systems

The moving target shares similarities with message authentication codes or MACs used for authentication in cyber security. This is described below. **Example:** In cyber security, MACs can be used to verify the integrity of a message. A message authentication code is computed by computing a keyed pseudorandom function of the sender's message. The receiver obtains both the sender's message and the MAC. The receiver, using the secret key shared with the sender, verifies that the MAC corresponds to the message. An attacker who attempts to modify the message will almost certainly fail to generate an appropriate MAC because he or she does not have access to the shared key.

We argue that the moving target approach allows us to introduce a cyber physical MAC. In the context of the moving target approach, suppose the message $\underline{m}$ corresponds to outputs y_k while the MAC is $\tilde{y}_k$. The MAC $\tilde{y}_k$ is correlated to the message y_k through the state x_{k-1} and the input u_{k-1}. The key is the seed which determines the sequence of system matrices. The defender uses knowledge of y_k and the sequence of system matrices to estimate $\tilde{y}_k$. Under normal operation, $\tilde{y}_k$ and its estimate $\hat{\tilde{y}}_k$ closely agree, as seen by a residue based detector, and as a result the MAC is verified.

On the other hand, suppose an adversary performs integrity attacks using knowledge of (A, B, C, Q, R). The attacker could generate convincing outputs y_k, while biasing the states x_k through a false data injection or zero dynamics attack. At the same time, (s)he will also bias the states $\tilde{x}_k$ and thus the MAC outputs $\tilde{y}_k$ if the time varying matrices are properly chosen. Having no

knowledge of the seed, the adversary can not know the time varying matrices. Moreover, the time varying dynamics act as a moving target, hindering system identification. As a result, the attacker can not generate a convincing cyber-physical MAC output $\tilde{y}_k$.

4.1.3 Attack Model against Moving Target

In this subsection, we consider possible attacks on the moving target. This will motivate an examination of bounds which characterize fundamental detectability with this approach. We assume the following attacker capabilities:

1) The attacker can insert arbitrary inputs into the system and can arbitrarily alter the sensor measurements. As a result, when under attack, the system has dynamics given by

$$\begin{bmatrix} \tilde{x}_{k+1} \\ x_{k+1} \end{bmatrix} = \mathcal{A}_k \begin{bmatrix} \tilde{x}_k \\ x_k \end{bmatrix} + \mathcal{B}_k(u_k + u_k^a) + \begin{bmatrix} \tilde{w}_k \\ w_k \end{bmatrix}, \tag{4.18}$$

$$\begin{bmatrix} \tilde{y}_k^a \\ y_k^a \end{bmatrix} = \mathcal{C}_k \begin{bmatrix} \tilde{x}_k \\ x_k \end{bmatrix} + \begin{bmatrix} \tilde{v}_k \\ v_k \end{bmatrix} + \begin{bmatrix} \tilde{d}_k^a \\ d_k^a \end{bmatrix} = \begin{bmatrix} \tilde{y}_k \\ y_k \end{bmatrix} + \begin{bmatrix} \tilde{d}_k^a \\ d_k^a \end{bmatrix}. \tag{4.19}$$

where u_k^a is the attacker's control input and $\tilde{d}_k^a$ and d_k^a are the biases injected on the extraneous sensors and ordinary sensors respectively.

2) The attacker can read the true outputs of the system $\tilde{y}_k, y_k$ and the inputs being sent by the defender to the plant u_k for all time k. Note that this essentially corresponds to a man in the middle attack occurring between the plant and system operator so that the attacker can manipulate and read all communication channels arbitrarily.

3) The attacker has full knowledge of the system model $\mathcal{M} \triangleq \{A, B, C, K, L, \mathcal{Q}, \mathcal{R}\}$. Moreover, the adversary knows the probability density function (pdf) of random matrices $A_{1,k}, A_{2,k}, B_k, C_{k+1}$. While conservative, the adversary can obtain his knowledge of the system model by observing the communication channels for an extended period of time and performing system identification.

Note, we introduce some slight notational differences from the attacks modeled in the previous chapters. In particular, to more easily distinguish the attacker's information and the defender's

information, we differentiate between the modified outputs the defender receives $\tilde{y}_k^a, y_k^a$ and the true outputs of the system $\tilde{y}_k, y_k$.

Based on the above definitions we can define the private information available to the attacker ($\mathcal{I}_k^A$) and defender ($\mathcal{I}_k^D$) and the public information ($\mathcal{I}_k^P$) available to all parties at time k in the same order as follows:

$$\mathcal{I}_k^A \triangleq \{\tilde{y}_j, y_j, \tilde{d}_{j-1}^a, d_{j-1}^a, u_{j-1}^a\} \quad \forall\, j \leq k, \tag{4.20}$$

$$\mathcal{I}_k^D \triangleq \{A_{1,j-1}, A_{2,j-1}, B_{j-1}, C_j\} \quad \forall\, j, \tag{4.21}$$

$$\mathcal{I}_k^P \triangleq \{\mathcal{M}, f(A_1, A_2, B, C), u_{j-1}, \tilde{y}_{j-1}^a, y_{j-1}^a\} \quad \forall\, j \leq k. \tag{4.22}$$

Thus the defender's information is $\mathcal{I}_k \triangleq \mathcal{I}_k^D \cup \mathcal{I}_k^P$ while the attacker's information is $\mathcal{I}_k^a \triangleq \mathcal{I}_k^A \cup \mathcal{I}_k^P$.

We illustrate this moving target adversary via the cyber-physical attack space, see Fig. 4.2. While the attacker has full disclosure and disruption resource here, in particular the ability to read and modify all inputs and outputs, the attacker is limited by his or her imperfect understanding of the authenticating subsystem.

We now propose two main attack strategies. Without loss of generality we assume any attack begins at $k = 0$.

Attack Strategy 1 - Subtract Influence:

In the first attack strategy the attacker aims to estimate his influence on the control system and subtract it. Define $\bar{d}_k^a \triangleq [\tilde{d}_k^{a\,T}\ d_k^{a\,T}]^T$. Recall that if

$$\bar{x}_{k+1}^a = \mathcal{A}_k \bar{x}_k^a + \mathcal{B}_k u_k^a, \quad \bar{d}_k^a = -\mathcal{C}_k \bar{x}_k^a, \tag{4.23}$$

with initial state $\bar{x}_0^a = 0$, an attack is completely stealthy. As the adversary does not know the time varying matrices, we assume he computes an estimate of $\mathcal{C}_k \bar{x}_k^a$ and uses that to subtract his influence on the sensor measurements. Thus, we would have

$$\bar{d}_k^a = -\mathbb{E}[\mathcal{C}_k \bar{x}_k^a | \mathcal{I}_k^a]. \tag{4.24}$$

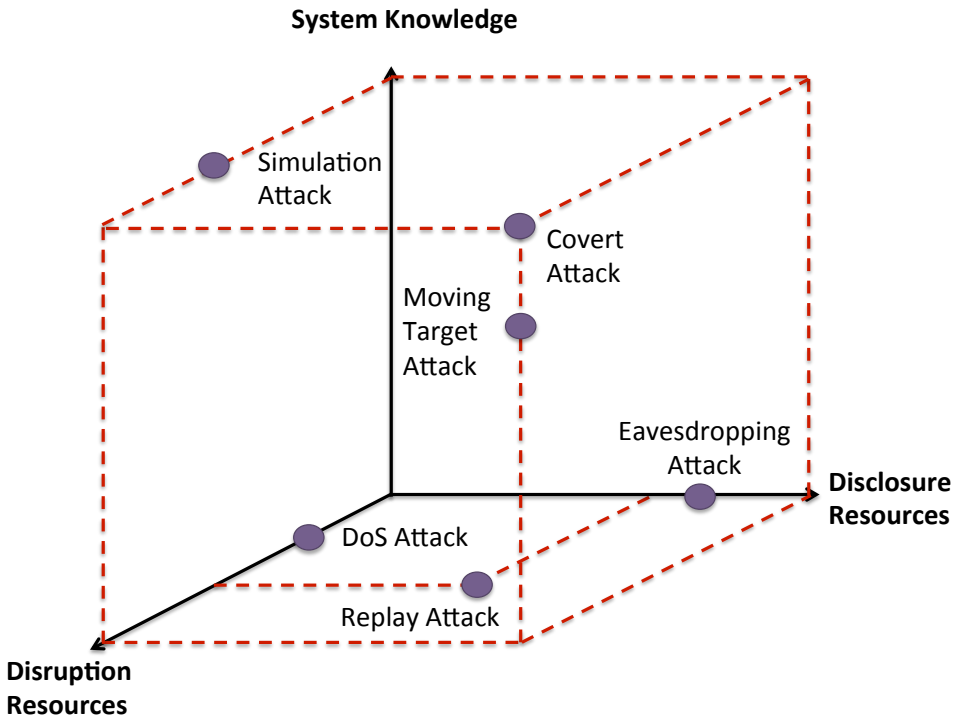


Figure 4.2: Cyber-Physical Attack Space with Moving Target Adversary

Observe that the adversary can exactly subtract his influence from measurements y_k due to his knowledge of the system model. However, the adversary should be unable to completely subtract his bias from the extraneous sensors $\tilde{y}_k$.

Define $\bar{y}_k^a \triangleq [\tilde{y}_k^{aT}\ y_k^{aT}]^T$, $\bar{x}_k \triangleq [\tilde{x}_k^T\ x_k^T]^T$, $\bar{w}_k \triangleq [\tilde{w}_k^T\ w_k^T]^T$, $\bar{v}_k \triangleq [\tilde{v}_k^T\ v_k^T]^T$, and $\bar{y}_k \triangleq [\tilde{y}_k^T\ y_k^T]^T$. The adversary's observations can be formulated through the following linear time-varying system,

$$\begin{bmatrix} \bar{x}_{k+1} \\ \bar{x}_{k+1}^a \end{bmatrix} = \begin{bmatrix} \mathcal{A}_k & 0 \\ 0 & \mathcal{A}_k \end{bmatrix} \begin{bmatrix} \bar{x}_k \\ \bar{x}_k^a \end{bmatrix} + \begin{bmatrix} \mathcal{B}_k & \mathcal{B}_k \\ 0 & \mathcal{B}_k \end{bmatrix} \begin{bmatrix} u_k \\ u_k^a \end{bmatrix} + \begin{bmatrix} \bar{w}_k \\ 0 \end{bmatrix}, \tag{4.25}$$

$$\bar{y}_k = \begin{bmatrix} \mathcal{C}_k & 0 \end{bmatrix} \begin{bmatrix} \bar{x}_k \\ \bar{x}_k^a \end{bmatrix} + \bar{v}_k. \tag{4.26}$$

To estimate $\Delta\bar{y}_k^a$ at time k, assume the adversary knows the following distribution $f(\bar{x}_k, \bar{x}_k^a, C_k | \mathcal{I}_k^a)$.

Then we have

$$\bar{d}_k^a = -\int_{\bar{x}_k}\int_{\bar{x}_k^a}\int_{C_k} \mathcal{C}_k \bar{x}_k^a f(\bar{x}_k, \bar{x}_k^a, C_k|\mathcal{I}_k^a)\mathrm{d}\bar{x}_k\mathrm{d}\bar{x}_k^a\mathrm{d}C_k. \tag{4.27}$$

We show that the pdf can be recursively computed at each step. Letting $\zeta_{k+1} = \{\bar{x}_{k+1}, \bar{x}_{k+1}^a, C_{k+1}\}$ we have

$$\begin{aligned} f(\zeta_{k+1}|\mathcal{I}_{k+1}^a) &= f(\zeta_{k+1}|\mathcal{I}_k^a, \bar{y}_k^a, \bar{y}_{k+1}, \bar{d}_k^a, u_k^a, u_k), \\ &= f(\zeta_{k+1}|\mathcal{I}_k^a, \bar{y}_{k+1}, u_k^a, u_k), \\ &= \frac{f(\bar{y}_{k+1}|\mathcal{I}_k^a, \zeta_{k+1}) f(\zeta_{k+1}|\mathcal{I}_k^a, u_k, u_k^a)}{f(\bar{y}_{k+1}|\mathcal{I}_k^a, u_k, u_k^a)}. \end{aligned} \tag{4.28}$$

The second equality follows from the conditional independence of ζ_{k+1} and $\bar{y}_k^a, \bar{d}_k^a$ given $\bar{y}_k$ and u_k. The last equality follows from Bayes rule and the conditional independence of $\bar{y}_{k+1}$ and u_k, u_k^a given ζ_{k+1}. We note that this distribution can be theoretically computed given the attacker's information. That is, we know that

$$f(\bar{y}_{k+1}|\mathcal{I}_k^a, \zeta_{k+1}) \sim \mathcal{N}\left(\mathcal{C}_{k+1}\bar{x}_{k+1}, \mathcal{R}\right). \tag{4.29}$$

Moreover, ζ_{k+1} and $\bar{y}_{k+1}$ are deterministic functions of ζ_k, u_k, u_k^a and random variables $A_{1,k}$, $A_{2,k}$, B_k, C_{k+1}, $\bar{w}_k$, $\bar{v}_{k+1}$, which are independent of ζ_k given $\mathcal{I}_k^a$. Thus, $f(\zeta_{k+1}|\mathcal{I}_{k+1}^a)$ can be recursively computed from $f(\zeta_k|\mathcal{I}_k^a)$.

Remark 4.4. *If the attacker subtracts his influence, he might be susceptible to a growing cancellation error if he attempts to excite the system's unstable dynamics. Instead of subtracting his influence the attacker can instead directly estimate what the defender expects to see as summarized in the next section.*

Attack Strategy 2 - Estimate the Defender's State Estimate:

In the next strategy, the adversary aims to track the system operator's state estimate. This attack is very similar to the robust attack formulated in Chapter 2. Using the system operator's state estimate, the adversary attempts to generate stealthy outputs. Let $\hat{\bar{x}}_k = [\hat{\bar{x}}_{k|k-1}^T \hat{x}_{k|k-1}^T]^T$. The

attacker's observations and strategy can be formulated as follows

$$\begin{bmatrix} \bar{x}_{k+1} \\ \hat{\bar{x}}_{k+1} \end{bmatrix} = \begin{bmatrix} \mathcal{A}_k & 0 \\ 0 & \mathcal{A}_k(I - \mathcal{K}_k\mathcal{C}_k) \end{bmatrix} \begin{bmatrix} \bar{x}_k \\ \hat{\bar{x}}_k \end{bmatrix} + \begin{bmatrix} \bar{w}_k \\ 0 \end{bmatrix} + \begin{bmatrix} \mathcal{B}_k & \mathcal{B}_k & 0 \\ \mathcal{B}_k & 0 & \mathcal{A}_k\mathcal{K}_k \end{bmatrix} \begin{bmatrix} u_k \\ u_k^a \\ \bar{y}_k^a \end{bmatrix}, \tag{4.30}$$

$$\bar{y}_k = \begin{bmatrix} \mathcal{C}_k & 0 \end{bmatrix} \begin{bmatrix} \bar{x}_k \\ \hat{\bar{x}}_k \end{bmatrix} + \bar{v}_k, \quad \bar{d}_k^a = \mathbb{E}[\mathcal{C}_k\hat{\bar{x}}_k|\mathcal{I}_k^a] - \bar{y}_k. \tag{4.31}$$

The attacker wishes to track $\zeta_k = \{\bar{x}_k, \hat{\bar{x}}_k, C_k, \mathcal{P}_k\}$. The use of the preceding attack design is motivated by the ensuing result which states that the chosen attack vector minimizes a fixed quadratic function of the measurement residues. This illustrates the potential effectiveness of the attack, when countered by a χ^2 detector.

Theorem 4.1. *Let $\Sigma \succeq 0$ be a positive semidefinite matrix.*

$$\mathbb{E}[\mathcal{C}_k\hat{\bar{x}}_k|\mathcal{I}_k^a] - \bar{y}_k = \arg\min_{\bar{d}_k^a} \mathbb{E}[z_k^T\Sigma z_k|\mathcal{I}_k^a]. \tag{4.32}$$

Proof. Observe that

$$\mathbb{E}[z_k^T\Sigma z_k|\mathcal{I}_k^a] = \int_{\zeta_k} z_k^T\Sigma z_k f(\zeta_k|\mathcal{I}_k^a)\mathrm{d}\zeta_k. \tag{4.33}$$

Taking the gradient with respect to $\bar{d}_k^a$ and setting the resulting expression equal to 0, we obtain

$$\int_{\zeta_k} \Sigma(\bar{y}_k + \bar{d}_k^a - \mathcal{C}_k\hat{\bar{x}}_k) f(\zeta_k|\mathcal{I}_k^a)\mathrm{d}\zeta_k = 0. \tag{4.34}$$

Solving gives

$$\bar{d}_k^a = -\bar{y}_k + \int_{\zeta_k} \mathcal{C}_k\hat{\bar{x}}_k f(\zeta_k|\mathcal{I}_k^a)\mathrm{d}\zeta_k, \tag{4.35}$$

and the result holds. □

To determine $\bar{d}_k^a$ at time k assume the adversary has access to the following distribution $f(\zeta_k|\mathcal{I}_k^a)$. As done before, the attacker can theoretically compute $\bar{d}_k^a$ by taking a conditional expectation. Additionally, similar to (4.28) we have

$$f(\zeta_{k+1}|\mathcal{I}_{k+1}^a) = \frac{f(\bar{y}_{k+1}|\mathcal{I}_k^a, \zeta_{k+1}) f(\zeta_{k+1}|\mathcal{I}_k^a, u_k, u_k^a, \bar{y}_k^a)}{f(\bar{y}_{k+1}|\mathcal{I}_k^a, u_k, u_k^a, \bar{y}_k^a)}. \tag{4.36}$$

Moreover, by similar analysis as in the first attack strategy, we can demonstrate that $f(\zeta_{k+1}|\mathcal{I}^a_{k+1})$ can be recursively computed from $f(\zeta_k|\mathcal{I}^a_k)$. The main difference here is that the adversary must also estimate $\mathcal{P}_k$.

In practice the proposed attacks are likely impossible to execute for an adversary since it is numerically intractable to compute the necessary distribution functions and expected values. This makes it difficult in general to quantify the potential detectability of intelligent attackers. As a result, we next aim to provide bounds on the attacker's estimation performance in terms of mean square error matrices.

4.1.4 Bounds on Attack Detectability

We now attempt to characterize lower bounds on the error matrices associated with the states ζ_k defined in attack strategy 1 and 2. This will be a measure of how well an attacker can estimate the relevant states in our moving target system. From there, we can attempt to characterize how well the adversary can design $\bar{d}^a_k$ to fool the bad data detector. We leverage conditional posterior Cramer-Rao lower bounds for Bayesian sequences derived by [48]. The authors here make use of the Bayesian Cramer-Rao lower bound or Van Trees bound derived in [49] which states that for observations y and states ζ the mean squared error matrix is bounded by the Fisher information as follows

$$\mathbb{E}_{f(\zeta,y)}\left[[\hat{\zeta}(y)-\zeta][\hat{\zeta}(y)-\zeta]^T\right] \geq I^{-1}, \tag{4.37}$$

where the Fisher information matrix I is given by

$$I = \mathbb{E}_{f(\zeta,y)}\left[-\triangle^{\zeta}_{\zeta}\log f(\zeta,y)\right]. \tag{4.38}$$

Note that

$$\triangle^{y}_{x} g(x,y) \triangleq \nabla_x \nabla^T_y g(x,y),$$

where ∇ is the gradient operator. In [48], this result is extended to nonlinear Bayesian sequences with dynamics given by

$$\zeta_{k+1} = F_k(\zeta_k, \omega_k), \;\; \bar{y}_k = G_k(\zeta_k, \bar{v}_k), \tag{4.39}$$

where ω_k and $\bar{v}_k$ are independent process and sensor noise respectively. In our case, we slightly adapt these results to account for the fact there is feedback in our system so that

$$\zeta_{k+1} = F_k(\zeta_k, \bar{y}_{1:k}, \omega_k), \quad \bar{y}_k = G_k(\zeta_k, \bar{v}_k). \tag{4.40}$$

The inputs u_k, u_k^a and $\bar{d}_k^a$ are incorporated into the definition of F_k, while uncertainty in the model $(A_{1,k}, A_{2,k}, B_k, C_{k+1})$ can be incorporated in the process noise ω_k.

To obtain a conditional Cramer Rao lower bound, let $f_{k+1}^c \triangleq f(\zeta_{0:k+1}, \bar{y}_{k+1}|\bar{y}_{1:k})$.

Assumption 4.1.1. *For any entry ζ of $\zeta_{0:k+1}$, $\bar{y}_{k+1}$, $\frac{\partial f_{k+1}^c}{\partial \zeta}$ and $\frac{\partial^2 f_{k+1}^c}{\partial \zeta^2}$ exist and both are absolutely integrable with respect to $\zeta_{0:k+1}$ and $\bar{y}_{k+1}$*

Assumption 4.1.2. *For any entry ζ^i of $\zeta_{0:k+1}$, ζ^i is defined over a compact interval $-\infty \leq a_i \leq \zeta^i \leq b_i \leq \infty$. Moreover,*

$$\lim_{\zeta^i \to a_i} f(\zeta_{0:k+1}) = \lim_{\zeta^i \to a_i} a_i f(\zeta_{0:k+1}) = \lim_{\zeta^i \to b_i} b_i f(\zeta_{0:k+1}) = \lim_{\zeta^i \to b_i} f(\zeta_{0:k+1}) = 0.$$

Then from Proposition 1 of Chapter 3 in [50], we have.

$$\mathbb{E}_{f_{k+1}^c}\left[\bar{e}_{0:k+1}\bar{e}_{0:k+1}^T|\bar{y}_{1:k}\right] \geq I^{-1}(\zeta_{0:k+1}|\bar{y}_{1:k}), \tag{4.41}$$

where

$$\bar{e}_{0:k+1} \triangleq \zeta_{0:k+1} - \hat{\zeta}_{0:k+1}(\bar{y}_{k+1}|\bar{y}_{1:k}), \tag{4.42}$$

$$I(\zeta_{0:k+1}|\bar{y}_{1:k}) \triangleq \mathbb{E}_{f_{k+1}^c}\left[-\triangle_{\zeta_{0:k+1}}^{\zeta_{0:k+1}} \log f_{k+1}^c|\bar{y}_{1:k}\right]. \tag{4.43}$$

Remark 4.5. *We remark that since F_k is defined by inputs u_k, u_k^a and $\bar{s}_k^a$, f_{k+1}^c is implicitly conditioned on $u_{0:k}, \bar{s}_{1:k}^a, u_{0:k}^a$. Moreover, f_{k+1}^c is defined given the adversary's knowledge of $\mathcal{M}, f(A_1, A_2, B, C)$.*

Observe that (4.41) gives us an expected lower bound for the error matrix associated with the entire state history $\zeta_{0:k+1}$ with knowledge of measurements $\bar{y}_{1:k}$. This expectation is taken over the state history as well the measurement $\bar{y}_{k+1}$ so that $\hat{\zeta}_{0:k+1}$ is a function of the measurement $\bar{y}_{k+1}$. Observe that unlike the traditional Cramer-Rao bound which is limited to unbiased estimators, the Bayesian Cramer-Rao bound here considers both biased and unbiased estimators $\hat{\zeta}$.

While the lower bound given here applies to the entire state history $\zeta_{0:k+1}$, in practice we care about estimating a lower bound on the current state ζ_{k+1}. Nonetheless, it can be easily shown that

$$\mathbb{E}_{f^c_{k+1}}\left[e_{k+1}e^T_{k+1}|\bar{y}_{1:k}\right] \geq I^{-1}(\zeta_{k+1}|\bar{y}_{1:k}), \tag{4.44}$$

where $I^{-1}(\zeta_{k+1}|\bar{y}_{1:k})$ is the $\dim(\zeta_k) \times \dim(\zeta_k)$ lower right submatrix of $I^{-1}(\zeta_{0:k+1}|\bar{y}_{1:k})$. In practice, computing $I^{-1}(\zeta_{k+1}|\bar{y}_{1:k})$ from $I^{-1}(\zeta_{0:k+1}|\bar{y}_{1:k})$ is impractical since it requires computing and taking the inverse of a Fisher information matrix which grows in dimension at each time step. As a result, we would like a recursion to compute $I^{-1}(\zeta_{k+1}|\bar{y}_{1:k})$. From [48] we have the following result,

$$I(\zeta_{k+1}|\bar{y}_{1:k}) = D^{22}_k - D^{21}_k\left[D^{11}_k + I_A(\zeta_k|\bar{y}_{1:k})\right]^{-1} D^{12}_k, \tag{4.45}$$

where

$$\begin{aligned}
D^{11}_k &= \mathbb{E}_{f^c_{k+1}}\left[-\triangle^{\zeta_k}_{\zeta_k}\log f(\zeta_{k+1}|\zeta_k,\bar{y}_{1:k})\right],\\
D^{12}_k &= \mathbb{E}_{f^c_{k+1}}\left[-\triangle^{\zeta_{k+1}}_{\zeta_k}\log f(\zeta_{k+1}|\zeta_k,\bar{y}_{1:k})\right] = (D^{21}_k)^T,\\
D^{22}_k &= \mathbb{E}_{f^c_{k+1}}\left[-\triangle^{\zeta_{k+1}}_{\zeta_{k+1}}\log f(\zeta_{k+1}|\zeta_k,\bar{y}_{1:k})f(\bar{y}_{k+1}|\zeta_{k+1})\right].
\end{aligned}$$

In addition,

$$I_A(\zeta_k|\bar{y}_{1:k}) = E^{22}_k - E^{21}_k\left(E^{11}_k\right)^{-1} E^{12}_k, \tag{4.46}$$

where

$$\begin{aligned}
E^{11}_k &= \mathbb{E}_{f(\zeta_{0:k}|\bar{y}_{1:k})}\left[-\triangle^{\zeta_{0:k-1}}_{\zeta_{0:k-1}}\log f(\zeta_{0:k}|\bar{y}_{1:k})\right],\\
E^{12}_k &= \mathbb{E}_{f(\zeta_{0:k}|\bar{y}_{1:k})}\left[-\triangle^{\zeta_k}_{\zeta_{0:k-1}}\log f(\zeta_{0:k}|\bar{y}_{1:k})\right] = (E^{21}_k)^T,\\
E^{22}_k &= \mathbb{E}_{f(\zeta_{0:k}|\bar{y}_{1:k})}\left[-\triangle^{\zeta_k}_{\zeta_k}\log f(\zeta_{0:k}|\bar{y}_{1:k})\right].
\end{aligned}$$

We observe that it is still difficult to obtain matrices $E_k^{11}, E_k^{12}, E_k^{21}, E_k^{22}$ so [48] introduces the following approximate recursion

$$I_A(\zeta_k|\bar{y}_{1:k}) \approx S_k^{22} - S_k^{12\,T}\left[S_k^{11} + I_A(\zeta_{k-1}|\bar{y}_{1:k-1})\right]^{-1} S_k^{12}, \tag{4.47}$$

where

$$\begin{aligned}
S_k^{11} &= \mathbb{E}_{f(\zeta_{0:k}|\bar{y}_{1:k})}\left[-\triangle_{\zeta_{k-1}}^{\zeta_{k-1}} \log f(\zeta_k|\zeta_{k-1}, \bar{y}_{1:k-1})\right], \\
S_k^{12} &= \mathbb{E}_{f(\zeta_{0:k}|\bar{y}_{1:k})}\left[-\triangle_{\zeta_{k-1}}^{\zeta_k} \log f(\zeta_k|\zeta_{k-1}, \bar{y}_{1:k-1})\right], \\
S_k^{22} &= \mathbb{E}_{f(\zeta_{0:k}|\bar{y}_{1:k})}\left[-\triangle_{\zeta_k}^{\zeta_k} \log f(\zeta_k|\zeta_{k-1}, \bar{y}_{1:k-1}) f(\bar{y}_k|\zeta_k)\right].
\end{aligned}$$

We observe that in practice it may still be difficult to compute the exact expectations because high dimensional integration is generally involved. Nonetheless, particle filters as described in [51] can be used to approximate these expectations. Alternative approximations for the conditional posterior Cramer-Rao lower bound can be found in [52]. Unconditional bounds can be found in [53].

The algorithm above enables the defender to compute an approximate lower bound on the mean square error matrix of the attacker's state ζ_k for a given set of inputs $u_{0:k}^a, \bar{d}_{1:k}^a$ and observation history $\bar{y}_{1:k}$. This allows us to obtain a lower bound on the expected value of the squared 2-norm of our residue z_k (defined in (4.16)). As we have seen, the residue is a common statistic used to characterize the health of a system. Under normal operation, there exists good agreement between measured and expected behavior so the residue is expected to be small. We are able to characterize how small an attacker is able to make a residue given his information.

Theorem 4.2. *Consider the special case that $\{C_j\}$ is known to the adversary for all $j \in \mathbb{Z}$. Suppose an attacker attempts to estimate $\zeta_k = \{\bar{x}_k, \hat{\bar{x}}_k, \mathcal{P}_k\}$ as in attack strategy 2. Let $\hat{\bar{x}}_k^e(\bar{y}_k)$ be an estimate of $\hat{\bar{x}}_k$ as a function of $\bar{y}_k$ given $\bar{y}_{1:k-1}$ and $\hat{e}_k = \hat{\bar{x}}_k - \hat{\bar{x}}_k^e(\bar{y}_k)$. Suppose a lower bound Z_k on the error matrix of $\hat{\bar{x}}_k$ is obtained so that*

$$\mathbb{E}_{f_k^c}\left[\hat{e}_k \hat{e}_k^T\right] \geq Z_k. \tag{4.48}$$

Then we have

$$\min_{\bar{y}_k^a} \mathbb{E}_{f^*}\left[z_k^T z_k\right] \geq \operatorname{tr}(\mathcal{C}_k Z_k \mathcal{C}_k^T), \tag{4.49}$$

where $f^* = f(\hat{\bar{x}}_k, \bar{y}_k | \mathcal{I}_{k-1}^a, u_{k-1}^a, \bar{d}_{k-1}^a, u_{k-1})$.

Proof. First, observe from Remark 4.5

$$f(\zeta_{0:k}, \bar{y}_k | \mathcal{I}_{k-1}^a, u_{k-1}^a, \bar{d}_{k-1}^a, u_{k-1}) = f_k^c. \tag{4.50}$$

We now have the following.

$$\begin{aligned}
&\min_{\bar{y}_k^a} \mathbb{E}_{f^*}\left[z_k^T z_k\right] \\
&= \min_{\bar{y}_k^a} \mathbb{E}_{f^*}\left[\operatorname{tr}\left((\bar{y}_k^a - \mathcal{C}_k\hat{\bar{x}}_k)(\bar{y}_k^a - \mathcal{C}_k\hat{\bar{x}}_k)^T\right)\right], \\
&= \min_{\bar{y}_k^a} \operatorname{tr}\left(\mathbb{E}_{f^*}\left[(\bar{y}_k^a - \mathcal{C}_k\hat{\bar{x}}_k)(\bar{y}_k^a - \mathcal{C}_k\hat{\bar{x}}_k)^T\right]\right), \\
&= \operatorname{tr}\left(\min_{\bar{y}_k^a} \left(\mathbb{E}_{f^*}\left[(\bar{y}_k^a - \mathcal{C}_k\hat{\bar{x}}_k)(\bar{y}_k^a - \mathcal{C}_k\hat{\bar{x}}_k)^T\right]\right)\right), \\
&= \operatorname{tr}\left(\mathcal{C}_k \min_{\hat{\bar{x}}_k^e} \left(\mathbb{E}_{f^*}\left[(\hat{\bar{x}}_k^e - \hat{\bar{x}}_k)(\hat{\bar{x}}_k^e - \hat{\bar{x}}_k)^T\right]\right)\mathcal{C}_k^T\right), \\
&= \operatorname{tr}\left(\mathcal{C}_k \min_{\hat{\bar{x}}_k^e} \left(\mathbb{E}_{f_k^c}\left[(\hat{\bar{x}}_k^e - \hat{\bar{x}}_k)(\hat{\bar{x}}_k^e - \hat{\bar{x}}_k)^T\right]\right)\mathcal{C}_k^T\right), \\
&\geq \operatorname{tr}(\mathcal{C}_k Z_k \mathcal{C}_k^T).
\end{aligned} \tag{4.51}$$

The first two equalities follow from properties of the trace and expectation. The third equality follows from monotonicity properties of the trace function. The fourth equality is based on the fact that given $\mathcal{C}_k$, a minimizer lies in the range space of $\mathcal{C}_k$. The fifth equality is due to (4.50). The final inequality follows from (4.48). □

Remark 4.6. *In general, the adversary's ability to estimate* $\{\zeta_k\}$ *is dependent on the inputs* $\{u_k^a\}, \{\bar{d}_k^a\}$. *For instance, the more the adversary biases the state away from its expected region of operation, the more challenging it is to perform estimation. Thus, if the system operator wishes to analyze how well an adversary can generate stealthy outputs, he must consider a particular sequence of attack inputs* $u_k^a, \bar{d}_k^a$.

Remark 4.7. *In practice, it may be difficult to perform performance analysis when assuming $\mathcal{P}_k$ is an unknown state. However, one can still approximate a lower bound on the error matrix by assuming that the adversary has an oracle which allows him to know $\mathcal{P}_k$, $\mathcal{K}_k$, $I - \mathcal{K}_k\mathcal{C}_k$.*

Remark 4.8. *The design of the authentication subsystem matrices is not considered in this thesis and left for future work. However, we expect that increasing the covariance of the random matrices will make the problem of system identification more difficult for the attacker, thereby increasing the lower bound. Similar to watermarking, there likely exists an optimal direction for our perturbations that maximize performance.*

4.1.5 Numerical Example

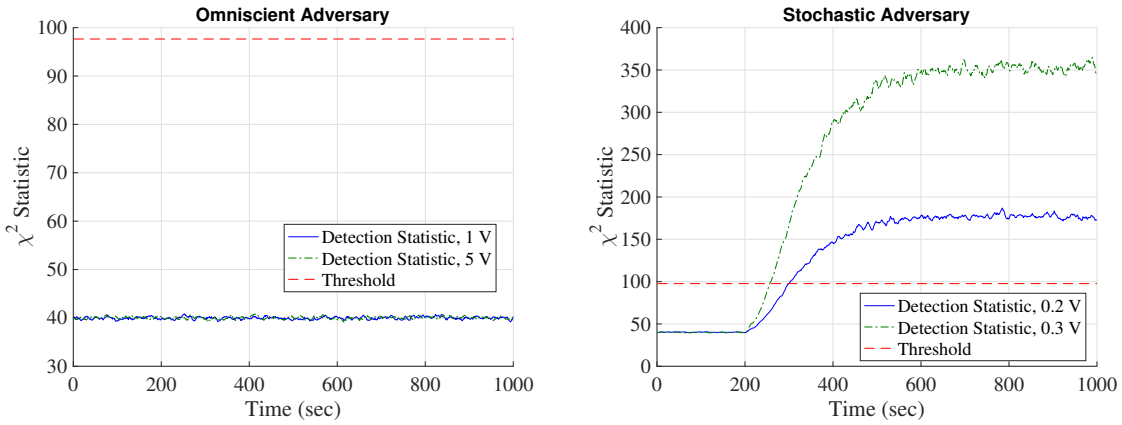
As in section 3.1, we test the moving target on the quadruple tank process, a four state system [39]. The goal is to control the water level of two of four tanks using two pumps. Two sensors measure water heights. We use an LQG controller with weights following suggestions in [40]. Q and R are created by generating a matrix from a uniform distribution, multiplying it by its transpose, and dividing by 100.

4 extra states and 2 extra outputs are added. The time varying matrices $A_{1,k}, A_{2,k}, B_k, C_{k+1}$ are somewhat sparse (50% of entries nonzero). The non-zero elements follow a multivariate Gaussian distribution with means generated from $U(-0.5, 0.5)$. The covariances of the nonzero parameters are created by generating a matrix from a uniform distribution, multiplying it by its transpose, and dividing by 100.

We consider an adversary who, starting at time 200 sec, adds a constant input (in Volts) to the optimal LQG input and avoids detection by trying to subtract his own influence from the measurements. First, in Figs. 4.3(a), 4.4(a), we assume the attacker knows the time varying system matrices. Secondly, we assume the attacker does not know the realization of $A_{1,k}, A_{2,k}, B_k, C_{k+1}$, but instead performs his attack by sampling the matrices from the appropriate distribution, (Figs.

4.3(b), 4.4(b)). Note, that any sort of optimal attack as described in this section is in practice infeasible due to the required numerical and computational complexity.

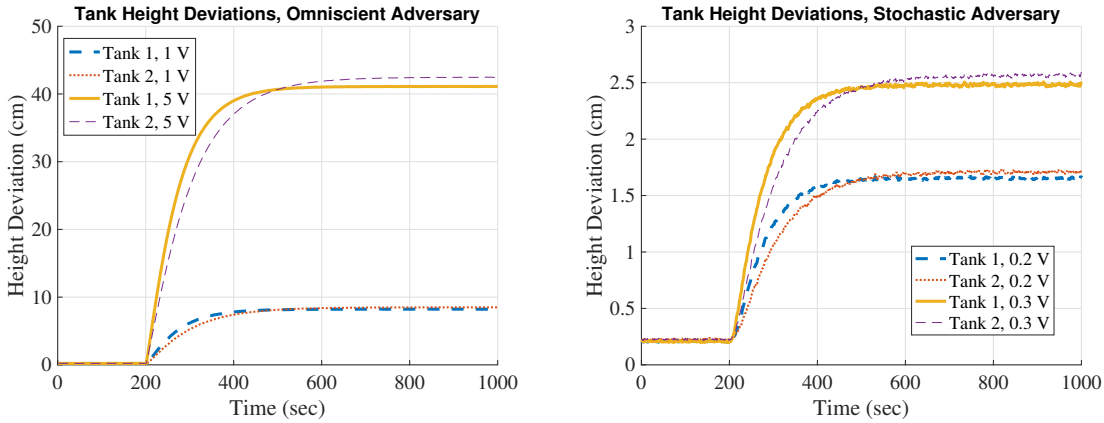
We plot a χ^2 detector statistic (window 10, $\alpha = 10^{-7}$) in Fig. 4.3(a) and 4.3(b) and system performance in Fig. 4.4(a) and 4.4(b), both averaged over 1000 trials. The asymptotic probability of detection vs false alarm is found in Fig. 4.5. Here α corresponds to the probability of false alarm, which is constant in time due to the stationary behavior of the state. Given full knowledge of the system matrices, the attacker can significantly affect water levels while remaining perfectly stealthy. However, with stochastic knowledge of the system matrices, the attack is easily revealed, even for small system perturbations and small α. In practice, the attack can be improved by using the measurements $\tilde{y}_k$ to perform system identification. We expect improvements to be marginal since the system changes at each time step. Thus, it is important to analyze the effectiveness of an attacker who performs machine learning in a scenario where the moving target changes at a lower frequency.



(a) Full Knowledge: Det. Stat vs Time

(b) Stochastic knowledge: Det. Stat vs Time

Figure 4.3: Detection Statistic of Moving Target: Quadruple Tank



(a) Height deviation (cm): Full Knowledge (b) Height deviation (cm): Stochastic Knowledge

Figure 4.4: Mean absolute height deviation (cm): Quadruple Tank with Moving Target

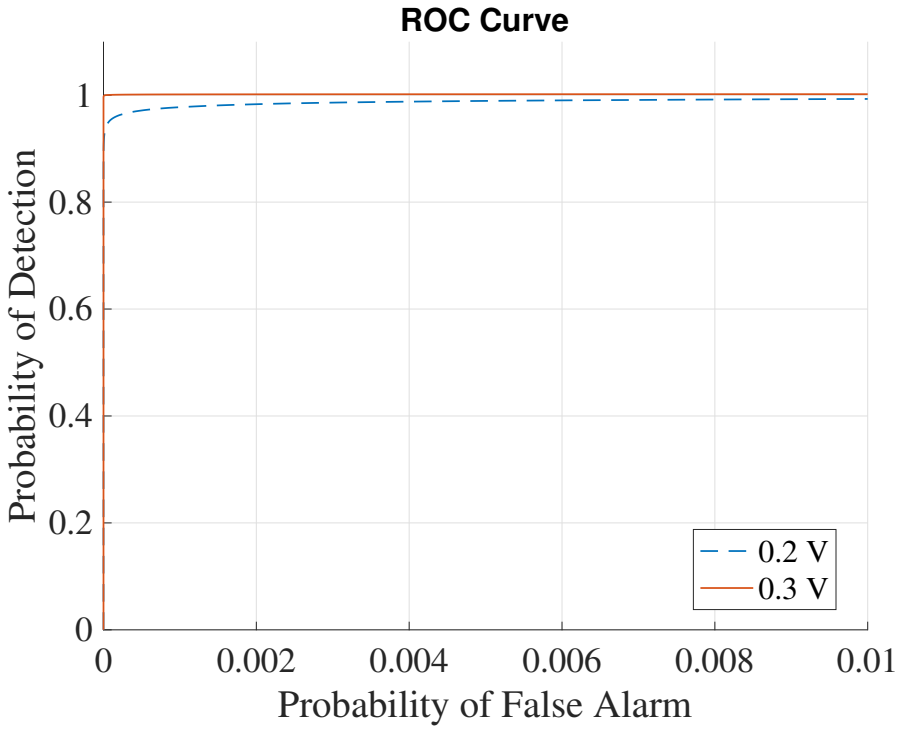


Figure 4.5: ROC Curve, Moving Target: Stochastic Attack

4.2 The Hybrid System Approach

In this section, we consider an alternative moving target design, which we refer to as the hybrid system approach. Here, instead of introducing an additional authenticating subsystem to a cyber-

physical system that we do not care about, we perform active detection by changing the parameters of the true plant itself in a discrete fashion. We will demonstrate how this technique can aid us not only in the detection of malicious adversaries, but also the problem of identification. We will focus in particular on sensor attacks.

4.2.1 System Description

To begin, we model our control system as a discrete deterministic system. We will consider the stochastic case later in the section. The dynamics are given by

$$x_{k+1} = Ax_k + B(u_k(y_{0:k})), \;\; y_k = Cx_k + D^a d_k^a. \tag{4.52}$$

where $x_k \in \mathbb{R}^n$ is the state at time k, $u_k(y_{0:k}) \in \mathbb{R}^p$ is the control input, and $y_k \in \mathbb{R}^m$ are the sensor outputs. The sensor outputs, y_k, consist of m scalar sensor outputs, defined by the set $S = \{1, 2, \cdots, m\}$. We assume for now that (A, C) is observable.

The adversary performs an attack on an ordered set of sensors $K = \{s_1, s_2, \cdots, s_{|K|}\} \subseteq S$ using additive inputs $d_k^a \in \mathbb{R}^{|K|}$, starting at time $k = 0$. Here, a denotes that the input is an attack input. Consequently, we define $D^a \in \mathbb{R}^{m \times |K|}$ entrywise as

$$D^a_{uv}(K) = \mathbb{I}_{u=s_i, v=i}, \tag{4.53}$$

where $\mathbb{I}$ is the indicator function. Note that D^a is fully determined by the set K. Implicitly, we assume that the set of sensors which the adversary targets is constant due to (ideally) the inherent difficulty in the task of hijacking sensors. When performing an integrity attack, the adversary's goal is to adversely affect the physical system by preventing proper feedback. In particular, a defender with incorrect sensor measurements may not be able to perform adequate state estimation and thus will not be able to apply appropriate corrective measures to the system.

We assume that the defender knows the system dynamics $\mathcal{M} = \{A, B, C\}$ as well as the input and output histories given by $u_{0:k-1}$ and $y_{0:k}$, but is unaware of the set K. Furthermore, we assume that in the deterministic setup, the defender is unaware of the initial state x_0. Thus, the information

$\mathcal{I}_k$ available to the defender at time k is given by

$$\mathcal{I}_k = \{\mathcal{M}, u_{0:k-1}, y_{0:k}\}. \tag{4.54}$$

Remark 4.9. *In the deterministic case, we explore attacks where the defender has no knowledge of the initial state. While this is certainly not realistic, the attack vectors developed in this scenario can still remain stealthy in a practical stochastic setting if the adversary carefully ensures that his initial attack inputs remain hidden by the noise of the system.*

From a defender's perspective it is important to identify trusted sensor nodes. Estimation and control algorithms can then be tuned to ignore attacked nodes. We note that the problem of identifying malicious nodes is independent of the control input, since the defender is aware of the model and input history. Thus, in the ensuing discussions we will disregard the control input so that

$$x_{k+1} = Ax_k, \;\; y_k = Cx_k + D^a d_k^a. \tag{4.55}$$

Prior Results on Attack Identification

In this section, beyond looking at the problem of detection, we focus on how active techniques can enable attack isolation or identification. The ability to isolate the actions of malicious defenders aids a defender in his endeavor to provide resilient countermeasures. For instance, identifying malicious sensors could allow a defender to design resilient feedback control laws which bypass these sensors and are still able to meet system specifications. Similar to [44], we define a notion of attack identification in control systems.

Definition 4.1. *An attack input $\{D^a(K)d_k^a\}$ on a deterministic system with unknown state x_0 is unidentifiable if and only if*

1. *there exists sets $K' \subset S$ with $K' \neq K$*
2. $|K'| \leq |K|$

3. there exists $x_0' \in \mathbb{R}^n$ and inputs $\{\bar{d}_k^a\}$ satisfying

$$y(x_0, D^a(K)d_k^a, k) = y(x_0', D^a(K')\bar{d}_k^a, k), \quad \forall\, k \geq 0. \tag{4.56}$$

We assume every sensor in K is attacked at least once given input $\{D^a(K)d_k^a\}$ and every sensor in K' is attacked at least once given input $\{D^a(K')\bar{d}_k^a\}$

Here, $y(x_0, D^a(K)d_k^a, k)$ is the output y_k due to the initial state x_0 and the sequence of attacks $\{D^a(K)d_0^a, \cdots, D^a(K)d_k^a\}$. An attack is unidentifiable if there exists an alternative attack on a smaller set of sensors that achieves the same output. Note, here we must make the restriction that $|K'| \leq |K|$. This is due to the fact that there always exists an attack on the complement of K which can generate the same outputs as an attack on K'.

In particular, suppose $D^a(K)d_k^a = \Psi(K)CA^k\Delta x_0$, where we define $\Psi(K) \in \mathbb{R}^{m\times m}$ entrywise as

$$\Psi(K)(i,j) = \mathbb{I}_{i=j, i\in K} \tag{4.57}$$

We then have

$$y(x_0, D^a(K)d_k^a, k) = y(x_0 + \Delta x_0, D^a(K^c)\bar{d}_k^a, k), \quad \forall\, k \geq 0. \tag{4.58}$$

where $D^a(K^c)\bar{d}_k^a = -\Psi(K^c)CA^k\Delta x_0$. Here, $K^c = S - K$. More generally, we can equate the existence of an unidentifiable attack to sparse observability. A similar result is obtained in [54]. The proof is included here for completeness.

Definition 4.2. *A system (A, C) is $s-$sparse observable if and only if one can remove any s rows from the matrix C and the resulting system remains observable.*

Theorem 4.3. *Suppose (A, C) is observable. There exists an unidentifiable attack on q or fewer sensors if and only if (A, C) is $2q$ sparse observable.*

Proof. Suppose (A, C) is not $2q$ sparse observable. Suppose $m > 2q$. Choose sets $K_1, K_2, K_3 \subseteq S$ such that $K_1 \cap K_2 = \emptyset$, $\max(|K_1|, |K_2|) \leq q$, $|K_1| + |K_2| \leq 2q$, $K_3 = S - K_2 - K_1$ and (A, C^{K_3})

is not observable. Here, C^X are the rows of C indexed by X. Let x_0^1 be arbitrary. Let $x_0^2 \neq 0$ belong to the unobservable subspace of (A, C^{K_3}).

For all $k \geq 0$, let $D^a(K_1)d_k^a = \Psi(K_1)CA^k x_0^2$ and $D^a(K_2)\bar{d}_k^a = -\Psi(K_2)CA^k x_0^2$. Without loss of generality, assume injecting $D^a(K_1)d_k^a = \Psi(K_1)CA^k x_0^2$ requires attacking a set $K_1' \subseteq K_1$ where every sensor in K_1' is attacked at least once, and suppose $D^a(K_1)d_k^a = D^a(K_1')d_k^{a'}$. Moreover assume without loss of generality, injecting $D^a(K_2)\bar{d}_k^a = -\Psi(K_2)CA^k x_0^2$ requires attacking a set $K_2' \subseteq K_2$ where every sensor in K_2' is attacked at least once, and suppose $D^a(K_2)\bar{d}_k^a = D^a(K_2')\bar{d}_k^{a'}$. Finally, without loss of generality assume $|K_1'| \geq |K_2'|$. Since, (A, C) is observable, $|K_1'| > 0$. Moreover, the number of sensors attacked is less than or equal to q. Observe, for all $k \geq 0$

$$C^{K_3}A^k x_0^1 = C^{K_3}A^k(x_0^1 + x_0^2).$$

Thus, $y(x_0^1, D^a(K_1')d_k^{a'}, k) = y(x_0^1 + x_0^2, D^a(K_2')\bar{d}_k^{a'}, k)$ for all $k \geq 0$, where $|K_2'| \leq |K_1'|$ and $K_2' \neq K_1'$. Thus, there is an unidentifiable attack.

If $m \leq 2q$, take K_1 and K_2 such that $K_1 \cup K_2 = S$, $K_1 \cap K_2 = \emptyset$, and $\max(|K_1|, |K_2|) \leq q$. Select arbitrary x_0^1 and $x_0^2 \neq 0$. For all $k \geq 0$, let $D^a(K_1)d_k^a = \Psi(K_1)CA^k x_0^2$ and $D^a(K_2)\bar{d}_k^a = -\Psi(K_2)CA^k x_0^2$. Without loss of generality, assume injecting $D^a(K_1)d_k^a = \Psi(K_1)CA^k x_0^2$ requires attacking a set $K_1' \subseteq K_1$ where every sensor in K_1' is attacked at least once, and suppose $D^a(K_1)d_k^a = D^a(K_1')d_k^{a'}$. Moreover assume without loss of generality, injecting $D^a(K_2)\bar{d}_k^a = -\Psi(K_2)CA^k x_0^2$ requires attacking a set $K_2' \subseteq K_2$ where every sensor in K_2' is attacked at least once, and suppose $D^a(K_2)\bar{d}_k^a = D^a(K_2')\bar{d}_k^{a'}$. Finally, without loss of generality assume $|K_1'| \geq |K_2'|$. Since, (A, C) is observable, $|K_1'| > 0$. Moreover, the number of sensors attacked is less than or equal to q. Again,$y(x_0^1, D^a(K_1')d_k^{a'}, k) = y(x_0^1 + x_0^2, D^a(K_2')\bar{d}_k^{a'}, k)$ for all $k \geq 0$, where $|K_2'| \leq |K_1'|$ and $K_2' \neq K_1'$. Thus, there is an unidentifiable attack.

Now suppose there exists a nonzero unidentifiable attack on a set of K, where $|K| \leq q$. Then there exists set K' and inputs d_k^a and $\bar{d}_k^a$ such that for some x_0 and x_0', (4.56) holds. Where $|K'| \leq |K|$ and $K' \neq K$. If $x_0 = x_0'$, then $D^a(K)d_k^a = D^a(K')\bar{d}_k^a$, for all $k \geq 0$, which contradicts $K \neq K'$. Thus, without loss of generality assume $x_0 \neq x_0'$. Then, by linearity, there exists set K^*

with $|K^*| \leq 2q$ and input $\tilde{d}_k^a$ such that

$$y(x_0 - x_0', D^a(K^*)\tilde{d}_k^a, k) = 0, \quad \forall\, k \geq 0$$

However, this implies that $x_0^1 - x_0^2 \neq 0$ lies in the unobservable subspace of (A, C^{S-K^*}). Note that $|S - K^*| \geq m - 2q$. As such, (A, C) is not $2q$-sparse observable. □

We see here that in order to identify q attacks, one must be able to perform state estimate even when $2q$ sensors are removed. This is because the defender must have enough redundant information to distinguish $m - q$ trusted sensors from q malicious sensors. In a system with $2q$ sensors, a defender would be unable to determine in general which sensors are trustworthy and which sensors are malicious. As such additional sensors are necessary to ensure identifiability. In particular, we need enough additional sensors to guarantee $2q$ sparse observability.

4.2.2 A Hybrid System Moving Target

In the previous section we demonstrated that there exist limitations on the number of attacks a defender can potentially identify. Specifically, we saw that to identify all attacks of size q, the system must be $2q$ sparse observable. This can result in expenditures to add more sensing in order to withstand more attacks or sacrificing security in order to use fewer components. However, in this section we argue that generating unidentifiable attacks requires knowledge of the model. By limiting this knowledge, we hope to prevent such attacks. To begin we define the following.

Definition 4.3. *A nonzero attack on sensor s is unambiguously identifiable at time t if there is no $x_0^* \in \mathbb{R}^n$ satisfying $y_k^s = y^s(x_0^*, 0, k)$ for $0 \leq k \leq t$. An attack on sensor s is unambiguously identifiable if it is unambiguously identifiable for all t.*

The notion of unambiguous identifiability characterizes when the defender can be certain that sensor s is faulty or under attack. This scenario occurs only if there exists no initial state which produces the output sequence at y^s. We envision designing a system that forces the attacker to

generate unambiguously identifiable attacks on all sensors which he targets. Consequently, we can identify misbehaving sensors.

Thus, instead of requiring our system to be $2q$ sparse observable to perform perfect estimation with q attacks or $2q$ detectable to perform stable estimation [55], forcing an attacker to generate unambiguously identifiable attacks will allow the defender to perform stable estimation when the system is only q detectable (detectable after removing any q sensors). This allows the system to withstand more powerful attacks or use fewer sensing devices while maintaining the same level of security. We now characterize attacks which are not unambiguously identifiable. For notational simplicity let the sth row of D^a be denoted as D^s.

Theorem 4.4. *An attack on sensor s is not unambiguously identifiable at time t if and only if there exists an x_0^* such that $D^s d_k^a = C^s A^k x_0^*$ for all $0 \leq k \leq t$ and $C^s A^k x_0^* \neq 0$ for some time $0 \leq k \leq t$.*

Proof. Suppose $D^s d_k^a = C^s A^k x_0^*$ for time $0 \leq k \leq t$. Assume this attack is nonzero. Then, $y_k^s = y^s(x_0 + x_0^*, 0, k)$. Suppose instead that there is no x_0^* such that $D^s d_k^a \neq C^s A^k x_0^*$ for $0 \leq k \leq t$. Then there is no $\bar{x}_0$ such that $C^s A^k x_0 + D^s d_k^a = C^s A^k \bar{x}_0$. The result immediately follows. □

As a result, to prevent attacks on sensor s from being unambiguously identifiable at time k, an adversary must insert attacks which lie in the image of $\mathcal{O}_{k+1}^s$ given by

$$\mathcal{O}_{k+1}^s = \begin{bmatrix} (C^s)^T & (C^s A)^T & \cdots & (C^s A^k)^T \end{bmatrix}^T. \tag{4.59}$$

To insert such attacks, the adversary likely has to be aware of both the matrix A and the matrix C^s. In the sequel, we aim to minimize this knowledge to prevent an attacker from generating unidentifiable attacks.

Ideally, we would like to simply assume the adversary has no knowledge of (A, C) and consequently will likely always be unambiguously identifiable. However, in practice, the processes associated with the physical plant may be well known or previously public so that the attacker is aware of (A, C). Alternatively, the defender can change parameters of the system to ensure a

knowledgeable adversary is still thwarted. Specifically, we propose changing the system matrix A and C in a time varying and unpredictable fashion from the adversary's point of view so that

$$x_{k+1} = A_k x_k, \quad y_k = C_k x_k + D^a d_k^a. \tag{4.60}$$

We assume that $(A_k, C_k) \in \Gamma = \{(A(1), C(1)), \cdots, (A(l), C(l))\}$. The system matrices are changed in a discrete fashion, resembling a hybrid system. We henceforth refer to this design as the hybrid system moving target approach. For this time varying system, we can similarly characterize the set of unambiguously identifiable attacks.

Theorem 4.5. *An attack on sensor s in (4.60) is not unambiguously identifiable at time t if and only if there exists an x_0^* such that $D^s d_k^a = C_k^s(\prod_{j=0}^{k-1} A_j)x_0^*$ for all time $0 \leq k \leq t$ and $C_k^s(\prod_{j=0}^{k-1} A_j)x_0^* \neq 0$ for some time $0 \leq k \leq t$.*

Proof. The proof is similar to that of Theorem 4.4. □

Changing the system matrices as a function of time allows the system to act like a moving target. In particular, even if an attacker is aware of the existing configurations of the system, defined by Γ, he will likely be forced to generate unambiguously identifiable attacks since he is not aware of the sequence of system matrices. Moreover, since the system matrices keep changing, it is unlikely the attacker can remain unidentifiable by pure chance.

Remark 4.10. *The matrices (A_k, C_k) can be changed randomly using a cryptographically secure pseudo random number generator where the random seed is known both by the defender and the plant, but is unavailable to the adversary. From a security perspective, the seed would form the root of trust. The set Γ can be obtained by leveraging or introducing degrees of freedom in the dynamics and sensing in our control system. While the defender likely would have to change his control strategy to account for the time varying dynamics, we will leave the analysis of such strategies for future work.*

Remark 4.11. *Compared to the authenticating subsystem approach, the hybrid system approach has the advantage that it may not need to introduce external dynamics if there exists means to*

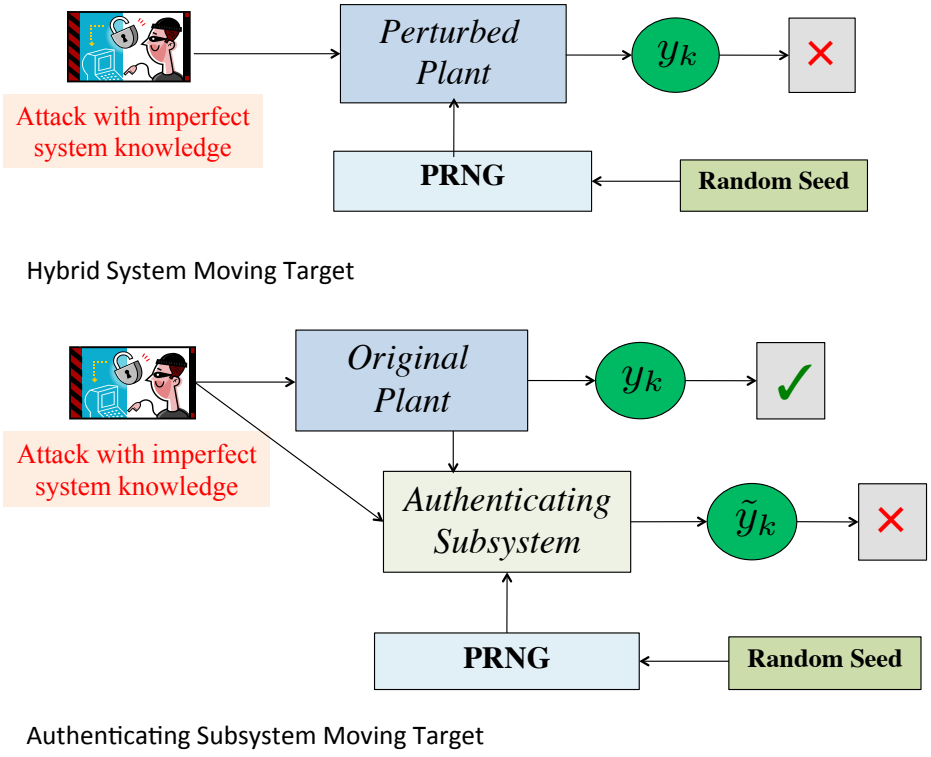


Figure 4.6: A Comparison of the underlying structure of the Hybrid and Authenticating Subsystem Moving Targets

switch parameters in the plant. The disadvantage as mentioned above is that one must consider system performance and control when switching dynamics of the original plant. Fig. 4.6 provides a high level look at the underlying structure of both moving target strategies. This thesis focuses on necessary design recommendations for the time varying dynamics, which do not take into account constraints in control. Investigating fundamental tradeoffs is left for future work.

Given the proposed setup, we are now ready to define the attacker's information and an admissible attacker strategy.

Attacker Information

1. The adversary has no knowledge of either the input sequence $u_{0:k}$ or the true output sequence.

2. $\mathcal{I}_k^a = \{\Gamma, D^a d_{0:k-1}^a, f(\{(A_k, C_k)\})\}$.

If the adversary can observe the output sequence in a zero input deterministic setting, he can multiply the true outputs by some constant factor to avoid generating unambiguously identifiable inputs. The control inputs are also secret so that the attacker will be unable to leverage the input process to gain information about the system model. A realistic adversary may use physical attacks to bias sensors without reading their outputs. Another possible scenario with such a adversary can occur if an attacker uses public key encryption, but the encryption is homomorphic with respect to addition. In this case, the attacker would not be able to gauge the true measurement y_k^s from its ciphertext. However, knowing the ciphertext would allow the attacker to compute the encrypted version of $y_k^s + D^s d_k^a$. Future work will examine relaxing restrictions on the attacker's information.

We assume Γ is known as well as the sequence of attack inputs. Also, the probability distribution of the sequence of system matrices, $f(\{(A_k, C_k)\})$, is public.

Definition 4.4. *An admissible attack policy is a sequence of deterministic mappings* $\Omega_k : \mathcal{I}_k^a \rightarrow$ *Im*$(D^a(K))$ *such that* $D^a d_k^a = \Omega_k(\mathcal{I}_k^a)$.

Here, we assume the attacker can only leverage his information to construct a stealthy attack input. Consequently, while there may exist attacks that bypass identification, in order to be admissible, they must leverage the attacker's knowledge and can not be a function of unknown and unobserved stochastic processes (namely the sequence of $\{A_k\}$ and $\{C_k\}$). A real adversarial strategy may be to bias sensors with the goal of affecting state estimation, without being identified by the defender. Thus, the adversary can impact the system without corrective measures being put in place.

4.2.3 System Design for Deterministic Identification

We now consider criteria that can allow a defender to design an effective set Γ. Given the attacker's knowledge of Γ, an adversary can guess the sequence of system matrices chosen by the defender. If

the adversary guesses correctly, he can generate attacks which are not unambiguously identifiable. We would now like to characterize the scenario where an attacker can guess the sequence of matrices incorrectly yet still generate an unambiguously identifiable attack.

Theorem 4.6. *Suppose an adversary generates an attack on sensor s by guessing a sequence $\{l_k\}$ where $l_i \in \{1, \cdots, l\}$ and creating inputs by applying Theorem 4.5. Specifically, there exists an x_0^1 such that $D^s d_k^a = C^s(l_k)(\prod_{j=0}^{k-1} A(l_j))x_0^1$ for all time $0 \leq k \leq t$ and $D^s d_\eta^a \neq 0$ for some time $0 \leq \eta \leq t$. Such a strategy may avoid generating an unambiguously identifiable attack on sensor s at time t if and only if*

$$\textit{null}\begin{pmatrix}\mathcal{O}(l_s,t) & \mathcal{O}(s,t)\end{pmatrix} > \textit{null}\begin{pmatrix}\mathcal{O}(l_s,t)\end{pmatrix} + \textit{null}\begin{pmatrix}\mathcal{O}(s,t)\end{pmatrix}, \tag{4.61}$$

$$\mathcal{O}(l_s,t) = \begin{bmatrix}(C^s(l_0))^T & (C^s(l_1)A(l_0))^T & \cdots & (C^i(l_t)\prod_{j=0}^{t-1} A(l_j))^T\end{bmatrix}^T,$$
$$\mathcal{O}(i,t) = \begin{bmatrix}(C_0^s)^T & (C_1^s A_0)^T & \cdots & (C_t^s \prod_{j=0}^{t-1} A_j)^T\end{bmatrix}^T,$$

where null refers to the dimension of the null space.

Proof. From Theorem 4.5, an attack is not unambiguously identifiable at time t if and only if there exists some x_0^2 such that $D^s d_k^a = C_k^s(\prod_{j=0}^{k-1} A_j)x_0^2$ for $0 \leq k \leq t$ and this sequence is nonzero. Thus, the proposed strategy can generate a nonzero unambiguously identifiable attack on sensor s at time t if and only if

$$C^s(l_k)\left(\prod_{j=0}^{k-1} A(l_j)\right)x_0^1 = C_k^s\left(\prod_{j=0}^{k-1} A_j\right)x_0^2,$$

for all $0 \leq k \leq t$ and moreover for some $0 \leq k \leq t$ this expression is nonzero. The result immediately follows. □

It is undesirable to change the parameters of the system at each time step due to the system's inertia. Consequently, we would like to consider systems where (A_k, C_k) remains constant for longer periods of time. For now, we assume $(A_k, C_k) \subset \{\Gamma\}$, but is *constant*. An adversary, can

use his knowledge of Γ to guess a pair $(A_k, C_k) \in \Gamma$ and generate unidentifiable attack inputs. Define the matrix

$$\mathcal{O}^{\mathcal{S}}_{t,j} = \begin{bmatrix} C^{\mathcal{S}}(j)^T & (C^{\mathcal{S}}(j)A(j))^T & \cdots & (C^{\mathcal{S}}(j)A(j)^{t-1})^T \end{bmatrix}^T. \tag{4.62}$$

If the attacker guesses the matrices $(A(j), C(j))$ and chooses to attack sensor s, he would need to ensure $\begin{bmatrix} (D^s d^a_0)^T & \cdots & (D^s d^a_t)^T \end{bmatrix}^T$ lies in the image of $\mathcal{O}^s_{t+1,j}$ to avoid deterministic identification. We next determine when an attacker is able to guess an incorrect pair and avoid generating an unambiguously identifiable attack.

Theorem 4.7. *Suppose $(A, C) = (A(1), C(1))$ and an adversary generates a nonzero attack input on sensor s using $(A(2), C(2))$ by inserting attacks along the image of $\mathcal{O}^s_{t,2}$. Let $\Lambda^1 = \{\lambda^1_1, \cdots, \lambda^1_{q_1}\}$ be the set of distinct eivenvalues associated with $A(1)$ and $\Lambda^2 = \{\lambda^2_1, \cdots, \lambda^2_{q_2}\}$ be the set of distinct eigenvalues of $A(2)$. Let*

$$\{v^{\lambda,j}_{1,1}, \cdots v^{\lambda,j}_{r_1,1}, v^{\lambda,j}_{1,2}, \cdots v^{\lambda,j}_{r_2,2}, \cdots, v^{\lambda,j}_{1,l_{\lambda,j}}, \cdots v^{i,j}_{r_{l_{\lambda,j}},l_{\lambda,j}}\}$$

be a maximal set of linearly independent (generalized) eigenvectors associated with eigenvalue λ of $A(j)$ satisfying

$$A(j)v^{\lambda,j}_{1,l} = \lambda v^{\lambda,j}_{1,l}, \quad A(j)v^{\lambda,j}_{k+1,l} = \lambda v^{\lambda,j}_{k+1,l} + v^{\lambda,j}_{k,l}. \tag{4.63}$$

Noting that each r_i is in general fully determined by λ and j, let $r(\lambda) = \max_{i,j} r_i(\lambda, j)$. Define $V^{\lambda,j}_{s,k} \in \mathbb{C}^{r(\lambda)\times r_k}$ as

$$\begin{bmatrix} C^s(j)v^{\lambda,j}_{1,k} & C^s(j)v^{\lambda,j}_{2,k} & C^s(j)v^{\lambda,j}_{3,k} & \cdots & \cdots & C^s(j)v^{\lambda,j}_{r_k-1,k} & C^s(j)v^{\lambda,j}_{r_k,k} \\ \mathbf{0} & C^s(j)v^{\lambda,j}_{1,k} & C^s(j)v^{\lambda,j}_{2,k} & \cdots & \cdots & C^s(j)v^{\lambda,j}_{r_k-2,k} & C^s(j)v^{\lambda,j}_{r_k-1,k} \\ \mathbf{0} & \mathbf{0} & C^s(j)v^{\lambda,j}_{1,k} & \cdots & \cdots & C^s(j)v^{\lambda,j}_{r_k-3,k} & C^s(j)v^{\lambda,j}_{r_k-2,k} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \ddots & \ddots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \cdots & C^s(j)v^{\lambda,j}_{1,k} & C^s(j)v^{\lambda,j}_{2,k} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \cdots & \mathbf{0} & C^s(j)v^{\lambda,j}_{1,k} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \cdots & \mathbf{0} & \mathbf{0} \end{bmatrix}.$$

There exists an attack on sensor s, which is not unambiguously identifiable for all time if and only if $\Lambda^1 \cap \Lambda^2 \neq \emptyset$ and there exist some $\lambda \in \Lambda^1 \cap \Lambda^2$ such that

$$\textit{null}\begin{pmatrix}\mathcal{V}_s^{\lambda,1} & \mathcal{V}_s^{\lambda,2}\end{pmatrix} > \textit{null}\begin{pmatrix}\mathcal{V}_s^{\lambda,1}\end{pmatrix} + \textit{null}\begin{pmatrix}\mathcal{V}_s^{\lambda,2}\end{pmatrix},$$

where

$$\mathcal{V}_s^{\lambda,j} = \begin{pmatrix}V_{s,1}^{\lambda,j} & \cdots & V_{s,l_{\lambda,j}}^{\lambda,j}\end{pmatrix}.$$

Otherwise the attack can be detected in time $t \leq 2n-1$.

Proof. The proof is lengthy and found in the appendix. □

Roughly speaking, given enough observations, the output at sensor s for a time invariant system will be dominated by the observable mode(s) that have the largest eigenvalue. Thus, if the eigenvalues between two system matrices are distinct, we are able to distinguish the resulting outputs. The previous theorem gives the defender an efficient way to determine if the attacker can guess Γ incorrectly yet still remain undetected in the case that system matrices are kept constant for at least a period of $2n$ time steps. It also prescribes a means to perform perfect identification.

Design Recommendations

1. For all pairs $i \neq j \in \{1, \cdots l\}$, $\Lambda^i \cap \Lambda^j = \emptyset$.
2. The system matrices (A_k, C_k) are periodically changed after every $N \geq 2n$ time steps.
3. Let $\{l_k\}$ be a sequence where $l_k \in \{1, \cdots, l\}$. Let q_k denote the indices of a subsequence. $\Pr((A_{q_k}, C_{q_k}) = (A(l_k), C(l_k)),\ \forall k) = 0$.
4. The pair $(A(i), C(i))$ is observable all $i \in \{1, \cdots, l\}$.
5. For all $i \in \{1, \cdots l\}$, $0 \notin \Lambda^i$.

Corollary 4.1. *Assume a defender follows the design recommendations. Suppose sensor s is attacked and there is no t^* such that $D^s d_k^a = 0$ for all $k \geq t^*$. Then, the sensor attack will be unambiguously identifiable with probability 1.*

Proof. Since the attack is persistently nonzero, the adversary must guess a correct infinite subsequence of system matrices due to recommendations 1 and 2. From recommendation 3, this occurs with probability 0. □

As a result, an attacker who persistently biases a sensor will be perfectly identified. Note that recommendation 3 can be achieved with an IID assumption or an aperiodic and irreducible Markov chain. The last 2 recommendations are not needed for this result, but are justified in the next subsection when we consider stochastic systems.

Remark 4.12. *Note, the fact that we would like to keep system matrices constant for a long enough period of time appears counter-intuitive for a moving target. However, the given adversary is not performing system identification and is instead guessing the system matrices. As such, keeping the dynamics constant does not provide useful information for an attacker. Additionally, keeping the matrices constant long enough gives the defender the information he needs to distinguish between the different hybrid states. Similar to the problem of observability, the problem of identification involves a rank deficient matrix until enough measurements have been gathered.*

Before concluding, we would like to provide some intuition into the difficulty of changing the eigenvalues of a matrix. To begin we combine the results of [56][Theorem 7, pg. 130] and [57][Theorem 6.3.12].

Theorem 4.8. *Let $A, E \in \mathbb{R}^{n\times n}$ and suppose λ is a simple eigenvalue of A. Let x and y be right and left eigenvectors of A associated with λ so that $w^T A = \lambda w^T$ and $Av = \lambda v$. Then*

1. *for each given $\epsilon > 0$ there exists a $\delta > 0$ such that, for all $t \in \mathbb{R}$ such that $|t| < \delta$, there is a unique eigenvalue $\lambda(t)$ of $A + tE$, such that $|\lambda(t) - \lambda - tw^T Ev/w^T v| \leq |t|\epsilon$.*
2. *$\lambda(t)$ is continuous at $t = 0$ and $\lim_{t\to 0} \lambda(t) = \lambda$.*
3. *For t small enough, $\lambda(t)$ depends differentiably on t. Moreover,*

$$\frac{d\lambda(t)}{dt}|_{t=0} = \frac{w^T Ev}{w^T v}$$

From Theorem 4.8, we can obtain the following.

Corollary 4.2. *Suppose matrix $A \in \mathbb{R}^n$ has n distinct eigenvalues. Moreover, suppose there exists $i \in \{1, \cdots, n\}, j \in \{1, \cdots, n\}$ such that the ith entry of each left eigenvector of A is nonzero and the jth entry of each right eigenvector of A is nonzero. Select such an i and j. Let $E_{ij} \in \mathbb{R}^{n \times n}$ be a matrix of zeros except the (i, j) entry, which has value 1. Then for all $t \in \mathbb{R}, \quad t \neq 0$, $\Lambda(A) \cap \Lambda(A + tE_{ij}) = \emptyset$.*

Proof. Since A has distinct eigenvalues $\{\lambda_1, \cdots, \lambda_n\}$ all the eigenvalues are simple. Let w_q, v_q be left and right eigenvectors associated with λ_q. For each $q \in \{1, \cdots, n\}$ select an $\epsilon_q > 0$ such that

$$\epsilon_q < \frac{|w_q^T E_{ij} v_q|}{|w_q^T v_q|}. \tag{4.64}$$

This is possible because by construction $w_q^T v_q$ is nonzero, and by assumption the ith entry of w_q and the jth entry of v_q is nonzero. For each $q \in \{1, \cdots, n\}$, we know that $|\lambda_q(t) - \lambda_q - tw_q^T E_{ij} v_q / w_q^T v_q| \leq |t|\epsilon_q$ if $|t| < \delta_q(\epsilon_q)$. Let $\mathcal{D}_\lambda = \min_{s,r \in \{1,\cdots,n\}, s \neq r} |\lambda_s - \lambda_r|$. Choose $t^* > 0$ such that

$$|t^*|\epsilon_q + |t^* w_q^T E_{ij} v_q / w_q^T v_q| < \mathcal{D}_\lambda / 2, \;\; t^* < \delta_q(\epsilon_q), \;\; \forall\, q \in \{1, \cdots, n\}. \tag{4.65}$$

Then we have $||\lambda_q(t^*) - \lambda_q| - t^*|w_q^T E_{ij} v_q / w_q^T v_q|| \leq t^* \epsilon_q$, which implies

$$-t^*\epsilon_q + t^*|w_q^T E_{ij} v_q / w_q^T v_q| \leq |\lambda_q(t^*) - \lambda_q| \leq t^*\epsilon_q + t^*|w_q^T E_{ij} v_q / w_q^T v_q| \tag{4.66}$$

From (4.64) and (4.66), we know that $0 < |\lambda_q(t^*) - \lambda_q|$. Moreover, from (4.65) and (4.66), we have for arbitrary $s, r \in \{1, \cdots, n\}, s \neq r$

$$\begin{aligned} |\lambda_s(t^*) - \lambda_r| &= |\lambda_s(t^*) - \lambda_s + \lambda_s - \lambda_r| \\ &\geq |\lambda_s - \lambda_r| - |\lambda_s - \lambda_s(t^*)| \\ &> \mathcal{D}_\lambda - \mathcal{D}_\lambda / 2 \\ &> 0. \end{aligned}$$

Consequently, the eigenvalue spectra of $A + t^* E_{ij}$ is disjoint from the eigenvalue spectra of A. From Laplace's formula, we observe that there are polynomials $g(\lambda)$ and $h(\lambda)$ such that

$$\det(\lambda I - A - tE_{ij}) = g(\lambda) + th(\lambda).$$

For a given λ_q, we know that $g(\lambda_q) = 0$ and $t^* h(\lambda_q) \neq 0$. Thus $h(\lambda_q) \neq 0$. As a result, for $t \neq 0$, we have $\det(\lambda_q I - A - tE_{ij}) \neq 0$. The result follows. □

We see that the ability to modify the eigenvalues of a matrix, is heavily linked to the sparsity of the eigenvectors. Previous work [58, 59] has has investigated the sparsity of eigenvectors as a function of the nonzero structure of the matrix A. Further analysis here is left for future work.

4.2.4 False Data Injection Detection

In this section, we examine the effectiveness of the moving target defense for detection in the case of a stochastic system. Here, we assume that

$$x_{k+1} = A_k x_k + w_k, \;\; y_k = C_k x_k + D^a d_k^a + v_k. \tag{4.67}$$

w_k and v_k are independent and IID Gaussian process and sensor noise where $w_k \sim \mathcal{N}(0, Q)$ and $v_k \sim \mathcal{N}(0, R)$. For notational simplicity we assume that the covariances $Q \geq 0$ and $R > 0$ are constant. However, we can obtain the ensuing results even in the case that Q and R are dependent on A_k and C_k.

The adversary's and defender's information and strategy is unchanged except we assume the defender has knowledge of the distribution of the initial state. Specifically, $f(x_0|\mathcal{I}_{-1}) = \mathcal{N}(\hat{x}_0^-, P_{0|-1})$. Moreover, both the defender and attacker are aware of the noise statistics. We first would like to show that a moving target defense leveraging the design recommendations listed above can almost surely detect harmful false data injection attacks. To characterize detection performance, we consider the additive bias the adversary injects on the normalized residues Δz_k due to his sensor attacks. The residues, z_k, are the normalized difference between the observed

measurements and their expected values. This is slightly different from the residues considered in previous chapters which were unnormalized. The bias on the normalized residues is given by

$$\begin{aligned}
\Delta e_k &= (A_{k-1} - K_k C_k A_{k-1})\Delta e_{k-1} - K_k D^a d_k^a, \quad \Delta e_{-1} = 0, \\
\Delta z_k &= \bar{P}_k^{-\frac{1}{2}} \left(C_k A_{k-1} \Delta e_{k-1} + D^a d_k^a\right), \\
\bar{P}_k &= (C_k P_{k|k-1} C_k^T + R), \quad K_k = P_{k|k-1} C_k^T (C_k P_{k|k-1} C_k^T + R)^{-1}, \\
P_{k+1|k} &= A_k P_{k|k-1} A_k^T + Q - A_k P_{k|k-1} C_k^T (C_k P_{k|k-1} C_k^T + R)^{-1} C_k P_{k|k-1} A_k^T,
\end{aligned}$$

where Δe_k is the bias injected on the a posteriori state estimation error obtained by an optimal Kalman filter, and $P_{k|k-1}$ is the a priori error covariance. As we have seen, a residue detector such as the χ^2 detector will recognize large residues and mark them as belonging to an attack. We now show that an admissible adversary is restricted in the bias he can inject on the state estimation error without significantly biasing the residues and incurring detection. In particular, we have the following result.

Theorem 4.9. *Suppose a defender uses a moving target defense leveraging the design recommendations listed above. Then* $\limsup_{k\to\infty} \|\Delta e_k\| = \infty \implies \limsup_{k\to\infty} \|\Delta z_k\| = \infty$ *with probability 1.*

Proof. In this section, the norm $\|\cdot\|$ refers specifically to the 2 norm. Assume to the contrary that the residues are bounded $\|\Delta z_k\| \leq M$. Define the indices of a peak subsequence as follows. $i_0 = 0$, $i_k = \min \kappa$ such that $\kappa > i_{k-1}, \|\Delta e_\kappa\| > \|\Delta e_t\| \; \forall t \leq \kappa$. Such a sequence exists since the estimation bias is unbounded. Also define the indices j_k such that $j_k = \min \kappa$ such that $j_k \geq i_k$, $j_k \mod N = N - 1$. Observe that

$$\Delta e_k = A_{k-1} \Delta e_{k-1} - K_k \bar{P}_k^{\frac{1}{2}} \Delta z_k. \tag{4.68}$$

As a result, we have $A_{j_k} \Delta e_{j_k} = A_{i_k}^{j_k - i_k + 1} \Delta e_{i_k} - \sum_{t=i_k+1}^{j_k} A_{i_k}^{j_k+1-t} K_t \bar{P}_t^{\frac{1}{2}} \Delta z_t$. Define $a_m > 0$ and $a_M > 0$ as

$$a_m \triangleq \min_{\substack{j\in\{1,\cdots,l\} \\ q\in\{0,\cdots,N\}}} \sigma_{min}(A(j)^q), \;\; a_M \triangleq \max_{\substack{j\in\{1,\cdots,l\} \\ q\in\{0,\cdots,N-1\}}} \|A(j)^q\|.$$

where $\sigma_{min}(\cdot)$ denotes the smallest singular value. Moreover let p_M and c_M be given by

$$p_M = \sup_k \|P_{k|k-1}\|, \quad c_M = \max_{j\in\{1,\cdots,l\}} \|C(j)\|.$$

Observe that a_m is nonzero since each $A(i)$ is invertible from recommendation 5. a_M and c_M are bounded above since we are taking the maximum over a finite set of bounded elements. Moreover, p_M is bounded above since the error covariance is bounded above. A complete argument is omitted due to space considerations. However, since all pairs $(A(i), C(i)) \in \Gamma$ are observable from recommendation 4 it can be shown that $x_{Nk+n}, k \in \mathbb{N}$ is a linear combination of $y_{Nk:Nk+n-1}$ and $2n$ random variables, where the linear combination is dependent only on $(A(Nk), C(Nk))$. Thus, the covariance of x_{Nk+n} given $y_{0:Nk+n-1}$ is bounded. It can be shown that the covariance of $x_{Nk+n+j}, j \in \{1, \cdots, N-1\}$ is bounded given $y_{0:Nk+n+j-1}$ simply by computing predictive covariances given $y_{0:Nk+n-1}$. As a result, we have

$$\|A_{j_k}\Delta e_{j_k}\| \geq a_m\|\Delta e_{i_k}\| - (N-1)a_M p_M c_M \frac{M}{\sqrt{\lambda_{min}(R)}}.$$

where $\lambda_{min}(R)$ is the smallest eigenvalue of R. $\lambda_{min}(R)$ is nonzero since $R > 0$. Therefore, since $\|\Delta e_{i_k}\| \to \infty$, we have that $\|A_{j_k}\Delta e_{j_k}\| \to \infty$.

Now, with some abuse of notation let $D^a d^a_{t_1:t_2} = \begin{bmatrix}(D^a d^a_{t_1})^T & \cdots & (D^a d^a_{t_2})^T\end{bmatrix}^T$. Suppose $(A_{j_{k+1}}, C_{j_{k+1}}) = (A(q_1), C(q_1))$. Then,

$$D^a d_{j_k+1:j_k+N} = -\mathcal{O}^S_{N,q_1} A_{j_k}\Delta e_{j_k} + F_{j_k+1}(q_1)\Delta z^{(q_1)}_{j_k+1:j_k+N}.$$

where $F_{j_k+1}(q_1) =$

$$\begin{bmatrix} \mathcal{P}^{\frac{1}{2}}_{j_k+1}(q_1) & 0 & \cdots & 0 \\ C(q_1)A(q_1)K_{j_k+1}(q_1)\bar{P}^{\frac{1}{2}}_{j_k+1}(q_1) & \bar{P}^{\frac{1}{2}}_{j_k+2}(q_1) & \cdots & 0 \\ C(q_1)A^2(q_1)K_{j_k+1}(q_1)\bar{P}^{\frac{1}{2}}_{j_k+1}(q_1) & C(q_1)A(q_1)K_{j_k+2}(q_1)\bar{P}^{\frac{1}{2}}_{j_k+2}(q_1) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ C(q_1)A^{N-1}(q_1)K_{j_k+1}(q_1)\bar{P}^{\frac{1}{2}}_{j_k+1}(q_1) & C(q_1)A^{N-2}(q_1)K_{j_k+2}(q_1)\bar{P}^{\frac{1}{2}}_{j_k+2}(q_1) & \cdots & \bar{P}^{\frac{1}{2}}_{j_k+N}(q_1) \end{bmatrix}. \tag{4.69}$$

Through a similar analysis as done above, it can shown that $\|F_{j_k+1}(q_1)\|$ is bounded above. Alternatively, if $q_2 \neq q_1$, is chosen then

$$D^a d_{j_k+1:j_k+N} = -\mathcal{O}^S_{N,q_2} A_{j_k} \Delta e_{j_k} + F_{j_k+1}(q_2) \Delta z^{(q_2)}_{j_k+1:j_k+N}.$$

Thus, to insert valid inputs for modes q_1 and q_2, we require

$$(\mathcal{O}^S_{N,q_1} - \mathcal{O}^S_{N,q_2}) A_{j_k} \Delta e_{j_k} = F_{j_k+1}(q_1) \Delta z^{(q_1)}_{j_k+1:j_k+N} - F_{j_k+1}(q_2) \Delta z^{(q_2)}_{j_k+1:j_k+N}. \tag{4.70}$$

If the residues are bounded in both scenarios, then the right hand side of (4.70) is bounded. Next, due to design recommendations 1 and 2 and Theorem 4.7, there is no solution to $0 \neq \mathcal{O}^S_{N,q_1} v_1 = \mathcal{O}^S_{N,q_1} v_2$. Using this fact and the assumption that each (A_k, C_k) pair is observable, $(\mathcal{O}^S_{N,q_1} - \mathcal{O}^S_{N,q_2})$ has no nontrivial null space. Thus, the left hand side of (4.70) is unbounded. As a result, there is no way for the attacker to guess incorrectly and insert bounded residues. From, recommendation 3, there is a nonzero probability the attacker guesses (A_{j_k+1}, C_{j_k+1}) incorrectly and the result holds. □

Thus the attacker is able to destabilize the estimation error only by destabilizing the residues. As such, there is a point where an attacker is unable to introduce additional bias to the estimation error without revealing his presence due to his effect on the measurement residues.

Remark 4.13. *Design recommendation 1 can be relaxed in the stochastic case for purposes of detecting false data injection attacks. In particular for all non-equal pairs $i, j \in \{1, \cdots, l\}$ we only require $0 \neq \mathcal{O}^S_{N,i} v \neq \mathcal{O}^S_{N,j} v$ for all v instead of $0 \neq \mathcal{O}^S_{N,i} v_i \neq \mathcal{O}^S_{N,j} v_j$ for all v_i, v_j. Here, a big difference is that in the stochastic case we give the defender some knowledge of the distribution of the initial state.*

4.2.5 Resilient Estimation and Identification

While the moving target approach guarantees we can detect unbounded false data injection attacks, we wish to also identify specific malicious sensors as in the deterministic case. In the remainder of this section, we construct a resilient estimator. We will fuse state estimates generated by individual

sensors since previous results [60, 61] suggest such an estimator has better fault tolerance. This is desirable in our work since we are attempting to force a normally stealthy adversary to generate faults. We will show that an attacker can destabilize this estimator only if the culprit sensors can be identified. In particular, we will show that the estimation error will become unbounded only if the bias on a sensor residue is also unbounded.

To begin, we assume that for each sensor s,

$$NS(\mathcal{O}^s_{n,1}) = NS(\mathcal{O}^s_{n,2}) = \cdots = NS(\mathcal{O}^s_{n,l}), \tag{4.71}$$

where $NS(A)$ denotes the null space of A. Such a condition is realistic since it implies that changing the system dynamics does not affect what portion of the state the sensor itself can observe. As a result, using the Kalman decomposition, for each sensor s, there exists a state transformation $T_s = \begin{bmatrix} T^{uo}_s & T^o_s \end{bmatrix}$ such that

$$\begin{bmatrix} T^{uo}_s & T^o_s \end{bmatrix} \begin{bmatrix} \zeta^{uo}_{k,s} \\ \zeta_{k,s} \end{bmatrix} = x_k, \quad \begin{bmatrix} T^{uo}_s & T^o_s \end{bmatrix} \begin{bmatrix} \omega^{uo}_{k,s} \\ \omega_{k,s} \end{bmatrix} = w_k.$$

Here, the columns of T^{uo}_s are a basis for $NS(\mathcal{O}^s_{n,1})$, while the columns of T^o_s should be chosen so the resulting T_s is invertible.

Moreover, using the same transform T_s, there exists a $\Gamma^s = \{(C_s(1), A_s(1)), \cdots (C_s(l), A_s(l))\}$ corresponding to Γ such that

$$\zeta_{k+1,s} = A_{k,s}\zeta_{k,s} + \omega_{k,s}, \quad y^s_k = C_{k,s}\zeta_{k,s} + v^s_k, \tag{4.72}$$

where each pair $(A_{k,s}, C_{k,s})$ is observable and belongs to Γ^s.

Before we continue, we remark that (4.71) allows us to improve the guarantees obtained in the deterministic case with Corollary 4.1. In particular, we argue the attacker will be forced to perpetually insert inputs to remain stealthy. In particular, if an attacker has inserted input $\begin{bmatrix} (C_s(i))^T & (C_s(i)A_s(i))^T & \cdots & (C_s(i)A^{N-1}_s(i))^T \end{bmatrix}^T \zeta^*_0$ for some ζ^*_0 not equal to 0, he must next insert $\begin{bmatrix} (C_s(j))^T & (C_s(j)A_s(j))^T & \cdots & (C_s(j)A^{N-1}_s(j))^T \end{bmatrix} A^{N-1}_s(i)\zeta^*_0$ for some $j \in \{1, \cdots, l\}$

to have an opportunity to remain stealthy. We argue these next set of N inputs are nonzero. First, $A_s^{N-1}(i)\zeta_0^*$ is nonzero. This is because $A(j)$ is by assumption invertible and thus $A_s(j)$ is invertible by properties of the Kalman decomposition. As a result, since $(A_s(j), C_s(j))$ is observable, the conjecture holds.

By performing a change of variables on $\hat{x}_0^-$, a Kalman filter with bounded covariance (see proof of Theorem 4.9) can be constructed to estimate $\zeta_{k,i}$ given $y_{0:k}^i$. Specifically, define

$$\begin{bmatrix} - \\ \hat{\zeta}_{0,s}^- \end{bmatrix} \triangleq T_s^{-1}\hat{x}_0^-, \quad \begin{bmatrix} - & - \\ - & Q_{s_1,s_2} \end{bmatrix} \triangleq T_{s_1}^{-1} Q T_{s_2}^{-1\,T}, \quad \begin{bmatrix} - & - \\ - & P_{0|-1}^{s_1,s_2} \end{bmatrix} \triangleq T_{s_1}^{-1} P_{0|-1} T_{s_2}^{-1\,T}.$$

From the definition of the Kalman filter, we have

$$\begin{aligned} \hat{\zeta}_{k,s} &= (I - K_{k,s}C_{k,s})\hat{\zeta}_{k,s}^- + K_{k,s}y_k^s, \quad \hat{\zeta}_{k+1,s}^- = A_{k,s}\hat{\zeta}_{k,s}, \\ K_{k,s} &= P_{k|k-1}^{s,s} C_{k,s}^T (C_{k,s} P_{k|k-1}^{s,s} C_{k,s}^T + R_{ss})^{-1}, \quad P_{k+1|k}^{s_1,s_2} = A_{k,s_1} P_k^{s_1,s_2} A_{k,s_2}^T + Q_{s_1 s_2}, \\ P_k^{s,s} &= P_{k|k-1}^{s,s} - K_{k,s}C_{k,s}P_{k|k-1}^{s,s}, \quad z_{k,s} = (C_{k,s}P_{k|k-1}^{s,s}C_{k,s}^T + R_{ss})^{-\frac{1}{2}}(y_k^s - C_{k,s}\hat{\zeta}_{k,s}^-), \end{aligned} \tag{4.73}$$

where R_{ij} is the (i,j) entry of R.

Here $\hat{\zeta}_{k,s} = \mathbb{E}[\zeta_k | y_{0:k}^s]$, $\hat{\zeta}_{k,s}^- = \mathbb{E}[\zeta_k | y_{0:k-1}^s]$ are optimal estimates of the reduced state for sensor s. In the construction of our fusion estimator, we will also need to compute $\mathbb{E}[e_{k,s_1} e_{k,s_2}^T]$ and $\mathbb{E}[e_{k,s_1}^- e_{k,s_2}^{-T}]$ where $e_{k,s} \triangleq \zeta_{k,s} - \hat{\zeta}_{k,s}$ and $e_{k,s}^- = \zeta_{k,s} - \hat{\zeta}_{k,s}^-$. We observe that $P_k^{s,s} = \mathbb{E}[e_{k,s} e_{k,s}^T]$ and $P_{k|k-1}^{s,s} = \mathbb{E}[e_{k,s}^- e_{k,s}^{-T}]$. Moreover, note that $P_{0|-1}^{s_1,s_2} = \mathbb{E}[e_{0,s_1}^- e_{0,s_2}^{-T}]$. The error dynamics for a given sensor s are given by

$$e_{k,s} = (I - K_{k,s}C_{k,s})e_{k,s}^- - K_{k,s}v_k^s, \qquad e_{k+1,s}^- = A_{k,s}e_{k,s} + \omega_{k,s} \tag{4.74}$$

Let $P_k^{s_1,s_2} \triangleq \mathbb{E}[e_{k,s_1} e_{k,s_2}^T]$ and $P_{k|k-1}^{s_1,s_2} \triangleq \mathbb{E}[e_{k,s_1}^- e_{k,s_2}^{-T}]$. From (4.74), we have

$$\begin{aligned} P_k^{s_1,s_2} &= (I - K_{k,s_1}C_{k,s_1}) P_{k|k-1}^{s_1,s_2} (I - K_{k,s_2}C_{k,s_2})^T + K_{k,s_1} R_{s_1 s_2} K_{k,s_2}^T, \\ P_{k+1|k}^{s_1,s_2} &= A_{k,s_1} P_k^{s_1,s_2} A_{k,s_2}^T + Q_{s_1 s_2}. \end{aligned} \tag{4.75}$$

Note that (4.75) also holds for $s_1 = s_2$. We would like to use the individual state estimates $\hat{\zeta}_{k,s}$ associated with each sensor s to obtain an overall state estimate of x_k. To do this, first define $x_{k,s}^o$

as

$$x^o_{k,s} = T^o_s \hat{\zeta}_{k,s} + \eta_{k,s} \tag{4.76}$$

where $\eta_{k,s}$ is an IID sequence of Gaussian random variables with $\eta_{k,s} \sim \mathcal{N}(\mathbf{0}, \epsilon I)$ for some small $\epsilon > 0$. Moreover $\{\eta_{k,s_1}\}$ and $\{\eta_{k,s_2}\}$ are independent sequences. $\eta_{k,s}$ is a mathematical artifact introduced so the subsequent estimator has a simplified closed form and can be easily removed or mitigated by letting ϵ tend to 0. Now, we observe that

$$x_k = T^{uo}_s \zeta^{uo}_{k,s} + x^o_{k,s} + T^o_s e_{k,s} - \eta_{k,s}. \tag{4.77}$$

From here we obtain

$$\hat{\mathbf{y}}_\mathbf{k} = W\mathbf{x}_\mathbf{k} + \eta_\mathbf{k}, \tag{4.78}$$

$$\hat{\mathbf{y}}_\mathbf{k} = \begin{bmatrix} x^o_{k,1} \\ x^o_{k,2} \\ \vdots \\ x^o_{k,m} \end{bmatrix}, \mathbf{x}_\mathbf{k} = \begin{bmatrix} \zeta^{uo}_{k,1} \\ \zeta^{uo}_{k,2} \\ \vdots \\ \zeta^{uo}_{k,m} \\ x_k \end{bmatrix}, \eta_\mathbf{k} = \begin{bmatrix} -T^o_1 e_{k,1} + \eta_{k,1} \\ -T^o_2 e_{k,2} + \eta_{k,2} \\ \vdots \\ -T^o_m e_{k,m} + \eta_{k,m} \end{bmatrix}, W = \begin{bmatrix} -T^{uo}_1 & \mathbf{0} & \cdots & \mathbf{0} & I \\ \mathbf{0} & -T^{uo}_2 & \cdots & \mathbf{0} & I \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & -T^{uo}_m & I \end{bmatrix}.$$

It can be seen that $\eta_\mathbf{k}$ is normally distributed so that $\eta_\mathbf{k} \sim \mathcal{N}(0, \mathcal{Q})$, where $\mathcal{Q} > 0$ consists of $m \times m$ blocks where the (i, j) block is given by $(T^o_i P^{i,j}_k T^{oT}_j + \delta_{ij}\epsilon I)$. Here, δ_{ij} is the Kronecker delta. The minimum variance unbiased estimate (MVUB) [33] of $\mathbf{x}_k$ given $\hat{\mathbf{y}}_\mathbf{k}$ is given by

$$\hat{\mathbf{x}}_k = (W^T \mathcal{Q}^{-1} W)^{-1} W^T \mathcal{Q}^{-1} \hat{\mathbf{y}}_\mathbf{k} \tag{4.79}$$

The last n entries of $\hat{\mathbf{x}}_\mathbf{k}$, denoted as $\hat{x}^*_k$, constitute a (MVUB) estimate of x_k given the set of sensor estimates $\hat{\mathbf{y}}_\mathbf{k}$. The covariance of this estimate is given by

$$\text{Cov}(\mathbf{x}_k - \hat{\mathbf{x}}_k) = (W^T \mathcal{Q}^{-1} W)^{-1}. \tag{4.80}$$

The proposed estimator is well defined since $NS(W) = 0$. If, W had a nontrivial null space, this would imply there exists $x \neq 0$ and $\zeta_1, \cdots, \zeta_m$ such that

$$T^{uo}_i \zeta_i = x, \quad \forall\, i \in \{1, \cdots, m\}. \tag{4.81}$$

This would imply that $\cap_{i=1}^{m} NS(\mathcal{O}_{n,1}^{i}) \neq 0$, which contradicts the observability of each pair (A_k, C_k). We next show that the proposed estimator of x_k has bounded covariance.

Theorem 4.10. *Consider the estimator of x_k defined by (4.73),(4.75),(4.76),(4.78),(4.79). The estimator has bounded covariance.*

Proof. We first prove that $\mathcal{Q} > 0$ is bounded above. The ith diagonal block of $\mathcal{Q}$ has covariance $(T_i^o P_k^{i,i} T_i^{oT} + \epsilon I)$. Using the same argument as in the proof of Theorem 4.9, we see that $P_k^{i,i}$ is bounded. Consequently $(T_i^o P_k^{i,i} T_i^{oT} + \epsilon I)$ and $\mathcal{Q}$ are bounded.

Next consider $\hat{\mathbf{x}}_{\mathbf{k}}^{\mathbf{uw}} = (W^T W)^{-1} W^T \hat{\mathbf{y}}_{\mathbf{k}}$. Since $\mathbf{x}_k - \hat{\mathbf{x}}_{\mathbf{k}}^{\mathbf{uw}} = -(W^T W)^{-1} W^T \eta_{\mathbf{k}}$, $\hat{\mathbf{x}}_{\mathbf{k}}^{\mathbf{uw}}$ is an unbiased estimator of $\mathbf{x}_k$ with covariance $(W^T W)^{-1} W^T \mathcal{Q} W (W^T W)^{-1}$. Since W is fixed and $\mathcal{Q} > 0$ is bounded above, $(W^T W)^{-1} W^T \mathcal{Q} W (W^T W)^{-1}$ is also bounded above.

Finally, since the proposed estimator is MVUB, we see

$$\text{tr}\left((W^T \mathcal{Q}^{-1} W)^{-1}\right) \leq \text{tr}\left((W^T W)^{-1} W^T \mathcal{Q} W (W^T W)^{-1}\right).$$

Thus, $\text{Cov}(\mathbf{x_k} - \hat{\mathbf{x}}_k)$ and the covariance of x_k defined by the last $n \times n$ block of $\text{Cov}(\mathbf{x_k} - \hat{\mathbf{x}}_{\mathbf{k}})$ are bounded. □

To close this subsection, we demonstrate that the proposed estimator is sensitive to biases in individual residues $\Delta z_{k,s}$, specifically showing that an infinite bias introduced into the estimator implies that the residues are also infinite. Define $\mathbf{e_k} \triangleq \mathbf{x_k} - \hat{\mathbf{x}}_{\mathbf{k}}$ and $\Delta \mathbf{e}_k$ as the bias inserted on $\mathbf{e_k}$ due to the adversary's inputs. Moreover, let $e_k^* = x_k - \hat{x}_k^*$ and let Δe_k^* and $\Delta e_{k,i}$ be the bias inserted on e_k^* and $e_{k,i}$ respectively due to the adversary's inputs. We have the following result.

Theorem 4.11. *Consider the estimator of x_k defined by (4.73),(4.75),(4.76),(4.78),(4.79). Then, with probability 1, $\limsup_{k\to\infty} \|\Delta e_k^*\| = \infty \implies \limsup_{k\to\infty} \|\Delta z_{k,i}\| = \infty$ for some $i \in \{1, \cdots, m\}$.*

Proof. First, we observe that

$\mathbf{e}_k = -(W^T\mathcal{Q}^{-1}W)^{-1}W^T\mathcal{Q}^{-1}\eta_{\mathbf{k}}$. As a result,

$$\Delta\mathbf{e}_k = (W^T\mathcal{Q}^{-1}W)^{-1}W^T\mathcal{Q}^{-1}T_{diag}\Delta e_{k,S}. \tag{4.82}$$

where

$$T_{diag} = \begin{bmatrix} T_1^o & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & T_2^o & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & T_m^o \end{bmatrix}, \quad \Delta e_{k,S} = \begin{bmatrix} \Delta e_{k,1} \\ \Delta e_{k,2} \\ \vdots \\ \Delta e_{k,m} \end{bmatrix}.$$

Next, we will show that $K^* = (W^T\mathcal{Q}^{-1}W)^{-1}W^T\mathcal{Q}^{-1}T_{diag}$ has bounded norm. In particular observe that

$$\|K^*\| = \|(W^T\frac{\mathcal{Q}^{-1}}{\|\mathcal{Q}^{-1}\|}W)^{-1}W^T\frac{\mathcal{Q}^{-1}}{\|\mathcal{Q}^{-1}\|}T_{diag}\| \leq \|(W^T\frac{\mathcal{Q}^{-1}}{\|\mathcal{Q}^{-1}\|}W)^{-1}\|\|W^T\frac{\mathcal{Q}^{-1}}{\|\mathcal{Q}^{-1}\|}T_{diag}\|.$$

Clearly, $\|W^T\frac{\mathcal{Q}^{-1}}{\|\mathcal{Q}^{-1}\|}T_{diag}\|$ has bounded norm. Moreover, using a similar argument as in the proof of Theorem 4.10, $\|(W^T\frac{\mathcal{Q}^{-1}}{\|\mathcal{Q}^{-1}\|}W)^{-1}\|$ has bounded norm. Thus, $\|K^*\|$ is bounded. Consequently, from (4.82), $\limsup_{k\to\infty}\|\Delta e_k^*\| = \infty \implies \limsup_{k\to\infty}\|\Delta e_{k,i}\| = \infty$ for some $i \in \{1, \cdots, m\}$. However, from Theorem 4.9, this implies $\limsup_{k\to\infty}\|\Delta z_{k,i}\| = \infty$ and the result holds. □

While the proposed estimator does not guarantee each malicious sensor will be identified, it does guarantee that the defender will be able to identify and remove sensors whose attacks cause unbounded bias in the estimation error simply by analyzing each sensor's measurements individually. This is due to the fact that the bias on residues of such sensors will grow unbounded, which can be easily detected by some χ^2 detector. As a result, for each individual sensor s, we propose the following detector at time k, which can be used to identify malicious behavior,

$$\sum_{j=k-T^*+1}^{k} z_{j,s}^2 \underset{\mathcal{H}_0^s}{\overset{\mathcal{H}_1^s}{\gtrless}} \tau_k^i. \tag{4.83}$$

In this scenario, $\mathcal{H}_1^s$ is the hypothesis that sensor s is malfunctioning and $\mathcal{H}_0^s$ is the hypothesis that sensor s is working normally. In practice a sensor s who repeatedly fails detection can be removed

from consideration when obtaining a state estimate and the proposed fusion based estimation scheme can be adjusted accordingly.

4.2.6 Numerical Example

We consider a numerical example where $l = 7$ and $A(j)$ and $C(j)$ are given by

$$A(k) = \begin{bmatrix} A_{11}(j) & A_{12}(j) & 0 & 0 & 0 \\ 0 & A_{22}(j) & 0 & A_{24}(j) & 0 \\ 0 & 0 & A_{33}(j) & 0 & A_{35}(j) \\ 0 & 0 & 0 & A_{44}(j) & A_{45}(j) \\ 0 & 0 & 0 & 0 & A_{55}(j) \end{bmatrix},$$

$$C(j) = \begin{bmatrix} C_1(j) \\ C_2(j) \end{bmatrix},$$

$$C_i(j) = \begin{bmatrix} C_{1,i}(j) & 0 & 0 & 0 & 0 \\ 0 & C_{2,i}(j) & 0 & 0 & 0 \\ 0 & 0 & C_{3,i}(j) & 0 & 0 \\ 0 & 0 & 0 & C_{4,i}(j) & 0 \\ 0 & 0 & 0 & 0 & C_{5,i}(j) \end{bmatrix}.$$

where $A_{ij}(j) \in \mathbb{R}^{3\times 3}$ and $C_{i,j}(j) \in \mathbb{R}^{1\times 3}$ are scaled uniformly random matrices with $A_{ii}(j)$ unstable. Moreover Q and R are appropriately sized matrices generated by multiplying a uniform random matrix by its transpose. The system matrices are changed independently and randomly every $2n$ time steps where $n = 15$ and each $(A(j), C(j))$ pair has equal likelihood.

We assume that the adversary biases the last 5 sensors (measured by $C_2(j)$) by performing the attack formulated in Theorem 4.5. Here, the attacker guesses the system matrices randomly every $2n$ time steps and x_0^* is chosen identically for each sensor. A χ^2 detector (4.83) with window 5 and false alarm probability $\alpha_k^i = 6.9 \times 10^{-8}$ is implemented for each sensor based on their local

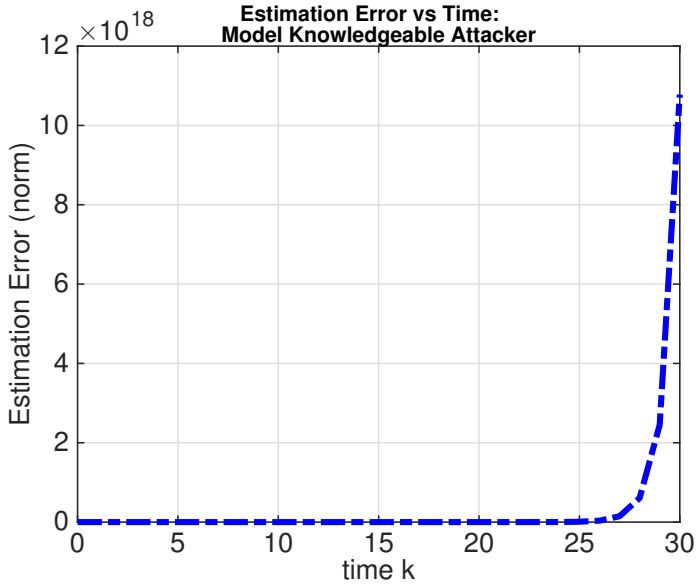


Figure 4.7: Estimation error vs time under attack when the adversary knows the system dynamics using Hybrid Moving Target

Kalman filters. A centralized χ^2 detector with window 3 derived from the optimal centralized Kalman filter performs detection with false alarm probability $\alpha_k = 4.2 \times 10^{-4}$.

We first note that under normal operation, the estimators achieve similar performance, with the average mean squared of the optimal Kalman filter at 22.9 and the average mean squared error of the proposed estimator at 23.4. In Fig. 4.7 and 4.8, we consider the system with the moving target under attack. However, we assume the attacker is aware of the exact sequence of time varying matrices. As such the attacker is able to destabilize the estimation error in Fig. 4.7 while the sensor residues appear normal in Fig. 4.8.

Finally, in Fig. 4.9, we plot the norm of the estimation error for both the proposed estimator and optimal Kalman filter as a function of time when the attacker is forced to randomly guess the system model. Here the attacker is detected in 2 time steps and perfectly identified in 8 time steps. When a sensor is identified, it is removed from consideration when performing fusion or optimal Kalman filtering. It can be seen that while under attack, the proposed fusion based estimator is better able to recover from the adversary's actions. While the estimation error becomes large, the attacker's

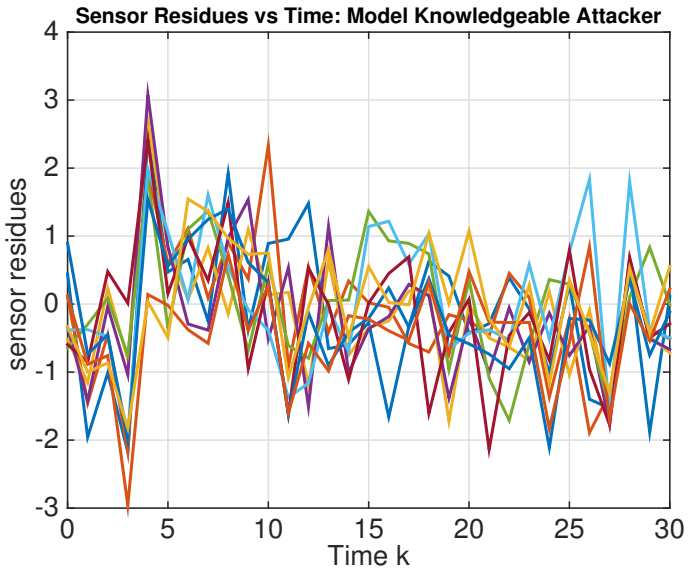


Figure 4.8: Residue vs time under attack when the adversary knows the system dynamics. The defender is using the Hybrid Moving Target

effect on the system can be mitigated. In particular, since an attack is detected within two time steps, a robust controller ignoring the incorrect state estimates can be utilized until identification has occurred. This will limit the effects of incorrect feedback.

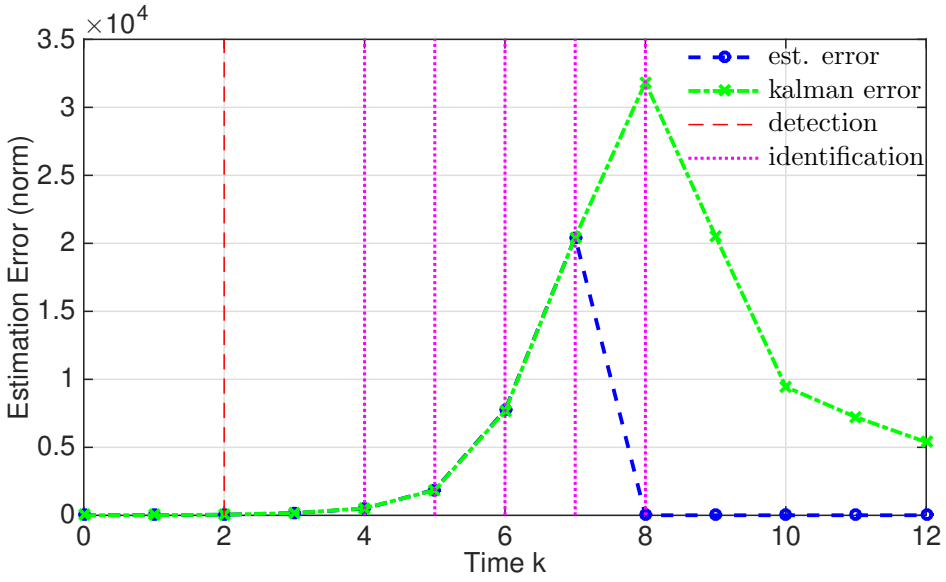


Figure 4.9: Estimation error vs time under attack when the adversary does not know the true dynamics of the Hybrid Moving Target. All sensor attacks are identified. The proposed fusion based estimator and the centralized Kalman filter are illustrated.

Chapter 5

Structural System Design

In the previous chapters, we have considered active detection as an online mechanism for detecting classes of integrity attacks. However, offline or robust design can be used to dampen the effectiveness of potential attackers. In this chapter, we consider how structural properties of the plant and sensing infrastructure can impact our ability to detect perfect attacks and zero dynamics attacks. Perfect attacks are able to characterize the set of stealthy attacks when the defender has knowledge of the initial state while zero dynamics attacks characterize the set of stealthy attacks when the defender has no knowledge of the initial state. By the careful design of these topologies, we can almost surely eliminate these attacks when considering a resource limited attacker. In the deterministic case, this forces a resource limited attacker to be deterministically detectable, while in the stochastic case, this can limit the impact of an attacker that wishes to remain stealthy. The chapter is summarized as follows. In section 5.1, we introduce background on perfect and zero dynamics attacks. In section 5.2, we provide structural conditions which allow a defender to eliminate the existence of these attacks. Finally, in section 5.3, we consider the minimal robust design of distributed control system to balance the costs of sensing and communication with the need for security. The results in this chapter are partially based on [62], [63], and [64].

5.1 A Background on Zero Dynamics Attacks

5.1.1 Detection in Deterministic Systems

We commence the study of zero dynamics and perfect attacks by examining the set of stealthy attacks against a deterministic control system.

$$x_{k+1} = Ax_k + Bu_k + B^a u_k^a, \;\; y_k = Cx_k + D^a d_k^a. \tag{5.1}$$

In this section, we will assume (A, C) is observable. Additionally, assume that an attack commences at time 0 and that x_0 is known to the defender.

We make the assumption here that an attacker is restricted to manipulate a fixed set of inputs and outputs described by the matrices B^a and D^a. Unless otherwise stated, we without loss of generality, assume B^a is full column rank. We define the set of attackable inputs and outputs respectively as $\mathcal{K}_u^a \triangleq \{\delta_1, \cdots, \delta_{p_{**}}\}$ and $\mathcal{K}_y^a \triangleq \{\eta_1, \cdots, \eta_{m_*}\}$. In this case, $j \in \mathcal{K}_u^a$ and $l \in \mathcal{K}_y^a$ implies an attack can modify the jth entry of u_k and lth entry of y_k for all k. B^a can be constructed as a matrix whose columns are basis vectors of a subspace generated by $\{B_{\delta_1}, \cdots, B_{\delta_{p_{**}}}\}$ where B_j is the jth column of B. B^a can be extended accordingly if an attacker introduces additional actuators. We assume $B^a \in \mathbb{R}^{n \times p_*}$. The sensor attack matrix $D^a \in \mathbb{R}^{m \times m_*}$ can be defined entrywise as follows

$$D^a(s,t) \triangleq \mathbb{I}_{s=\eta_i, t=i}. \tag{5.2}$$

Moreover, assume the defender's control policy at time k is a deterministic function of the model $\mathcal{M} = (A, B, C)$, the previous inputs $u_{0:k-1}$, the previous outputs $y_{0:k}$, and the initial state x_0 so that

$$u_k = \mathcal{U}_k(A, B, C, u_{0:k-1}, y_{0:k}, x_0). \tag{5.3}$$

For this system, it can be inductively shown that u_k is deterministic. As a result, y_k is deterministic for all k. Let $y_k(x_0, u_{0:k-1}, u_{0:k-1}^a, d_{0:k}^a)$ denote the output of y_k as a function of the initial state, the defender's input, and the attacker's input. Since y_k is deterministic for all k, we have the following definition:

Definition 5.1. *A nonzero attack $u^a_{0:T-1}, d^a_{0:T}$ on a deterministic system (5.1) with controller (5.3) and known state x_0 is stealthy or undetectable up to time T if and only if*

$$y_k(x_0, u_{0:k-1}, u^a_{0:k-1}, d^a_{0:k}) = y_k(x_0, u_{0:k-1}, 0, 0), \quad 0 \leq k \leq T. \tag{5.4}$$

By leveraging the linearity of the system, we arrive at the following equivalent result, characterizing the set of stealthy attacks.

Theorem 5.1. *A nonzero attack $u^a_{0:T-1}, d^a_{0:T}$ on a deterministic system (5.1) with controller (5.3) and known state x_0 is stealthy up to time T if and only if, there exists $\delta x_0, \cdots, \delta x_T$ such that*

$$\delta x_{k+1} = A\delta x_k + B^a u^a_k, \quad 0 \leq k \leq T-1, \qquad \delta x_0 = 0, \tag{5.5}$$

$$0 = C\delta x_k + D^a d^a_k, \quad 0 \leq k \leq T. \tag{5.6}$$

Proof. From (5.4), there exists a nonzero stealthy attack if and only if there exists inputs $u^a_{0:T-1}, d^a_{0:T}$, that satisfy

$$\sum_{j=0}^{k-1} CA^{k-1-j}B^a u^a_j + D^a d^a_k = 0, \;\; 0 \leq k \leq T$$

This is true if and only if $0 = C\delta x_k + D^a d^a_k$ for $0 \leq k \leq T$, where $\delta x_k = \sum_{j=0}^{k-1} A^{k-1-j}B^a u^a_j$. The result immediately follows. □

Note that the stealthiness of an attacker's inputs is independent of the defender's control strategy in the deterministic case. We next consider attacks that are stealthy for all $k \geq 0$. We define a perfect attack as follows.

Definition 5.2. *A nonzero attack $\{u^a_k\}, \{d^a_k\}$ is perfect if it satisfies*

$$y_k(x_0, u_{0:k-1}, u^a_{0:k-1}, d^a_{0:k}) = y_k(x_0, u_{0:k-1}, 0, 0), \quad k \geq 0. \tag{5.7}$$

In other words, the set of perfect attacks is the set of all attacks in deterministic systems with known initial state that are stealthy for all time k. We can relate perfect attacks to the fundamental property of left invertibility.

Definition 5.3. *Consider a system defined by* (A, B, C, D)*, where*

$$x_{k+1} = Ax_k + Bu_k, \;\; y_k = Cx_k + Du_k,$$

A system is left invertible if $y_k = 0, \; k \geq 0$ *and* $x_0 = 0$ *implies that* $u_k = 0, \; k \geq 0$.

Fundamentally, the left invertibility of a system implies that there exists a unique input sequence generating every output sequence. This is formalized below.

Theorem 5.2. *There exists a perfect attack on the system defined in* (5.1) *if and only if the system* $(A, [B^a \; 0_{n\times m_*}], C, [0_{m\times p_*} \; D^a])$ *is not left invertible.*

The proof of this result is similar to the proof of Theorem 5.1 and is thus omitted. We see an attacker is able to stealthily perturb the system if and only if he can change the state without changing the output. The ability to change an input without changing the output requires that the system not be left invertible.

The conditions for analyzing the left invertibility of a system can be analyzed by looking at the matrix pencil. In particular, we have the following based on results in [65] [Corollary 8.10].

Corollary 5.1. *There exists no perfect attack on the system defined in* (5.1) *if and only if for all but finitely many* $\lambda \in \mathbb{C}$*, we have*

$$\operatorname{rank}\left(\begin{bmatrix} \lambda I - A & -B^a & 0_{n\times m_*} \\ C & 0_{m\times p_*} & D^a \end{bmatrix}\right) = n + m_* + p_*. \tag{5.8}$$

The existence of perfect attacks can also be described graphically by considering the underlying structure of the inputs, outputs, and state variables. This will be revisited later in the chapter. The set of stealthy attacks in deterministic control systems can be increased if the defender is unaware of the initial state. We assume now that the defender's control strategy satisfies

$$u_k = \mathcal{U}_k(A, B, C, u_{0:k-1}, y_{0:k}), \tag{5.9}$$

where $\mathcal{U}_k$ is some deterministic function. It can be inductively shown that a system with the same output history will have the same input history. We can consequently define a stealthy or undetectable attack as follows.

Definition 5.4. *A nonzero attack $u^a_{0:T-1}, d^a_{0:T}$ on a deterministic system (5.1) with controller (5.9) and unknown state x_0 is stealthy or undetectable up to time T if and only if*

$$y_k(x_0, u_{0:k-1}, u^a_{0:k-1}, d^a_{0:k}) = y_k(x'_0, u_{0:k-1}, 0, 0), \quad 0 \le k \le T. \tag{5.10}$$

for some $x'_0 \in \mathbb{R}^n$. We refer to a nonzero attack that is stealthy for all time $k \ge 0$ as a zero dynamics attack.

Theorem 5.3. *A nonzero attack $u^a_{0:T-1}, d^a_{0:T}$ on a deterministic system (5.1) with controller (5.9) and unknown state x_0 is stealthy up to time T if and only if, there exists $\delta x_0, \cdots, \delta x_T$ such that*

$$\delta x_{k+1} = A\delta x_k + B^a u^a_k, \quad 0 \le k \le T-1, \qquad \delta x_0 \in \mathbb{R}^n, \tag{5.11}$$

$$0 = C\delta x_k + D^a d^a_k, \quad 0 \le k \le T. \tag{5.12}$$

Proof. From (5.10), there exists a nonzero stealthy attack if and only if there exists inputs $u^a_{0:T-1}, d^a_{0:T}$, that satisfy

$$CA^k(x_0 - x'_0) + \sum_{j=0}^{k-1} CA^{k-1-j}B^a u^a_j + D^a d^a_k = 0, \;\; 0 \le k \le T$$

This is true if and only if $0 = C\delta x_k + D^a d^a_k$ for $0 \le k \le T$, where $\delta x_k = A^k(x_0 - x'_0) + \sum_{j=0}^{k-1} A^{k-1-j}B^a u^a_j$. Since x'_0 is arbitrary, the result immediately follows. □

We remark that perfect attacks are a subclass of zero dynamics attacks. In practice, a defender may have some imperfect information about x_0. Thus, x'_0 must be chosen carefully to avoid an alarm. The existence of zero dynamics attacks is related to the strong observability of a system.

Definition 5.5. *Consider a system defined by (A, B, C, D), where*

$$x_{k+1} = Ax_k + Bu_k, \;\; y_k = Cx_k + Du_k,$$

A system is strongly observable if $y_k = 0, \; k \ge 0$ implies that $x_0 = 0$.

We now aim to characterize systems that are vulnerable to zero dynamics attacks. We have the following result.

Theorem 5.4. *Suppose (A, C) is observable. The following statements are equivalent.*

1. *There exists no zero dynamics attack on system (5.1).*

2. *There exists no nonzero inputs $\{u_k^a\}, \{d_k^a\}$ satisfying*

$$\delta x_{k+1} = A\delta x_k + B^a u_k^a, \;\; 0 = C\delta x_k + D^a d_k^a, \;\; \delta x_0 \in \mathbb{R}^n \;\; k \geq 0. \tag{5.13}$$

3. *$(A, [B^a \; 0_{n\times m_*}], C, [0_{m\times p_*} \; D^a])$ is strongly observable and left invertible.*

4. $\text{rank}\left(\begin{bmatrix} \lambda I - A & -B^a & 0_{n\times m_*} \\ C & 0_{m\times p_*} & D^a \end{bmatrix}\right) = n + m_* + p_*, \; \forall\, \lambda \in \mathbb{C}$

Proof. Statement 1 is equivalent to the absence of nonzero inputs such that

$$CA^k x_0 + D^a d_k^a + \sum_{j=0}^{k-1} CA^{k-1-j}(Bu_j + B^a u_j^a) = CA^k x_0' + \sum_{j=0}^{k-1} CA^{k-1-j} Bu_j, \;\; k \geq 0.$$

Since x_0' is arbitrary, statement 1 is equivalent to the absence of nonzero inputs and $\delta x_0 \in \mathbb{R}^n$ such that

$$CA^k \delta x_0 + D^a d_k^a + \sum_{j=0}^{k-1} CA^{k-1-j} B^a u_j^a = 0, \;\; k \geq 0.$$

This is in fact equivalent to statement 2. Statement 2 implies that if the output is $C\delta x_k + D^a d_k^a = 0$ for all k, the attack inputs must be identically 0. This implies left invertibility since the initial state is never specified. Since the system is observable, this implies $\delta x_0 = 0$, which implies strong observability. As a result, statement 2 implies statement 3. Moreover, strong observability in statement 3 implies that if $C\delta x_k + D^a d_k^a = 0$ in (5.13) for all k, $\delta x_0 = 0$. Left invertibility in statement 3 would then imply the inputs are necessarily 0. Thus statement 2 and 3 are equivalent. Finally statement 3 and 4 are equivalent due to Theorem 7.17 and Corollary 8.10 in [65]. □

We remark that if B^a and D^a each have full column rank (as currently constructed), then strong observability will imply the left invertibility of a system. We now wish to assess the impact of zero dynamics attacks. The true impact of the attack is dependent on the control strategy $\mathcal{U}_k$. For our purposes, we assume the defender's goal is to stabilize the system at 0. This can be accomplished even if x_0 is unknown if (A, B) is stabilizable and (A, C) is detectable by using state feedback and a stable observer.

Assume $y_k(x_0, u_{0:k-1}, u^a_{0:k-1}, d^a_{0:k}) = y_k(x'_0, u_{0:k-1}, 0, 0)$ for all $k \geq 0$. Let $x_k(x_0, u_{0:k-1}, u^a_{0:k-1}, d^a_{0:k-1})$ denote the state x_k generated by (5.1) as a function of the initial state, the defender's input, and the attacker's inputs. Under attack $x_k = x_k(x_0, u_{0:k-1}, u^a_{0:k-1}, d^a_{0:k-1})$. The defender, however has designed his feedback control inputs $u_{0:k-1}$ so that he stabilizes a system with initial state x'_0.

In this case, we make the assumption that

$$\lim_{k\to\infty} x_k(x'_0, u_{0:k-1}, 0, 0) = 0. \tag{5.14}$$

By the linearity of the system we see that

$$x_k(x_0, u_{0:k-1}, u^a_{0:k-1}, d^a_{0:k-1}) = x_k(x'_0, u_{0:k-1}, 0, 0) + x_k(x_0 - x'_0, 0, u^a_{0:k-1}, d^a_{0:k-1}). \tag{5.15}$$

Thus, if the attacker's goal is to destabilize a control system, he may wish to maximize $\|x_k(x_0 - x'_0, 0, u^a_{0:k-1}, d^a_{0:k-1})\|_2$. Using linearity, we can show that $0 = y_k(x_0 - x'_0, 0, u^a_{0:k-1}, d^a_{0:k-1})$. As a result, the attackers perturbations on the state x_k in our CPS can be approximately described by the dynamics of δx_k in (5.13). To understand the dynamics of δx_k we define the weakly unobservable subspace.

Definition 5.6. *The weakly unobservable subspace $\mathcal{V}_u(A, B^a, C, D^a)$ is the set of $\delta x_0 \in \mathbb{R}^n$ for which there exists $\{u^a_k\}, \{d^a_k\}$ which allow (5.13) to hold.*

It can be shown that $\delta x_k \in \mathcal{V}_u(A, B^a, C, D^a)$ for all $k \geq 0$. Moreover, we have the following result from [65][Theorem 7.10] characterizing the weakly unobservable subspace.

Lemma 5.1. *$\mathcal{V}_u(A, B^a, C, D^a)$ is the largest subspace of $\mathbb{R}^n$ for which there exists linear maps $F_1 \in \mathbb{R}^{p_* \times n}$ and $F_2 \in \mathbb{R}^{m_* \times n}$ satisfying*

$$(A + B^a F_1)\mathcal{V}_u \subset \mathcal{V}_u, \;\; (C + D^a F_2)\mathcal{V}_u = 0 \tag{5.16}$$

Methods to compute $\mathcal{V}_u$ as well as (non-unique) matrices F_1 and F_2 are provided in [65]. We can now describe the class of input strategies that allow an attacker to remain stealthy. To begin we define the subspace $(B^a)^{-1}\mathcal{V}_u = \{u \in \mathbb{R}^{p_*} | B^a u \in \mathcal{V}_u\}$. Moreover let L_1, L_2 be a linear maps such that $\text{Im}(L_1) = (B^a)^{-1}\mathcal{V}_u$ and $\text{Im}(L_2) = \text{Ker}(D^a)$ We have the following result based on the characterization of inputs exciting a system's zero dynamics found in [65] [Theorem 7.11].

Theorem 5.5. *An attack $\{u_k^a\}, \{d_k^a\}$ satisfies (5.13) if and only if $\delta x_0 \in \mathcal{V}_u$ and*

$$u_k^a = F_1 \delta x_k + L_1 \omega_k^1, \;\; d_k^a = F_2 \delta x_k + L_2 \omega_k^2 \tag{5.17}$$

where $\{\omega_k^1\}$ and $\{\omega_k^2\}$ are arbitrary sequences of real inputs of the proper dimension and F_1, F_2 satisfy (5.16).

We remark that since D^a is full column rank, in practice L_2 is an empty or zero matrix. We can see that δx_k can be expressed as

$$\delta x_k = (A + B^a F_1)^k \delta x_0 + \sum_{j=0}^{k-1} (A + B^a F_1)^{k-1-j} B^a L_1 \omega_j^1. \tag{5.18}$$

If the system is not left invertible the attacker is further restricted. In particular, we have the following result.

Corollary 5.2. *Suppose $(A, [B^a \; 0_{n \times m_*}], C, [0_{m \times p_*} \; D^a])$ is left invertible and that B^a and D^a are full column rank. Then $\{u_k^a\}, \{d_k^a\}$ satisfies (5.13) if and only if*

$$u_k^a = F_1 \delta x_k, \;\; d_k^a = F_2 \delta x_k. \tag{5.19}$$

Proof. Suppose the system is left invertible. We argue there is no nonzero input u^* such that $B^a u^* \in \mathcal{V}^u$. If there was such a u^*, we could let $d_0^a = 0, u_0^a = u^*$ and $\delta x_0 = 0$. The resulting

δx_1 is nonzero (since B^a is full column rank) and is in $\mathcal{V}^u$. Consequently, there is a nonzero input sequence that would force $C\delta x_k + D^a d_k^a = 0$ for all $k \geq 0$ with $\delta x_0 = 0$. This would contradict left invertibility. Thus, $\text{Im}(L_1) = 0$. Since D^a is full column rank, $\text{Im}(L_2) = 0$. As such (5.19) holds. Moreover, from Theorem 5.5, (5.19) implies that (5.13) holds. The result follows. □

Note, that if a system is left invertible, but has nontrivial zero dynamics, the adversary's entire attack sequence will be trajectory that is a deterministic function of his chosen perturbation δx_0. Indeed, if a system is left invertible, but not strongly observable, δx_k can be expressed as

$$\delta x_k = (A + B^a F_1)^k \delta x_0. \tag{5.20}$$

On the other hand, if the system is not left invertible, the attacker would instead be able to excite specific controllable subspaces. As such, we can see that an attacker can stealthily destabilize a system if the system is not left invertible. However, if the system is left invertible, with nontrivial zero dynamics, the attacker's ability to act on the system will depend on the stability of the zero dynamics as we will next see.

Theorem 5.6. *Suppose* $(A, [B^a\ 0_{n\times m_*}], C, [0_{m\times p_*}\ D^a])$ *is not left invertible and that* B^a *and* D^a *are full column rank. Then there exists inputs* $\{u_k^a\}, \{d_k^a\}$ *satisfying* (5.13) *while* $\limsup_{k\to\infty} \|\delta x_k\|_2 = \infty$. *Now suppose is* $(A, [B^a\ 0_{n\times m_*}], C, [0_{m\times p_*}\ D^a])$ *left invertible but not strongly observable. Then there exists inputs* $\{u_k^a\}, \{d_k^a\}$ *satisfying* (5.13) *while* $\limsup_{k\to\infty} \|\delta x_k\|_2 = \infty$ *if and only if there exists* $v \in \mathcal{V}_u$ *satisfying* $\limsup_{k\to\infty} \|(A + B^a F_1)^k v\|_2 = \infty$.

Proof. If the system is not left invertible, there exists a nonzero input u^* such that $B^a u^* \in \mathcal{V}^u$. If not, then for a system satisfying (5.13), we have $\delta x_k = 0$ for all k when $\delta x_0 = 0$. This also implies $\{d_k^a\}$ and $\{u_k^a\}$ are 0, which is a contradiction. Thus L_1 is nonzero and an attacker is able to perturb δx_k along the controllable subspace of $(A + B^a F_1, B^a L_1)$, which is nonzero. As such the attacker can destabilize δx_k. If the system is left invertible but not strongly observable. Then $\delta x_k = (A + B^a F_1)^k \delta x_0$. As such, δx_k can be destabilized if and only if there exists $v \in \mathcal{V}_u$ satisfying $\limsup_{k\to\infty} \|(A + B^a F_1)^k v\|_2 = \infty$. The result follows. □

The resources required by a zero dynamics attacker is also evident from Theorem 5.5 and Lemma 5.1. In particular, the attacker's system knowledge must include (A, B^a, C, D^a). The adversary, furthermore requires disruption resources to insert an attack along B^a and D^a. Finally, if an attacker can introduce additive perturbations, he or she will require no disclosure resources. If additive perturbations are impossible, then the attacker will need to be able to read the inputs and outputs of the actuators and sensors he chooses to modify. Fig. 5.1 illustrates the attack space with the zero dynamics attack. We distinguish between the scenario where an attacker is able to insert additive signals without reading the associated measurements and inputs and the scenario where an attacker must be able to read the appropriate channels.

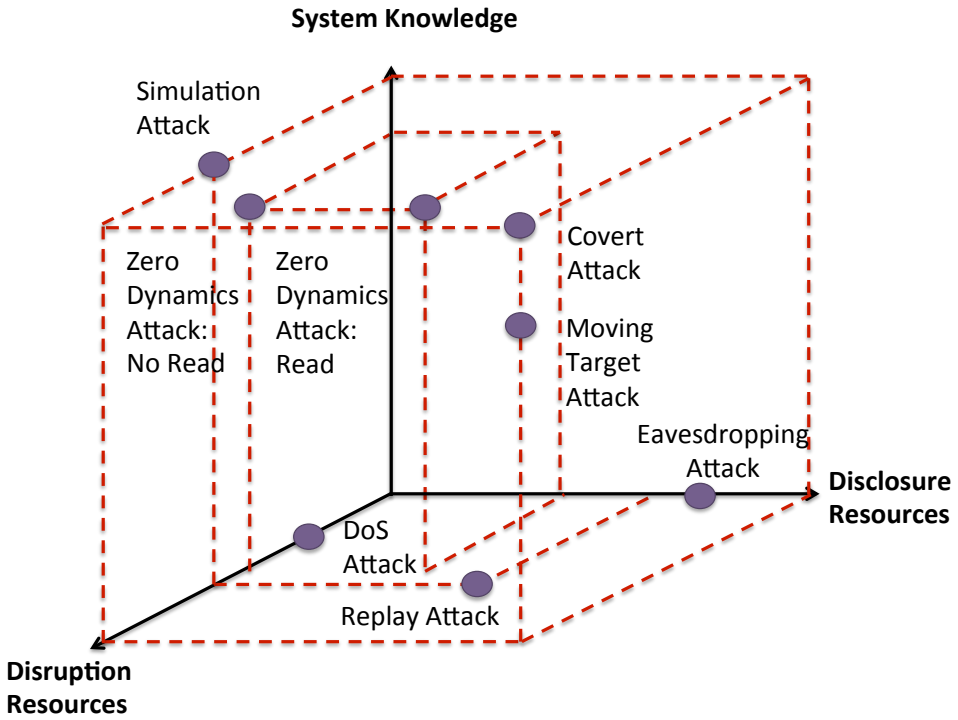


Figure 5.1: Cyber-Physical Attack Space with Zero Dynamics Attack

5.1.2 Detection in Stochastic Systems

Consider a stochastic system under integrity attacks.

$$x_{k+1} = Ax_k + Bu_k + B^a u_k^a + w_k, \;\; y_k = Cx_k + D^a d_k^a + v_k. \tag{5.21}$$

We assume the process noise $w_k \in \mathbb{R}^n$ and sensor noise $v_k \in \mathbb{R}^m$ are IID and independent of each other with $w_k \sim \mathcal{N}(0, Q)$ and $v_k \sim \mathcal{N}(0, R)$, We assume that $R > 0$, (A, C) is detectable, and $(A, Q^{\frac{1}{2}})$ is stabilizable. Moreover, x_0 is independent of the noise processes and has distribution $\mathcal{N}(\bar{x}_{0|-1}, \Sigma)$.

Estimation

We obtain a minimum mean squared error estimate by using a Kalman filter as follows:

$$\hat{x}_{k+1|k} = A\hat{x}_{k|k} + Bu_k, \; \hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k z_k, \tag{5.22}$$

$$z_k = y_k - C\hat{x}_{k|k-1}, K_k = P_{k|k-1}C^T(CP_{k|k-1}C^T + R)^{-1}, \tag{5.23}$$

$$P_{k|k} = P_{k|k-1} - K_k C P_{k|k-1}, \;\; P_{k+1|k} = AP_{k|k}A^T + Q, \tag{5.24}$$

where we define

$$\hat{x}_{k|k} \triangleq \mathbb{E}[x_k | y_{0:k}], \;\; \hat{x}_{k|k-1} \triangleq \mathbb{E}[x_k | y_{0:k-1}], \tag{5.25}$$

$$P_{k|k} \triangleq \mathbb{E}[e_k e_k^T | y_{0:k}], \;\; e_k \triangleq x_k - \hat{x}_{k|k}, \tag{5.26}$$

$$P_{k|k-1} \triangleq \mathbb{E}[e_{k|k-1} e_{k|k-1}^T | y_{0:k-1}], \;\; e_{k|k-1} \triangleq x_k - \hat{x}_{k|k-1}. \tag{5.27}$$

We observe that $P_{k|k-1}$ and K_k converge to unique matrices, which we define as P and K respectively. Assuming that the system has been running for a long time, we assume $P_{k|k-1} = P$ and $K_k = K$ for all k. Thus, here $\Sigma = P$.

We examine the effect of zero dynamics attacks on the stochastic control system. (5.21). In the case of a perfect attack, we can show that an adversary remains stealthy. Specifically, we have the following.

Theorem 5.7. *Suppose an attacker performs a perfect attack on (5.21). Moreover, assume the defender's control policy at time k is a deterministic function of $\mathcal{I}_k = \{\mathcal{M}, y_{0:k}, u_{0:k-1}, \hat{x}_{-1|0}\}$. Then the probability distribution of y_k under attack $f(y_k|\mathcal{H}_1)$ is equal to the probability distribution of y_k under normal operation $f(y_k|\mathcal{H}_0)$.*

Proof. Let $y_k(x_0, \hat{x}_{-1|0}, u_{0:k-1}, u^a_{0:k-1}, d^a_{0:k-1}, w_{0:k-1}, v_{0:k})$ denote the output as a function of the initial states, the defender's and attacker's inputs, and the noise sequences. From the properties of a perfect attack and the linearity of the system

$$y_k(x_0, \hat{x}_{-1|0}, u_{0:k-1}, u^a_{0:k-1}, d^a_{0:k-1}, w_{0:k-1}, v_{0:k}) = y_k(x_0, \hat{x}_{-1|0}, u_{0:k-1}, 0, 0, w_{0:k-1}, v_{0:k}).$$

Using the fact u_k is a deterministic function of $\mathcal{I}_k$, we can inductively show the sequence of control inputs remains the same both in the presence and absence of an attack. The result follows. □

To understand the impact of general zero dynamics, we examine attacker's effect on the residue z_k. Note under attack (5.21) applies to the system dynamics. The Kalman filter equations (5.22),(5.23),(5.24) are unchanged (though we assume K_k and $P_{k|k-1}$ have converge to K and P). Let $z_k(e_{0|-1}, v_{0:k}, w_{0:k-1}, u^a_{0:k-1}, d^a_{0:k})$ be the residue z_k generated from (5.21),(5.22),(5.23),(5.24) due to the initial state estimation error, the sensor noise, the process noise, and the attacker inputs. The attacker's bias on the residues is given by

$$\Delta z_k \triangleq z_k(e_{0|-1}, v_{0:k}, w_{0:k-1}, u^a_{0:k-1}, d^a_{0:k}) - z_k(e_{0|-1}, v_{0:k}, w_{0:k-1}, 0, 0). \tag{5.28}$$

We arrive at the ensuing result.

Theorem 5.8. *Suppose an attacker performs a zero dynamics attack on (5.21). Then, we have*

$$\Delta z_k = -C(A - AKC)^k \delta x_0. \tag{5.29}$$

Proof. We begin with the following Lemma.

Lemma 5.2. *For all $k \geq 0$,*

$$A^k - \sum_{j=0}^{k-1} A^{k-j} KC(A - AKC)^j = (A - AKC)^k. \tag{5.30}$$

Proof. We prove by induction. For *Case* $k = 0$: Both the left and right side of (5.30) are equal to the identity matrix.

For *Case* $k = t$: We assume (5.30) holds for $k = t$.

Case $k = t + 1$. We observe that

$$\begin{aligned}
&A^{t+1} - \sum_{j=0}^{t} A^{t+1-j} KC(A - AKC)^j, \\
&= A\left(A^t - \sum_{j=0}^{t-1} A^{t-j} KC(A - AKC)^j\right) - AKC(A - AKC)^t, \\
&= A(A - AKC)^t - AKC(A - AKC)^t, \\
&= (A - AKC)^{t+1}.
\end{aligned}$$

Thus, by induction the assertion holds. □

Under normal operation, we have

$$e_k = (A - KCA)e_{k-1} + (I - KC)w_{k-1} - Kv_k, \;\; z_k = CAe_{k-1} + Cw_{k-1} + v_k.$$

Under attack,

$$\begin{aligned}
e_k = (A - KCA)e_{k-1} + (I - KC)w_{k-1} - Kv_k + (I - KC)B^a u^a_{k-1} - KD^a d^a_k, \\
z_k = CAe_{k-1} + Cw_{k-1} + CB^a u^a_{k-1} + v_k + D^a d^a_k.
\end{aligned}$$

Define Δe_k as

$$\Delta e_k \triangleq e_k(e_{0|-1}, w_{0:k-1}, v_{0:k}, u^a_{0:k-1}, d^a_{0:k}) - e_k(e_{0|-1}, w_{0:k-1}, v_{0:k}, 0, 0).$$

As a result, we have that

$$\begin{aligned}
\Delta e_k = (A - KCA)\Delta e_{k-1} + (I - KC)B^a u^a_{k-1} - KD^a d^a_k, \\
\Delta z_k = CA\Delta e_{k-1} + CB^a u^a_{k-1} + D^a d^a_k,
\end{aligned}$$

where $\Delta e_{-1} = 0$. Rearranging terms we have

$$\Delta e_k = A\Delta e_{k-1} + B^a u^a_{k-1} - K\Delta z_k, \;\; \Delta z_k = CA\Delta e_{k-1} + CB^a u^a_{k-1} + D^a d^a_k.$$

As a result, an inductive argument can be used to show

$$\Delta e_{k-1} = \sum_{j=0}^{k-2} A^{k-2-j} B^a u_j^a - \sum_{j=0}^{k-1} A^{k-1-j} K \Delta z_j,$$

We then have

$$\begin{aligned}\Delta z_k &= \sum_{j=0}^{k-1} C A^{k-1-j} B^a u_j^a + D^a d_k^a - \sum_{j=0}^{k-1} C A^{k-j} K \Delta z_j, \\ &= -C A^k \delta x_0 - \sum_{j=0}^{k-1} C A^{k-j} K \Delta z_j. \end{aligned} \tag{5.31}$$

We now prove the main assertion now through induction.

Case $k = 0$. From (5.31), $\Delta z_0 = -C\delta x_0$.

Case $k = t$. We assume $\Delta z_j = -C(A - AKC)^j \delta x_0$ for $j \leq t$.

Case $k = t + 1$. From (5.31) and Lemma 5.2 we have

$$\begin{aligned}\Delta z_{t+1} &= -C A^{t+1} \delta x_0 + C \sum_{j=0}^{t} A^{t+1-j} K C (A - AKC)^j \delta x_0, \\ &= -C(A - AKC)^{t+1} \delta x_0.\end{aligned}$$

This proves the main assertion.

□

From the stability of the Kalman filter the bias on the residue Δz_k asymptotically approaches 0. In this case, we see that an attacker will be asymptotically stealthy against a χ^2 detector. The prior result also applies to alternative continuous residue based detectors with finite memory. We will soon demonstrate that small values of Δz_k fundamentally lead to poor detection performance. Using the same rationale as in the deterministic case, the impact of a zero dynamics attack on the state x_k in a stochastic system can be characterized using the state δx_k in (5.13). Specifically, we see that

$$\begin{aligned}&x_k(x_0, u_{0:k-1}, w_{0:k-1}, v_{0:k-1}, u^a_{0:k-1}, d^a_{0:k-1}) \\ &= x_k(x'_0, u_{0:k-1}, w_{0:k-1}, v_{0:k-1}, 0, 0) + x_k(x_0 - x'_0, 0, 0, 0, u^a_{0:k-1}, d^a_{0:k-1}).\end{aligned}$$

If the control strategy allows for $\lim_{k\to\infty} \mathbb{E}[x_k(x'_0, u_{0:k-1}, w_{0:k-1}, v_{0:k-1}, 0, 0)] = 0$, then we have

$$\lim_{k\to\infty} \mathbb{E}[x_k(x_0, u_{0:k-1}, w_{0:k-1}, v_{0:k-1}, u^a_{0:k-1}, d^a_{0:k-1})] - x_k(x_0 - x'_0, 0, 0, 0, u^a_{0:k-1}, d^a_{0:k-1}) = 0.$$

Consequently, in this case, the zero dynamics capture the expected asymptotic state trajectory.

Finally, we show that the presence of zero dynamics is necessary to secretly destabilize a noisy system. Consider the error and residue bias as defined earlier. The attacker designs his input sequence so that

$$\sqrt{\Delta z_k^T (CPC^T + R)^{-1} \Delta z_k} \leq \underline{B}, \quad \forall\, k \geq 0 \tag{5.32}$$

where we assume an attack begins at $k = 0$, Δz_k is defined in (5.28) and $\underline{B}$ is some chosen bound for the attacker. We remark that due to the stochastic nature of a system, an attacker practically does need not choose the bound $\underline{B} = 0$ to remain hidden, as long as the perturbations introduced in the measurements are within the uncertainty of the system.

Theorem 5.9. *Consider a false data injection attack $\{u^a_k\}, \{d^a_k\}$. There exists a feasible attack input sequence satisfying (5.32), which destabilizes Δe_k so that $\limsup_{k\to\infty} \|\Delta e_k\|_2 = \infty$ only if there exists a real matrix L^a and vector $v \in \mathbb{R}^n$ satisfying*

1. *$Cv \in$ Im(D^a),*
2. *v is an eigenvector of $(A + B^a L^a)$.*

Proof. Let

$$\Delta e_{k|k-1} \triangleq e_{k|k-1}(e_{0|-1}, w_{0:k-1}, v_{0:k-1}, u^a_{0:k-1}, d^a_{0:k-1}) - e_{k|k-1}(e_{0|-1}, w_{0:k-1}, v_{0:k-1}, 0, 0).$$

Observe that

$$\Delta e_{k+1|k} = (A - AKC)\Delta e_{k|k-1} + B^a u^a_k - AKD^a d^a_k, \;\; \Delta z_k = C\Delta e_{k|k-1} + D^a d^a_k.$$

From Lemma 1 in [66], there exists feasible actions with unbounded $\Delta e_{k|k-1}$ only if there exists $v \in \mathbb{R}^n$ satisfying

1. $(C + D^a L_2)v = 0$,

2. v is an eigenvector of $A - AKC + B^a L^a - AKD^a L_2$

for arbitrary real matrices L^a and L_2. Note that $(A - AKC + B^a L^a - AKD^a L_2)v = \lambda v$ implies $(A + B^a L^a)v = \lambda v$ since $(C + D^a L_2)v = 0$. Moreover, $(C + D^a L_2)v = 0$ implies Cv is in the image of D^a. Note under normal operation $\hat{x}_{k|k} = (I - KC)\hat{x}_{k|k-1} + KCx_k + Kv_k$ while under attack $\hat{x}_{k|k} = (I - KC)\hat{x}_{k|k-1} + KCx_k + Kv_k + KD^a d_k^a$. Thus,

$$\Delta e_k = \Delta e_{k|k-1} - K\Delta z_k$$

As a result, for a feasible attack sequence, $\Delta e_{k|k-1}$ is unbounded if and only if Δe_k is unbounded. The result immediately follows. □

We can leverage the prior theorem to relate the existence of zero dynamics attacks to destabilizing integrity attacks in the following result.

Corollary 5.3. *Consider a false data injection attack. Suppose (A, C) is observable. There exists a feasible attack input sequence satisfying (5.32), which destabilizes Δe_k so that $\limsup_{k\to\infty} \|\Delta e_k\|_2 = \infty$ only if there exists a zero dynamics attack.*

Proof. Theorem 5.9 implies the existence of matrices F_1 and F_2 and nonzero vector $v \in \mathbb{R}^n$ such that $(A + B^a F_1)v = \lambda v$ and $(C + D^a F_2)v = 0$. From Lemma 5.1, this implies that the weakly unobservable subspace $\mathcal{V}_u$ has nonzero dimension. Since (A, C) is observable this in turn implies the existence of zero dynamics attacks. □

5.1.3 Identification and Estimation in Deterministic Systems

We conclude our study of zero dynamics attacks, by relating such attacks to the class of unidentifiable attacks in control systems. We assume an adversary is unable to insert their own actuators. Suppose an attacker targets actuators $\mathcal{K}_u^a = \{\delta_1, \cdots, \delta_{p_*}\} \subset \{1, \cdots, p\}$ and sensors $\mathcal{K}_y^a = \{\eta_1, \cdots, \eta_{m_*}\} \subset \{p+1, \cdots, p+m\}$. To write the corresponding B^a and D^a uniquely

as a function of their attack set we, without loss of generality, assume all attack sets are given in ascending order. Here, $B^a(\mathcal{K}_u^a) = \begin{bmatrix} B_{\delta_1} & \cdots & B_{\delta_{p_*}} \end{bmatrix}$ where B_{δ_i} is the δ_ith column of B. $D^a(\mathcal{K}_y^a)$ can be obtained entrywise as follows $D^a(s,t) \triangleq \mathbb{I}_{s=\eta_i-p,t=i}$. We assume that if a sensor or actuator is targeted in a window $0 \leq k \leq T$, its value has been modified by an attacker at least once during this time frame.

We let $B^a(\mathcal{K})u_{0:k}^a = \{B^a(\mathcal{K})u_0^a, \cdots, B^a(\mathcal{K})u_k^a\}$. Similarly, we have $D^a(\mathcal{K})d_{0:k}^a = \{D^a(\mathcal{K})d_0^a, \cdots, D^a(\mathcal{K})d_k^a\}$. Roughly speaking, we say an attack is unidentifiable, if there exists an attack targeting a different (but possibly intersecting) set of nodes with size less than or equal to the original attack set. In other words, the nodes an adversary targets provides the unique simplest explanation of an attack. Similar to the notion of identifiability in [44], we have the following definition.

Definition 5.7. *An attack input $B^a(\mathcal{K}_u)u_{0:T-1}^a$, $D^a(\mathcal{K}_y)d_{0:T}^a$ on a deterministic system (5.1) with controller (5.9) and unknown state x_0 is* unidentifiable *up to time T if and only if*

1. *there exists sets $\mathcal{K}_u' \subset \{1, \cdots, p\}$ and $\mathcal{K}_y' \subset \{p+1, \cdots, p+m\}$ with $\mathcal{K}_u \neq \mathcal{K}_u'$ or $\mathcal{K}_y \neq \mathcal{K}_y'$*
2. *$|\mathcal{K}_u'| + |\mathcal{K}_y'| \leq |\mathcal{K}_u| + |\mathcal{K}_y|$.*
3. *there exists $x_0' \in \mathbb{R}^n$ and inputs $\bar{u}_{0:T-1}^a, \bar{d}_{0:T}^a$ satisfying.*

$$y_k(x_0, u_{0:k-1}, B^a(\mathcal{K}_u)u_{0:k-1}^a, D^a(\mathcal{K}_y)d_{0:k}^a) = y_k(x_0', u_{0:k-1}, B^a(\mathcal{K}_u')\bar{u}_{0:k-1}^a, D^a(\mathcal{K}_y')\bar{d}_{0:k}^a), \tag{5.33}$$

for $0 \leq k \leq T$.

We assume every sensor in $\mathcal{K}_y$ is attacked at least once given input $D^a(\mathcal{K}_y)d_{0:T}^a$. We assume every actuator in $\mathcal{K}_u$ is attacked at least once given $B^a(\mathcal{K}_u)u_{0:T-1}^a$. Likewise we assume every sensor in $\mathcal{K}_y'$ and every actuator in $\mathcal{K}_u'$ is attacked at least once given $D^a(\mathcal{K}_y')\bar{d}_{0:T}^a$ and $B^a(\mathcal{K}_u')\bar{u}_{0:k-1}^a$.

Additionally, we say attack set $\mathcal{K}_u \cup \mathcal{K}_y$ is unidentifiable if there exists an attack input targeting these nodes which is unidentifiable up to time $T = \infty$. Otherwise we say $\mathcal{K}_u \cup \mathcal{K}_y$ is identifiable.

To be explicit here, when we write y_k as a function, we must specify the set of attacked sensors and inputs. We can easily see that undetectable attacks are also unidentifiable as the attack input can be mistaken for a 0 attack. The class of unidentifiable attack inputs is closely related to the class of zero dynamics attacks. For instance, we have the following result.

Theorem 5.10. *There exist an unidentifiable attack set of size q or less if and only if there exists a zero dynamics attacks on a set of $2q$ or fewer actuators or sensors.*

Proof. Suppose $\mathcal{K} = \mathcal{K}_u \cup \mathcal{K}_y$ is an unidentifiable attack set with $|\mathcal{K}| \leq q$ and $\mathcal{K}_u \subset \{1, \cdots, p\}$ and $\mathcal{K}_y \subset \{p+1, \cdots, p+m\}$. Then, there exists $\mathcal{K}' = \mathcal{K}'_u \cup \mathcal{K}'_y$ with $|\mathcal{K}'| \leq |\mathcal{K}|$, $\mathcal{K}' \neq \mathcal{K}$, $\mathcal{K}'_u \subset \{1, \cdots, p\}$ and $\mathcal{K}'_y \subset \{p+1, \cdots, p+m\}$ satisfying (5.33) for all $k \geq 0$. It is assumed that for each entry $j \in \{1, \cdots, |\mathcal{K}'_u|\}$ there exists $k \geq 0$ satisfying $\bar{u}^a_k(j) \neq 0$, where $\bar{u}^a_k(j)$ is the jth entry of $\bar{u}^a_k(j)$. The assumption also applies to $\{\bar{d}^a_k\}$. This implies the existence of a sequence of states $\{\delta x_k\}$, and nonzero input sequence $\{\tilde{u}^a_k\}, \{\tilde{d}^a_k\}$

$$\delta x_{k+1} = A\delta x_k + B^a(\mathcal{K}_u \cup \mathcal{K}'_u)\tilde{u}^a_k, \;\; 0 = C\delta x_k + D^a(\mathcal{K}_y \cup \mathcal{K}'_y)\tilde{d}^a_k. \tag{5.34}$$

The input sequence is nonzero since $\mathcal{K}' \neq \mathcal{K}$ and all sensors and actuators are attacked. Thus, there exists a zero dynamics attack on a set of $2q$ or fewer actuators or sensors. Now suppose there is a zero dynamics attack on a set of $2q$ or fewer nodes $\mathcal{K}_*$. Assume, without loss of generality that all nodes are attacked. In addition, without loss of generality assume $\mathcal{K}_* = \mathcal{K} \cup \mathcal{K}'$ where $\mathcal{K} = \mathcal{K}_u \cup \mathcal{K}_y, \mathcal{K}' = \mathcal{K}'_u \cup \mathcal{K}'_y, \mathcal{K}'_u, \mathcal{K}_u \subset \{1, \cdots, p\}$, and $\mathcal{K}'_y, \mathcal{K}_y \subset \{p+1, \cdots, p+m\}$. Moreover, without loss of generality, assume $\mathcal{K} \leq q$, $\mathcal{K}' \leq q$, $\mathcal{K} \cap \mathcal{K}' = \emptyset$, and $|\mathcal{K}'| \leq |\mathcal{K}|$. We know there exists a zero dynamics attack $\{u^a_k\}$,$\{\bar{u}^a_k\}$,$\{d^a_k\}$,$\{\bar{d}^a_k\}$, with each node being attacked satisfying

$$\delta x_{k+1} = A\delta x_k + B^a(\mathcal{K}_u)u^a_k - B^a(\mathcal{K}'_u)\bar{u}^a_k, \tag{5.35}$$

$$0 = C\delta x_k + D^a(\mathcal{K}_y)d^a_k - D^a(\mathcal{K}'_y)\bar{d}^a_k. \tag{5.36}$$

Thus, for all $k \geq 0$, we have an attack sequence $\{\bar{u}^a_k\}$,$\{d^a_k\}$ targeting all sensors and actuators in $\mathcal{K}$

satisfying

$$y_k(x_0, u_{0:k-1}, B^a(\mathcal{K}_u)u^a_{0:k-1}, D^a(\mathcal{K}_y)d^a_{0:k}) = y_k(x_0 - \delta x_0, u_{0:k-1}, B^a(\mathcal{K}'_u)\bar{u}^a_{0:k-1}, D^a(\mathcal{K}'_y)\bar{d}^a_{0:k})$$

□

As a result, preventing zero dynamics attacks coming from all sets of $2q$ sensors and actuators will simultaneously prevent unidentifiable attacks. This can be done by guaranteeing strong observability and left invertibility for all sets of $2q$ sensors and actuators.

Corollary 5.4. *Suppose (A, C) is observable. There exist no unidentifiable attack set of size q or less if and only if for all $\mathcal{K} = \mathcal{K}_u \cup \mathcal{K}_y$ satisfying $\mathcal{K}_u \subset \{1, \cdots, p\}$ and $\mathcal{K}_y \subset \{p+1, \cdots, p+m\}$ with $|\mathcal{K}| \leq 2q$, $\left(A, [B^a(\mathcal{K}_u) \;\; 0_{n\times|\mathcal{K}_y|}], C, [0_{m\times|\mathcal{K}_u|} \;\; D^a(\mathcal{K}_y)]\right)$ is strongly observable and left invertible.*

This result follows immediately from Theorem 5.4 and Theorem 5.10. We note that if B is not injective, this provides a path for an adversary to generate unidentifiable attacks. For instance, if redundant actuators are used and one or more are compromised, it would be impossible for a defender to determine which if any actuators are secure. While redundancy could compromise the ability to identify attacks, it does not affect the ability to perform resilient estimation.

Definition 5.8. *Suppose an attacker can target up to q sensors and actuators so that $|\mathcal{K}_u \cup \mathcal{K}_y| \leq q$. We say that a defender can uniquely recover the state x_j given $\{y_j, y_{j+1}, \cdots\}$ in the presence of attack input $\{B^a(\mathcal{K}_u)u^a_k\}$, $\{D^a(\mathcal{K}_y)d^a_k\}$ on a deterministic system (5.1) with controller (5.9) if there exists no $x'_j \in \mathbb{R}^n$ with $x'_j \neq x_j$ and sequences $\{B^a(\mathcal{K}'_u)\bar{u}^a_k\}$, $\{D^a(\mathcal{K}'_y)\bar{d}^a_k\}$ satisfying*

$$y_k(x_j, u_{j:k-1}, B^a(\mathcal{K}_u)u^a_{j:k-1}, D^a(\mathcal{K}_y)d^a_{j:k}) = y_k(x'_j, u_{j:k-1}, B^a(\mathcal{K}'_u)\bar{u}^a_{j:k-1}, D^a(\mathcal{K}'_y)\bar{d}^a_{j:k}), \quad k \geq j \tag{5.37}$$

where $|\mathcal{K}'_u \cup \mathcal{K}'_y| \leq |\mathcal{K}_u \cup \mathcal{K}_y|$. It is assumed all mentioned sensors and actuators are attacked at least once.

In other words, we state that a defender can recover x_j for a given attack sequence, if there is no other state x'_j and feasible set of attack inputs that can generate the same output sequence. Similar, to Corollary 5.4, we can characterize systems for which the initial state is always recoverable.

Theorem 5.11. *Suppose an attacker can target up to q sensors and actuators. A defender can recover the state x_j for all feasible attack sequences if and only if for all $\mathcal{K} = \mathcal{K}_u \cup \mathcal{K}_y$ satisfying $\mathcal{K}_u \subset \{1, \cdots, p\}$ and $\mathcal{K}_y \subset \{p+1, \cdots, p+m\}$ with $|\mathcal{K}| \leq 2q$, we have $\big(A, [B^a(\mathcal{K}_u) \;\; 0_{n\times|\mathcal{K}_y|}], C, [0_{m\times|\mathcal{K}_u|} \;\; D^a(\mathcal{K}_y)]\big)$ is strongly observable.*

Proof. Without loss of generality let $j = 0$. Suppose x_0 can not be recovered given q sensor and actuator attacks. Then there exists sets $\mathcal{K}_u, \mathcal{K}'_u, \mathcal{K}_y, \mathcal{K}'_y$ such that $|\mathcal{K}_u \cup \mathcal{K}_y| \leq q$ and $|\mathcal{K}'_u \cup \mathcal{K}'_y| \leq |\mathcal{K}_u \cup \mathcal{K}_y|$ that satisfy

$$y_k(x_0, u_{0:k-1}, B^a(\mathcal{K}_u)u^a_{0:k-1}, D^a(\mathcal{K}_y)d^a_{0:k}) = y_k(x'_0, u_{0:k-1}, B^a(\mathcal{K}'_u)\bar{u}^a_{0:k-1}, D^a(\mathcal{K}'_y)\bar{d}^a_{0:k}),$$

for $x_0 \neq x'_0$ and for all $k \geq 0$. By linearity, we have for some $\mathcal{K}^*_u \subset \{1, \cdots, p\}, \mathcal{K}^*_y \subset \{p+1, \cdots, p+m\}$ where $|\mathcal{K}^*_u \cup \mathcal{K}^*_y| \leq 2q$

$$y_k(x_0 - x'_0, 0, B^a(\mathcal{K}^*_u)\tilde{u}^a_{0:k-1}, D^a(\mathcal{K}^*_y)\tilde{d}^a_{0:k}) = 0,$$

for all $k \geq 0$. Since $x_0 - x'_0 \neq 0$, we have that $\Big(A, [B^a(\mathcal{K}^*_u) \;\; 0_{n\times|\mathcal{K}^*_y|}], C, [0_{m\times|\mathcal{K}_u|} \;\; D^a(\mathcal{K}^*_y)]\Big)$ is not strongly observable.

Now suppose $\Big(A, [B^a(\mathcal{K}^*_u) \;\; 0_{n\times|\mathcal{K}^*_y|}], C, [0_{m\times|\mathcal{K}^*_u|} \;\; D^a(\mathcal{K}^*_y)]\Big)$ is not strongly observable where $\mathcal{K}^*_u \subset \{1, \cdots, p\}, \mathcal{K}^*_y \subset \{p+1, \cdots, p+m\}$ and $|\mathcal{K}^*_u \cup \mathcal{K}^*_y| \leq 2q$. Then, there exists some $x_0 - x'_0 \neq 0$ such that

$$y_k(x_0 - x'_0, 0, B^a(\mathcal{K}^*_u)\tilde{u}^a_{0:k-1}, D^a(\mathcal{K}^*_y)\tilde{d}^a_{0:k}) = 0.$$

Let $\mathcal{K}_u \cup \mathcal{K}'_u = \mathcal{K}^*_u$ where $\mathcal{K}_u \cap \mathcal{K}'_u = \emptyset$ and let $\mathcal{K}_y \cup \mathcal{K}'_y = \mathcal{K}^*_y$ where $\mathcal{K}_y \cap \mathcal{K}'_y = \emptyset$. Moreover, construct these sets so $|\mathcal{K}'_u \cup \mathcal{K}'_y| \leq |\mathcal{K}_u \cup \mathcal{K}_y| \leq q$. Then, we can construct outputs such that

$$y_k(x_0, u_{0:k-1}, B^a(\mathcal{K}_u)u^a_{0:k-1}, D^a(\mathcal{K}_y)d^a_{0:k}) = y_k(x'_0, u_{0:k-1}, B^a(\mathcal{K}'_u)\bar{u}^a_{0:k-1}, D^a(\mathcal{K}'_y)\bar{d}^a_{0:k}).$$

for all $k \geq 0$. Thus, x_0 is not recoverable. □

Note that the index j is arbitrary. Thus, if the property of strong observability is satisfied as stated in Theorem 5.11, then we know that given the output sequence $\{y_0, y_1, y_2, \cdots\}$, we can uniquely recover the state sequence $\{x_0, x_1, x_2, \cdots\}$.

5.2 Structural Analysis of Systems with Undetectable Attacks

In the previous section, we demonstrated that the class of perfect and zero dynamics attacks can be stealthy and harmful. We would like to begin the process of considering how we can design systems to prevent such attacks. In this section, we demonstrate the properties of left invertibility and strong observability are linked to the nonzero structure of a control system. In particular, we can use structural systems and graph theory to characterize systems which almost surely are strong observable and/or left invertible for all sets of feasible attacks.

5.2.1 System Model

Consider the control system

$$\mathbf{x}(k+1) = A\mathbf{x}(k) + B^a\mathbf{u}^a(k), \;\; \mathbf{y}(k) = C\mathbf{x}(k) + D^a\mathbf{u}^a(k).$$

Here $\mathbf{x}(k)$, the state, is in $\mathbb{R}^n$. Next, $\mathbf{y}(k)$, the output, is in $\mathbb{R}^m$. The system represents the attacked subsystem where $\mathbf{u}^a(k) \in \mathbb{R}^{q'}$ is the attacker's input. From a notational perspective, in this section on robust structural analysis, we write the discrete time index k as an argument of the states, inputs, and measurements as opposed to a subscript in order to distinguish vertices in graphs from numerical parameters. Also, for simplicity, when constructing corresponding graphs, we let $\mathbf{u}^a(k)$ collect inputs that both directly compromise actuators and sensors. Without loss of generality, we assume that $\begin{bmatrix} B^a \\ D^a \end{bmatrix}$ has full column rank. We will consider two scenarios, one where the adversary is able to attack both actuators and sensors, and one where the attacker is only able to attack actuators so $D^a = 0$.

We associate a tuple of structural matrices $([A], [B^a], [C], [D^a])$ with (A, B^a, C, D^a). For matrix $[M]$ associated with M, we have that $[M](i,j) = 0$ implies $M(i,j)$ is fixed to be 0. However, if $[M](i,j) \neq 0$, then $M(i,j)$ is a free parameter. It can be shown that the properties of left invertibility and strong observability are generic properties that are directly linked to structure of a system. Specifically, we have the following.

Definition 5.9. *$([A],[B^a],[C],[D^a])$ is structurally strongly observable if an admissible realization of (A, B^a, C, D^a) is strongly observable. $([A],[B^a],[C],[D^a])$ is structurally left invertible if an admissible realization of (A, B^a, C, D^a) is left invertible.*

From the definition, a system that is not structurally strongly observable (structurally left invertible) can not be strongly observable (left invertible). It has been shown that if a system is structurally strongly observable, it is strongly observable for all valid parameters except those lying on some low dimensional algebraic variety which has Lebesgue measure 0 [67]. Likewise, a system that is structurally left invertible is left invertible for all valid parameters except those lying on some low dimensional algebraic variety which has Lebesgue measure 0 [68]. Motivated by this fact, we wish to design systems that are structurally strongly observable and/or left invertible for all feasible attacks.

We next define a feasible attack. Here, we will consider a resource limited adversary so that at most q inputs in a system can be inserted. Without loss of generality, we also would like to make the assumption that $\begin{bmatrix} B^a \\ D^a \end{bmatrix}$ has full column rank. To do this graphically, we introduce the notion of the structural rank of a matrix.

Definition 5.10. *The structural rank of $[M]$ is the maximum rank of an admissible realization of $[M]$.*

Except for a set of measure 0, the structural rank of a matrix is equivalent to its rank. We are now ready to define a feasible attack.

Definition 5.11. *An attack on sensors and actuators is feasible if $\begin{bmatrix} [B^a] \\ [D^a] \end{bmatrix}$ has full column structural rank and the structural rank of $\begin{bmatrix} [B^a] \\ [D^a] \end{bmatrix}$ is less than or equal to q.*

In the case of actuator only attacks, we have the following definition for feasibility.

Definition 5.12. *An attack on only actuators is feasible if* $[B^a]$ *has full column structural rank and the structural rank of* $[B^a]$ *is less than or equal to* q*.*

A defender may wish to remove all feasible zero dynamics attacks in a system. As shown in the previous section, this will eliminate all stealthy destabilizing attacks in stochastic systems as well as eliminate all stealthy attacks in deterministic systems when the defender does not know the initial state. We will say a system that has some zero dynamics for a feasible attack strategy is discreetly attackable.

Definition 5.13. *A system* $([A], [C])$ *is discreetly attackable if there exists a feasible attack strategy for which* $([A], [B^a], [C], [D^a])$ *is not structurally strongly observable and left invertible.*

In some cases it may be sufficient for a defender to design a system to prevent perfect attacks. For instance, this can be the case if the zero dynamics are stable for all feasible attack strategies, or if the defender has exact knowledge of the initial state. We will say a system that can be targeted with a perfect attack is perfectly attackable.

Definition 5.14. *A system* $([A], [C])$ *is perfectly attackable if there exists a feasible attack strategy for which* $([A], [B^a], [C], [D^a])$ *is not structurally left invertible.*

Before, we conclude this section, we wish to construct the graphs associated with our structured system. For the system without attacks $([A], [C])$, we define $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ where $\mathcal{V} = \mathcal{X} \cup \mathcal{Y}$. Here, $\mathcal{X} = \{x_1, \cdots, x_n\}$. We let the vertex x_i be associated with the ith entry of $\mathbf{x}(k)$. Additionally, $\mathcal{Y} = \{y_1, \cdots, y_m\}$. We let the vertex y_i be associated with the ith entry of $\mathbf{y}(k)$. We let $\mathcal{E} = \mathcal{E}_{\mathcal{X},\mathcal{X}} \cup \mathcal{E}_{\mathcal{X},\mathcal{Y}}$. Here, $\mathcal{E}_{\mathcal{X},\mathcal{X}} = \{(x_i, x_j) : [A](j,i) \neq 0\}$ and $\mathcal{E}_{\mathcal{X},\mathcal{Y}} = \{(x_i, y_j) : [C](j,i) \neq 0\}$. For the system with $([A], [B^a], [C], [D^a])$, we define $\mathcal{G}^a = (\mathcal{V}^a, \mathcal{E}^a)$ where $\mathcal{V} = \mathcal{U}^a \cup \mathcal{X} \cup \mathcal{Y}$. Here $\mathcal{U}^a = \{u_1, \cdots, u_{q'}\}$. We let the vertex u_i be associated with the ith entry of $\mathbf{u}(k)$. In addition, $\mathcal{E}^a = \mathcal{E}_{\mathcal{X},\mathcal{X}} \cup \mathcal{E}_{\mathcal{U}^a,\mathcal{X}} \cup \mathcal{E}_{\mathcal{X},\mathcal{Y}} \cup \mathcal{E}_{\mathcal{U}^a,\mathcal{Y}}$ where $\mathcal{E}_{\mathcal{U}^a,\mathcal{X}} = \{(u_i, x_j) : [B^a](j,i) \neq 0\}$ and $\mathcal{E}_{\mathcal{U}^a,\mathcal{Y}} = \{(u_i, y_j) : [D^a](j,i) \neq 0\}$.

We will also consider the special case where an attacker introduces dedicated inputs to a set of nodes $F \subset \mathcal{X} \cup \mathcal{Y}$. Here, each input has a directed edge to exactly one node and no two inputs have a directed edge to the same node. We denote the corresponding attack input nodes as $\mathcal{U}^a_F$.

5.2.2 Graph Theory Preliminaries

In this section we introduce necessary preliminaries from graph theory. Consider a graph $G = (V, E)$. The incoming neighbors to a node v_i or $N^I_{v_i} \subset V$, and the outgoing neighbors $N^O_{v_i} \subset V$ from v_i are

$$N^I_{v_i} \triangleq \{v_j |\ (v_j, v_i) \in E\}, \quad N^O_{v_i} \triangleq \{v_j |\ (v_i, v_j) \in E\}. \tag{5.38}$$

The in-degree of v_i is $|N^I_{v_i}|$ and the out-degree of v_i is $|N^O_{v_i}|$.

Two edges (v_1, v_2) and (v'_1, v'_2) are vertex disjoint or v-*disjoint* if $v_1 \neq v'_1$ and $v_2 \neq v'_2$. A set of edges are v-disjoint if each pair are v-disjoint. Consider sets $\mathcal{A} \subset V$ and $\mathcal{B} \subset V$. An edge (v_1, v_2) from $\mathcal{A}$ to $\mathcal{B}$ has $v_1 \in \mathcal{A}$ and $v_2 \in \mathcal{B}$. We define

$$\theta(\mathcal{A}, \mathcal{B}) \triangleq \text{max number of } v - \text{disjoint edges from } \mathcal{A} \text{ to } \mathcal{B}.$$

θ allows us to characterize the structural rank of a matrix.

Theorem 5.12 ([69]). *The structural rank of* $\begin{bmatrix} [B^a] \\ [D^a] \end{bmatrix}$ *is* $\theta(\mathcal{U}^a, \mathcal{X} \cup \mathcal{Y})$. *Moreover, the structural rank of* $[B^a]$ *is* $\theta(\mathcal{U}^a, \mathcal{X})$.

As such a necessary condition for attack feasibility based on our prior definitions is that $\theta(\mathcal{U}^a, \mathcal{X} \cup \mathcal{Y}) = |\mathcal{U}^a|$ for actuator and sensor attacks and $\theta(\mathcal{U}^a, \mathcal{X}) = |\mathcal{U}^a|$ for actuator only attacks.

A *path* from a set $\mathcal{A} \subset V$ to $\mathcal{B} \subset V$, is a sequence $v_1, v_2, \cdots, v_r$ where $v_1 \in \mathcal{A}$, $v_r \in \mathcal{B}$, and $(v_i, v_{i+1}) \in E$ for $1 \leq i \leq r-1$. An *input output path* from $\mathcal{A} \subset V$ to $\mathcal{B} \subset V$ or IOP from $(\mathcal{A}, \mathcal{B})$ is a path from $\mathcal{A}$ to $\mathcal{B}$ with $v_j \notin \mathcal{A} \cup \mathcal{B},\ 2 \leq j \leq r-1$. A *simple* path has no repeated vertices. An $\mathcal{A}$-*rooted (topped) path* is a simple path with begin (end) vertex in $\mathcal{A}$. Two paths are *disjoint* if they contain no common vertices. Two paths are *internally disjoint* if they have no common vertices

except for possibly the starting and ending vertices. In general l paths are (internally) disjoint if every pair of paths are (internally) disjoint. A set of l disjoint and simple paths from $\mathcal{A} \subset V$ to $\mathcal{B} \subset V$ is referred to as a *linking* of size l or a l-linking from $\mathcal{A}$ to $\mathcal{B}$. We define

$$\rho(\mathcal{A}, \mathcal{B}) \triangleq \text{size of the largest linking between } \mathcal{A} \text{ and } \mathcal{B}.$$

A *vertex separator* between nonadjacent vertices $a \in V$ and $b \in V$ is a set $S \subset V \backslash \{a, b\}$ whose removal deletes all paths from a to b. As shorthand, we refer to S as a vertex separator between (a, b). A *minimum vertex separator* S between (a, b) is a vertex separator between (a, b) with the smallest size.

Theorem 5.13 (Menger [70])**.** *The size of a minimum vertex separator S between (a, b) is equal to the maximum number of internally disjoint paths between a and b.*

We define the set of *essential vertices*, $V_{ess}(\mathcal{A}, \mathcal{B}) \subset V$:

$$V_{ess}(\mathcal{A}, \mathcal{B}) \triangleq \{x | x \in \text{all } \rho(\mathcal{A}, \mathcal{B}) - \text{ linkings from } \mathcal{A} \text{ to } \mathcal{B}\}.$$

Suppose we add new vertices $\underline{a}$ and $\underline{b}$ to graph G where $\underline{a}$ has directed edges to $\mathcal{A}$ and $\underline{b}$ has directed edges coming from $\mathcal{B}$. Then, we have $V_{ess}(\mathcal{A}, \mathcal{B}) = \cup_{S \in \mathcal{S}} S$, where $\mathcal{S}$ is the set of all minimum vertex separators between $(\underline{a}, \underline{b})$ [67].

5.2.3 Perfectly Attackable Systems

In this section, we obtain structural conditions to describe when our system is perfectly attackable. Before beginning, we would like to characterize systems that are structurally left invertible for a given $\mathcal{G}^a$. We have the following result.

Theorem 5.14 ([68],[71])**.** *The system $[A], [B^a], [C], [D^a]$ associated with graph $\mathcal{G}^a$ is structurally left invertible if $\rho(\mathcal{U}^a, \mathcal{Y}) = |\mathcal{U}^a|$.*

Thus, in a structurally left invertible system, we must have a linking of size $|\mathcal{U}^a|$ from the attack inputs to the outputs. Intuitively, to recover the inputs of a system, we need an independent path

from each input to the set of outputs. At a minimum this implies that we need as many sensors as we have attack inputs. We now define conditions which ensure a system is not perfectly attackable regardless of the inputs the adversary is able to corrupt.

Define the graph $f(\mathcal{G}) \triangleq (\mathcal{V} \cup o, \mathcal{E}')$ by adding a node o with incoming directed edges from all sensors $\mathcal{Y}$ to graph $\mathcal{G}$. We have the following.

Theorem 5.15. *A system with sensor and actuator attacks is not perfectly attackable iff for all $x_i \in \mathcal{X}$, the minimum vertex separator S_i between (x_i, o) in $f(\mathcal{G})$ has size $|S_i| \geq q$.*

Proof. Suppose $|S_i| \geq q$ for all $x_i \in \mathcal{X}$.

Now suppose WLOG an adversary implements a feasible attack policy where $|\mathcal{U}^a| = q' \leq q$. Construct a graph $g^a(\mathcal{G}^a)$ by adding an additional vertex u with outgoing edges to $\mathcal{U}^a$ and an additional vertex o with incoming edges from $\mathcal{Y}$. The system is structurally left invertible if and only if the size of the minimum vertex separator between (u, o) in $g^a(\mathcal{G}^a)$ is of size q'.

By assumption, we know that there exists $F \subset \mathcal{X} \cup \mathcal{Y}$ such that $\theta(\mathcal{U}^a, F) = |\mathcal{U}^a|$. Fix such a F and without loss of generality, let $F = \{x_1, \cdots, x_l, y_{l+1}, \cdots, y_{q'}\}$. Moreover, assume $(u_i, x_i) \in \mathcal{E}^a$ for $1 \leq i \leq l$ and $(u_j, y_j) \in \mathcal{E}^a$ for $l+1 \leq j \leq q'$. Let S_u be a minimum vertex separator between (u, o) in $g^a(\mathcal{G}^a)$. Suppose $|S_u| < q'$. Since $|S_u| < q$, there must be a pair of nodes in $\{\{u_1, x_1\}, \cdots, \{u_l, x_l\}, \{u_{l+1}, y_{l+1}\}, \cdots, \{u_{q'}, y_{q'}\}\}$, which does not belong to S_u. If $\{u_j, y_j\}$ does not belong to S_u, there is a path u, u_j, y_j, o which remains even when S_u is removed, contradicting S_u as a vertex separator. Instead suppose $\{u_i, x_i\}$ does not belong to S_u. We know that x_i has $q > |S_u|$ disjoint paths to o. As a result, even when S_u is removed, a path x_i, P^*, o remains. Thus, u, u_i, x_i, P^*, o forms a path from u to o when S_u is removed. Thus $|S_u|$ can not be less than q'. As a result, the system is structurally left invertible.

Now suppose (x_1, o) has minimum vertex separator $S_1 = \{x_2, \cdots, x_l, y_{l+1}, \cdots, y_{r+1}\}$ in $f(\mathcal{G})$ where $r < q$. Choose $\mathcal{U}^a$ to be dedicated inputs where $(u_i, x_i) \in \mathcal{E}^a$ for $1 \leq i \leq l$ and $(u_j, y_j) \in \mathcal{E}^a$ for $l+1 \leq j \leq r+1$. Such an attack is feasible since $r+1 \leq q$. Now construct $g^a(\mathcal{G}^a)$ as before. We argue S_1 is a vertex separator between u and o. Indeed remove S_1. There are no paths from

u_j to o for $2 \leq j \leq r+1$ since each input u_j was a dedicated input to a vertex in S_1. Next, any path from u_1 to o must contain x_1 since u_1 is a dedicated input to x_1. But x_1 has no paths to o after removing S_1. Thus u_1 has no paths to o. As a result, $\rho(\mathcal{U}^a, Y) < |\mathcal{U}^a|$ and the system is not left invertible. □

Consequently to ensure each feasible set of inputs has a maximum linking to the set of outputs, we require that each vertex has q disjoint paths to the set of outputs. In the special scenario where the system has dedicated sensors and dedicated inputs, the prior result still holds. In particular, we assume each sensors measures exactly one state and no 2 sensors measure the same state. Additionally, we assume each attack node manipulates exactly one agent/sensor node, and no two attacks manipulate the same agent/sensor node. In this case we have the following.

Corollary 5.5. *A system with dedicator sensors, dedicated inputs, and sensor and actuator attacks is not perfectly attackable iff for all $x_i \in \mathcal{X}$, the minimum vertex separator S_i between (x_i, o) in $f(\mathcal{G})$ has size $|S_i| \geq q$.*

We next consider a system with only actuator/agent attacks. In this scenario, we make the assumption that each agent has a dedicated sensor so that each sensor measures exactly one state. Moreover, we assume no 2 sensors measure the same state. We have the following result.

Theorem 5.16. *A system with actuator attacks and dedicated sensors is not perfectly attackable iff for all unobserved $x_i \in \mathcal{X}$, the minimum vertex separator S_i between (x_i, o) in $f(\mathcal{G})$ has size $|S_i| \geq q$.*

Proof. Suppose $|S_i| \geq q$ for all unobserved x_i.

Now suppose WLOG an adversary implements a feasible attack policy where $|\mathcal{U}^a| = q' \leq q$. Again, construct a graph $g^a(\mathcal{G}^a)$ by adding an additional vertex u with outgoing edges to $\mathcal{U}^a$ and an additional vertex o with incoming edges from $\mathcal{Y}$. The system is structurally left invertible if and only if the size of the minimum vertex separator between (u, o) in $g^a(\mathcal{G}^a)$ is of size q'.

By assumption, we know that there exists $F \subset \mathcal{X}$ such that $\theta(\mathcal{U}^a, F) = |\mathcal{U}^a|$. Fix such a F and without loss of generality, let $F = \{x_1, \cdots, x_l, x_{l+1}, \cdots, x_{q'}\}$. Moreover, assume $(u_i, x_i) \in \mathcal{E}^a$ for $1 \leq i \leq q'$. Assume for $1 \leq i \leq l$, x_i is unobserved. For $l+1 \leq i \leq q'$, x_i is directly observed by sensor y_i. Let S_u be a minimum vertex separator between (u, o) in $g^a(\mathcal{G}^a)$. Suppose $|S_u| < q'$. Since $|S_u| < q$, there must be a set of nodes in $\{\{u_1, x_1\}, \cdots, \{u_l, x_l\}, \{u_{l+1}, x_{l+1}, y_{l+1}\}, \{u_{q'}, x_{q'}, y_{q'}\}\}$, which does not belong to S_u. If $\{u_j, x_j, y_j\}$ does not belong to S_u, there is a path u, u_j, x_j, y_j, o which remains even when S_u is removed, contradicting S_u as a vertex separator. Instead suppose $\{u_i, x_i\}$ does not belong to S_u where x_i is an unobserved agent. We know that x_i has $q > |S_u|$ disjoint paths to o. As a result, even when S_u is removed, a path x_i, P^*, o remains. Thus, u, u_i, x_i, P^*, o forms a path from u to o when S_u is removed. Thus $|S_u|$ can not be less than q'. As a result, the system is structurally left invertible.

Now suppose (x_1, o) has minimum vertex separator $S_1^* = \{x_2, \cdots, x_l, y_{l+1}, \cdots, y_{r+1}\}$ in $f(\mathcal{G})$ where $r < q$ and x_1 is unobserved. Assume dedicated sensor y_j observes state x_{t_j}. We argue that $S_1 = \{x_2, \cdots, x_l, x_{t_{l+1}}, \cdots, x_{t_{r+1}}\}$ is a vertex separator between x_i and o. If, S_1 is not a vertex separator, then if we remove S_1, any remaining path must contain a vertex in $\{y_{l+1}, \cdots, y_{r+1}\}$ since S_1^* is a vertex separator. However, since y_j are dedicated outputs, any path from x_1 to y_j must contain x_{t_j}. As a result, S_1 is a vertex separator. Since S_1^* is a minimum vertex separator, we know $|S_1| = |S_1^*|$ and there are no repeated vertices in S_1. Without loss of generality let $S_1 = \{x_2, \cdots, x_l, x_{l+1}, \cdots, x_{r+1}\}$.

Choose $\mathcal{U}^a$ to be dedicated inputs where $(u_i, x_i) \in \mathcal{E}^a$ for $1 \leq i \leq r+1$. Such an attack is feasible since $r+1 \leq q$. Now construct $g^a(\mathcal{G}^a)$ as before. We argue S_1 is a vertex separator between u and o. Indeed remove S_1. There are no paths from u_j to o for $2 \leq j \leq r+1$ since each input u_j was a dedicated input to a vertex in S_1. Next, any path from u_1 to o must contain x_1 since u_1 is a dedicated input to x_1. But x_1 has no paths to o after removing S_1. Thus u_1 has no paths to o. As a result, $\rho(\mathcal{U}^a, Y) < |\mathcal{U}^a|$ and the system is not structurally left invertible. □

Thus, by considering a smaller class of attacks, we reduce the structural requirements on the

system. Instead of requiring all agents to have q disjoint paths to the set of outputs, only the unobserved agents require q disjoint paths.

If we introduce the assumption that the attack inputs are dedicated, the prior result still holds.

Corollary 5.6. *A system with actuator attacks, dedicated inputs, and dedicated sensors is not perfectly attackable iff for all unobserved $x_i \in \mathcal{X}$, the minimum vertex separator S_i between (x_i, o) in $f(\mathcal{G})$ has size $|S_i| \geq q$.*

5.2.4 Discreetly Attackable Systems

In this section, we obtain structural conditions to describe when our system is discreetly attackable. Define the graph $f(\mathcal{G}) \triangleq (\mathcal{V} \cup o, \mathcal{E}')$ by adding a node o with incoming directed edges from all sensors $\mathcal{Y}$ to graph $\mathcal{G}$. We have the following:

Theorem 5.17. *A system with sensor and actuator attacks is not discreetly attackable iff:*

C1 For all $\mathcal{T} \subset \mathcal{X} \cup \mathcal{Y}$ with $|\mathcal{T}| = q$, $\theta(\mathcal{X}, (\mathcal{X} \cup \mathcal{Y}) \backslash \mathcal{T}) = n$.

C2 For all $x_i \in \mathcal{X}$, the minimum vertex separator S_i between (x_i, o) in $f(\mathcal{G})$ has size $|S_i| \geq q + 1$.

Proof. Sufficiency: We leverage the following result:

Lemma 5.3 ([72, 73])**.** *For fixed $\mathcal{U}^a$, a system is structurally strongly observable + left invertible iff for $\mathcal{G}^a$*

ci $\theta(\mathcal{X} \cup \mathcal{U}^a, \mathcal{X} \cup \mathcal{Y}) = n + |\mathcal{U}^a|$.

cii Every agent $x_i \in \mathcal{X}$ has a path to $\mathcal{Y}$.

ciii $\Delta_0 \subset V_{ess}(\mathcal{U}^a, \mathcal{Y})$

where $\Delta_0 = \{x \in \mathcal{X} | \rho(x \cup \mathcal{U}^a, \mathcal{Y}) = \rho(\mathcal{U}^a, \mathcal{Y})\}$.

C1 $\implies$ ci : Suppose C1 holds. We know by construction that $\theta(\mathcal{U}^a, \mathcal{X} \cup \mathcal{Y}) = |\mathcal{U}^a|$. Let $\mathcal{Z} \subset \mathcal{X} \cup \mathcal{Y}$ where $|\mathcal{Z}| = |\mathcal{U}^a|$ and $\theta(\mathcal{U}^a, \mathcal{Z}) = |\mathcal{U}^a|$. We know that $\theta(\mathcal{X}, (\mathcal{X} \cup \mathcal{Y})\backslash\mathcal{Z}) = n$ since $|\mathcal{U}^a| \leq q$. Thus, $\theta(\mathcal{X} \cup \mathcal{U}^a, \mathcal{X} \cup \mathcal{Y}) = n + |\mathcal{U}^a|$.

C2 $\implies$ cii, ciii: Suppose C2 holds. Then, cii trivially follows for all feasible attacks. Now, consider arbitrary feasible attack vertices $\mathcal{U}^a$. Suppose ciii does not hold so there exists $x_i \in \mathcal{X}$ satisfying $x_i \in \Delta_0,\ x_i \notin V_{ess}(\mathcal{U}^a, \mathcal{Y})$.

Define $f^a(\mathcal{G}^a) \triangleq (\mathcal{V}^a \cup o \cup u \cup u_i, \mathcal{E}^{a\prime})$ by adding to graph $\mathcal{G}^a$, a node o with edges from $\mathcal{Y}$, a node u with edges to $\mathcal{U}^a$, and a node u_i with edges to $\mathcal{U}^a \cup x_i$. Then, there is a vertex separator S in $f^a(\mathcal{G}^a)$ between (u_i, o) of size $\rho(\mathcal{U}^a, \mathcal{Y}) \leq q$, which is also a vertex separator between (u, o). Thus, $S \subset V_{ess}(\mathcal{U}^a, \mathcal{Y})$. $x_i \notin V_{ess}(\mathcal{U}^a, \mathcal{Y})$ implies $x_i \notin S$. Since x_i has $q + 1$ disjoint paths to o, removing S from $f^a(\mathcal{G}^a)$, does not delete all paths from u_i to o, contradicting S as a vertex separator. Thus, ciii holds for all feasible attacks.

Necessity: $\sim$ C1 $\implies$ $\sim$ ci. Suppose C1 does not hold for some $F' \subset \mathcal{X} \cup \mathcal{Y}$ with $|F'| = q$. Assume an adversary attacks F'. Since $\mathcal{U}^a_{F'}$ only has directed edges to F', and $\theta(\mathcal{X}, (\mathcal{X} \cup \mathcal{Y})\backslash F') < n$, we have $\theta(\mathcal{X} \cup \mathcal{U}^a_{F'}, \mathcal{X} \cup \mathcal{Y}) < n + q$.

Suppose C2 fails to hold. We show an attack to illustrate the presence of zero dynamics so that $\mathbf{y}(k) = 0$ for all $k \geq 0$, but $\mathbf{x}(0) \neq 0$. We let $\mathbf{x}(0) = e_i$, the ith canonical basis vector. Let S_i^* be a minimum vertex separator between x_i and o in $f(\mathcal{G})$. WLOG, let $S_i^* = \{x_1, \cdots, x_l, y_{s_{l+1}}, \cdots, y_{s_{q'}}\}$, $q' \leq q$ and $q' > 0$ (needed for cii). Let $F = S_i^*$ and add inputs $\mathcal{U}^a_F$. Moreover, select $\mathbf{u}^a(k)$ so $\mathbf{y}^{S_i^* \cap \mathcal{Y}}(k), \mathbf{x}_1(k), \cdots, \mathbf{x}_l(k) = 0$ for all $k \geq 0$. Here, $\mathbf{y}^H(k)$ corresponds to values of $\mathbf{y}(k)$ for sensors in H. WLOG $\mathcal{Y}/S_i^* \neq \emptyset$ and we must show $\mathbf{y}^{\mathcal{Y}/S_i^*}(k) = 0$. $\mathcal{X}$ can be partitioned as follows:

1. $\mathcal{X}_1 = \{x \in \mathcal{X} | x \notin x_i$-rooted path, $x \in \mathcal{Y}/S_i^*$-topped path$\}$,
2. $\mathcal{X}_2 = \{x \in \mathcal{X} | x \notin x_i$-rooted path, $x \notin \mathcal{Y}/S_i^*$-topped path$\}$,
3. $\mathcal{X}_3 = \{x \in \mathcal{X} | x \in x_i$-rooted path, $x \notin \mathcal{Y}/S_i^*$-topped path$\}$,
4. $\mathcal{X}_4 = \{x \in \mathcal{X} | x \in x_i$-rooted path, $x \in \mathcal{Y}/S_i^*$-topped path$\}$.

Note any vertex $x_j \in \mathcal{X}$ not in a x_i-rooted path, cannot be part of a $\mathcal{U}_F^a$-rooted path. Otherwise, if there was a $\mathcal{U}_F^a$-rooted path, then $\exists$ a simple path from $S_i^*/\mathcal{Y}$ to x_j. Since x_i has a simple path to all $s \in S_i^*/\mathcal{Y}$, x_j is part of an x_i-rooted path, which is a contradiction. Permuting $\mathbf{x}(k)$, we have:

$$A = \begin{bmatrix} A_{11} & 0 & 0 & 0 \\ A_{21} & A_{22} & 0 & 0 \\ A_{31} & A_{32} & A_{33} & A_{34} \\ A_{41} & 0 & 0 & A_{44} \end{bmatrix}, \; B^a = \begin{bmatrix} 0 \\ 0 \\ 0 \\ B_4 \end{bmatrix}, \; C^{\mathcal{Y}/S_i^*} = \begin{bmatrix} C_1 & 0 & 0 & C_4 \end{bmatrix}, \; \mathbf{x}(k) = \begin{bmatrix} \mathbf{x}^1(k) \\ \mathbf{x}^2(k) \\ \mathbf{x}^3(k) \\ \mathbf{x}^4(k) \end{bmatrix}.$$

$\mathbf{x}^j(k)$ is associated with agents $\mathcal{X}_j$. $C^{\mathcal{Y}/S_i^*}$ is the portion C associated with $\mathcal{Y}/S_i^*$. Since $\mathcal{X}_1$ and $\mathcal{X}_2$ are not part of x_i-rooted paths, they cannot be affected by $\mathcal{X}_3, \mathcal{X}_4$. Since $\mathcal{X}_2, \mathcal{X}_3$ are not part of $\mathcal{Y}/S_i^*$-topped paths, they do not affect $\mathcal{X}_4$ or $\mathcal{X}_1$. B^a is obtained from the fact that $S_i^*/\mathcal{Y} \subset \mathcal{X}_4$. $C^{\mathcal{Y}/S_i^*}$ is obtained since $\mathcal{X}_2, \mathcal{X}_3$ do not have $\mathcal{Y}/S_i^*$-topped paths.

Since $\mathbf{x}^1(k+1) = A_{11}\mathbf{x}^1(k), \; \mathbf{x}^1(0) = 0$, $\mathbf{x}^1(k) = 0$ for all k. Thus, the dynamics of sensors $\mathcal{Y}/S_i^*$ are given by

$$\mathbf{x}^4(k+1) = A_{44}\mathbf{x}^4(k) + B_4\mathbf{u}^a(k) + 0, \tag{5.39}$$

$$\mathbf{y}^{\mathcal{Y}/S_i^*}(k) = C_4\mathbf{x}^4(k) + 0. \tag{5.40}$$

In the special case that $S_i^* \subset \mathcal{Y}$, $\mathcal{X}_4 = \emptyset$ and the result follows. WLOG, assume $S_i^* \not\subset \mathcal{Y}$. To analyze $\mathcal{X}_4$, consider the partition $\bar{\mathcal{X}}_1, \bar{\mathcal{X}}_2, \bar{\mathcal{X}}_3, \bar{\mathcal{X}}_4, \bar{\mathcal{X}}_5 = S_i^*/\mathcal{Y}, \bar{\mathcal{X}}_6 = x_i$ where

1. $\bar{\mathcal{X}}_1 = \{x \in \mathcal{X}_4 \backslash (\bar{\mathcal{X}}_5 \cup \bar{\mathcal{X}}_6) | x \in$ IOP from $(x_i, S_i^*/\mathcal{Y})\}$,
2. $\bar{\mathcal{X}}_2 = \{x \in \mathcal{X}_4 \backslash \bar{\mathcal{X}}_5 | x \in$ IOP from $(S_i^*/\mathcal{Y}, \mathcal{Y}/S_i^*)\}$,
3. $\bar{\mathcal{X}}_3 = \{x \in \mathcal{X}_4 | x \in$ IOP from $(x_i, \mathcal{Y}/S_i^*)\} - (\bar{\mathcal{X}}_1 \cup \bar{\mathcal{X}}_2 \cup \bar{\mathcal{X}}_5 \cup \bar{\mathcal{X}}_6)$,
4. $\bar{\mathcal{X}}_4 = \{x \in \mathcal{X}_4 | x \notin$ IOP from $(x_i, \mathcal{Y}/S_i^*)\}$.

We verify this is a partition. If $x \in \bar{\mathcal{X}}_1$, $\exists$ an IOP from $(x_i, \mathcal{Y}/S_i^*)$ containing x since $\exists$ a path from $s \in S_i^*/\mathcal{Y}$ to $\mathcal{Y}/S_i^*$ without x_i. Indeed, consider q' internally disjoint paths from x_i to o which

WLOG do not contain x_i as an intermediate vertex. Each path contains exactly one vertex of S_i^* and thus $\exists$ a path from $s \in S_i^*/\mathcal{Y}$ to $\mathcal{Y}/S_i^*$ not containing x_i. If $x \in \bar{\mathcal{X}}_2$, $\exists$ an IOP from $(x_i, \mathcal{Y}/S_i^*)$ containing x since there is a path from x_i to any $s \in S_i^*/\mathcal{Y}$ and an IOP from $(S_i^*/\mathcal{Y}, \mathcal{Y}/S_i^*)$ cannot contain x_i. It is clear, $\cup_{j=1}^{6} \bar{\mathcal{X}}_1 = \mathcal{X}_4$.

Next, observe $\bar{\mathcal{X}}_3$ and $\bar{\mathcal{X}}_5$ are pairwise disjoint from all other subsets. Additionally, since S_i^* is vertex separator between (x_i, o) we note $x_i \notin \bar{\mathcal{X}}_2$. Thus, since $x_i \notin \bar{\mathcal{X}}_1$ and $x_i \notin \bar{\mathcal{X}}_4$, $\bar{\mathcal{X}}_6$ is pairwise disjoint from all other subsets. We next show $\bar{\mathcal{X}}_1 \cap \bar{\mathcal{X}}_2 = \emptyset$. The existence of $x \in \bar{\mathcal{X}}_1 \cap \bar{\mathcal{X}}_2$, implies $\exists$ a path from x_i to $\mathcal{Y}/S_i^*$ not containing $S_i^*/\mathcal{Y}$, which contradicts S_i^* as a vertex separator. Finally, $(\bar{\mathcal{X}}_1 \cup \bar{\mathcal{X}}_2) \cap \bar{\mathcal{X}}_4 = \emptyset$ since $x \in \bar{\mathcal{X}}_4$ cannot be part of an IOP from $(x_i, \mathcal{Y}/S_i^*)$.

We make the following claims about the partitioned sets.

Lemma 5.4. *Let $x \in \bar{\mathcal{X}}_3$. There is a path from $S_i^*/\mathcal{Y}$ to x.*

Proof. Suppose Not. If $x \in \bar{\mathcal{X}}_3$, $\exists$ an IOP from $(x_i, \mathcal{Y}/S_i^*)$ with x. Since $\nexists$ path from $S_i^*/\mathcal{Y}$ to x, $\exists$ an IOP from $(x_i, S_i^*/\mathcal{Y})$ with x, contradicting $\bar{\mathcal{X}}_1 \cap \bar{\mathcal{X}}_3 = \emptyset$. □

Lemma 5.5. $\theta(\bar{\mathcal{X}}_1 \cup \bar{\mathcal{X}}_3 \cup \bar{\mathcal{X}}_4 \cup \bar{\mathcal{X}}_6, \bar{\mathcal{X}}_2 \cup \mathcal{Y}/S_i^*) = 0$.

Proof. If there was a directed edge from $a \in \bar{\mathcal{X}}_1 \cup \bar{\mathcal{X}}_6$ to $b \in \bar{\mathcal{X}}_2 \cup \mathcal{Y}/S_i^*$, then there is a path from x_i to $\mathcal{Y}/S_i^*$ containing edge (a, b), not containing $S_i^*/\mathcal{Y}$, contradicting S_i^* as a vertex separator. If there was a directed edge from $a \in \bar{\mathcal{X}}_3$ to $b \in \bar{\mathcal{X}}_2 \cup \mathcal{Y}/S_i^*$, by Lemma 5.4 there is a IOP from $(S_i^*/\mathcal{Y}, \mathcal{Y}/S_i^*)$ containing edge (a, b). This contradicts $\bar{\mathcal{X}}_3 \cap \bar{\mathcal{X}}_2 = \emptyset$. If there was a directed edge from $a \in \bar{\mathcal{X}}_4$ to $b \in \bar{\mathcal{X}}_2 \cup \mathcal{Y}/S_i^*$, there would be an IOP from $(x_i, \mathcal{Y}/S_i^*)$ containing a, contradicting the definition of $\bar{\mathcal{X}}_4$. □

Let $\bar{\mathbf{x}}^j(k)$ be states associated with $\bar{\mathcal{X}}_j$. Leveraging Lemma 5.5 and the fact that only $\bar{\mathcal{X}}_5$ has edges from $\mathcal{U}_F^a$:

$$\bar{\mathbf{x}}^2(k+1) = \bar{A}_{22}\bar{\mathbf{x}}^2(k) + \bar{A}_{25}\bar{\mathbf{x}}^5(k),$$
$$\mathbf{y}^{\mathcal{Y}/S_i^*}(k) = \bar{C}_2\bar{\mathbf{x}}^2(k) + \bar{C}_5\bar{\mathbf{x}}^5(k),\ \bar{\mathbf{x}}^2(0) = 0.$$

Recall, that $\mathbf{u}^a(k)$ is chosen so that $\bar{\mathbf{x}}^5(k) = 0$. We then have that $\mathbf{y}^{\mathcal{Y}/S_i^*}(k) = 0$ and Theorem 5.17 holds. □

As such, preventing zero dynamics attacks as opposed to only perfect attacks requires additional structural considerations. Specifically, an extra independent path is needed from each state to the set of outputs. This may necessitate extra sensors.

Corollary 5.7. *A system with sensor and actuator attacks is not discreetly attackable only if it contains at least* $q + 1$ *sensors.*

Moreover, an extra maximum matching condition C1 is required. In general, it appears the problem of verifying C1 is combinatorial since we must verify there is a maximum matching in $m + n$ choose q distinct graphs. Fortunately, we can simplify required analysis if we consider the instance where each agent has a self-loop.

Corollary 5.8. *Suppose each agent* $x_i \in \mathcal{X}$ *has a self-loop. A system with sensor and actuator attacks is not discreetly attackable iff the minimum vertex separator* S_i *between* (x_i, o) *has size* $|S_i| \geq q + 1$.

Proof. It is sufficient to show that the self-loop condition implies ci for all feasible attacks. WLOG, consider an arbitrary feasible attack. We know that there exists $F \subset \mathcal{X} \cup \mathcal{Y}$ such that $\theta(\mathcal{U}^a, \mathcal{F}) = |\mathcal{U}^a|$. Construct a maximum linking $\mathcal{L}$ from $\mathcal{U}^a$ to $\mathcal{Y}$. Since each agent has $q+1$ paths to o, we know $\rho(\mathcal{U}^a, \mathcal{Y}) = |\mathcal{U}^a|$ from Theorem 5.15. Let $\mathcal{X}_\mathcal{L}$ be the set of vertices in $\mathcal{X}$ belonging to $\mathcal{L}$. $\mathcal{L}$ gives a maximum set of $v-$disjoint edges from $\mathcal{U}^a \cup \mathcal{X}_\mathcal{L}$ to $\mathcal{X}_\mathcal{L} \cup \mathcal{Y}$. Thus, $\theta(\mathcal{U}^a \cup \mathcal{X}_\mathcal{L}, \mathcal{X}_\mathcal{L} \cup \mathcal{Y}) = |\mathcal{X}_\mathcal{L}| + |\mathcal{U}^a|$. Since each agent has a self-loop, $\theta(\mathcal{X} \backslash \mathcal{X}_\mathcal{L}, \mathcal{X} \backslash \mathcal{X}_\mathcal{L}) = |\mathcal{X} \backslash \mathcal{X}_\mathcal{L}|$. Therefore, $\theta(\mathcal{U}^a \cup \mathcal{X}, \mathcal{X} \cup \mathcal{Y}) = n + |\mathcal{U}^a|$. □

As with perfect attacks, we note that Theorem 5.17 and Corollary 5.8 also hold in the special case when the system has dedicated sensors and dedicated inputs. This can be seen since Theorem 5.17 never makes an assumption about the structure of C and the proposed attack considers a strategy with dedicated inputs. We formalize this below.

Corollary 5.9. *A system with dedicated sensors, dedicated inputs, and sensor and actuator attacks is not discreetly attackable iff:*

C1 For all $\mathcal{T} \subset \mathcal{X} \cup \mathcal{Y}$ *with* $|\mathcal{T}| = q$, $\theta(\mathcal{X}, (\mathcal{X} \cup \mathcal{Y}) \backslash \mathcal{T}) = n$.

C2 For all $x_i \in \mathcal{X}$, *the minimum vertex separator* S_i *between* (x_i, o) *in* $f(\mathcal{G})$ *has size* $|S_i| \geq q + 1$.

If such a system has self loops for each agent $x_i \in \mathcal{X}$, *then it is not discreetly attackable by sensor and actuator attacks iff the minimum vertex separator* S_i *between* (x_i, o) *has size* $|S_i| \geq q + 1$.

Next, we consider the special case of actuator/agent only attacks. Here, we make the assumption each sensor is a dedicated sensor. That is each sensor measures exactly one agent. Moreover, we will assume no 2 sensors measure the same agent.

Theorem 5.18. *A system with actuator attacks and dedicated sensors is not discreetly attackable iff:*

D1 For all $\mathcal{T} \subset \mathcal{X}$ *with* $|\mathcal{T}| = q$, $\theta(\mathcal{X}, (\mathcal{X} \cup \mathcal{Y}) \backslash \mathcal{T}) = n$.

D2 For all unobserved agents x_i *the minimum vertex separator* S_i *between* (x_i, o) *in* $f(\mathcal{G})$ *has size* $|S_i| \geq q + 1$.

Proof. D1 $\implies$ ci : Suppose D1 holds. We know by construction that $\theta(\mathcal{U}^a, \mathcal{X}) = |\mathcal{U}^a|$. Let $\mathcal{Z} \subset \mathcal{X}$ where $|\mathcal{Z}| = |\mathcal{U}^a|$ and $\theta(\mathcal{U}^a, \mathcal{Z}) = |\mathcal{U}^a|$. We know that $\theta(\mathcal{X}, (\mathcal{X} \cup \mathcal{Y}) \backslash \mathcal{Z}) = n$ since $|\mathcal{U}^a| \leq q$. Thus, $\theta(\mathcal{X} \cup \mathcal{U}^a, \mathcal{X} \cup \mathcal{Y}) = n + |\mathcal{U}^a|$.

D2 $\implies$ cii, ciii: Suppose D2 holds. cii follows from the fact that observed agents have direct edges to $\mathcal{Y}$ and unobserved agents have a nontrivial vertex separator to o. Now, consider arbitrary feasible attack vertices $\mathcal{U}^a$. Suppose ciii does not hold so there exists $x_i \in \mathcal{X}$ satisfying $x_i \in \Delta_0,\ x_i \notin V_{ess}(\mathcal{U}^a, \mathcal{Y})$.

We first argue x_i can not be observed vertex. Suppose Not, so that x_i is observed by y_i. We define $f^a(\mathcal{G}^a) \triangleq (\mathcal{V}^a \cup o \cup u \cup u_i, \mathcal{E}^{a\prime})$ by adding to graph $\mathcal{G}^a$, a node o with edges from $\mathcal{Y}$, a node u with edges to $\mathcal{U}^a$, and a node u_i with edges to $\mathcal{U}^a \cup x_i$. Then, there is a vertex separator S in

$f^a(\mathcal{G}^a)$ between (u_i, o) of size $\rho(\mathcal{U}^a, \mathcal{Y}) \leq q$, which is also a vertex separator between (u, o), which also satisfies $S \cap y_i = \emptyset$. If $y_i \in S$, then we argue that $S' = S \cup x_i - y_i$ is also a vertex separator between (u_i, o). Indeed if one removes $S - y_i$ from $f^a(\mathcal{G}^a)$ any path to o from u_i must contain y_i. Since y_i is a dedicated output, such a path must also contain x_i. However, if S' is a vertex separator between u_i and o, it must also be a vertex separator between u and o. As such $S' \subset V_{ess}(\mathcal{U}^a, \mathcal{Y})$. However, by assumption $x_i \notin V_{ess}(\mathcal{U}^a, \mathcal{Y})$. Thus, S' is not a vertex separator between u_i and o and as such S can not contain y_i. Again $x_i \notin S$ because $S \subset V_{ess}(\mathcal{U}^a, \mathcal{Y})$. Thus, if S is removed from $f^a(\mathcal{G}^a)$, the path u_i, x_i, y_i, o still exists, contradicting S as a vertex separator.

As a result, x_i must be unobserved. Define $f^a(\mathcal{G}^a)$ as before. Then, there is a vertex separator S in $f^a(\mathcal{G}^a)$ between (u_i, o) of size $\rho(\mathcal{U}^a, \mathcal{Y}) \leq q$, which is also a vertex separator between (u, o). Thus, $S \subset V_{ess}(\mathcal{U}^a, \mathcal{Y})$. $x_i \notin V_{ess}(\mathcal{U}^a, \mathcal{Y})$ implies $x_i \notin S$. Since x_i has $q + 1$ disjoint paths to o, removing S from $f^a(\mathcal{G}^a)$, does not delete all paths from u_i to o, contradicting S as a vertex separator. Thus, ciii holds for all feasible attacks.

Necessity: $\sim$ D1 $\implies$ $\sim$ ci. Suppose D1 does not hold for some $F' \subset \mathcal{X}$ with $|F'| = q$. Assume an adversary attacks F' with dedicated inputs. Since $\mathcal{U}^a_{F'}$ only has directed edges to F' and $\theta(\mathcal{X}, (\mathcal{X} \cup \mathcal{Y}) \backslash F') < n$, we have $\theta(\mathcal{X} \cup \mathcal{U}^a_{F'}, \mathcal{X} \cup \mathcal{Y}) < n + q$.

Suppose D2 fails to hold for some unobserved x_i. We argue there exists a minimum vertex separator S_i^* between x_i and o in $f(\mathcal{G})$ such that $S_i^* \subset \mathcal{X}$. To see this let S_i be a minimum vertex between x_i containing vertex y_j observing x_j. We argue that $S_i \cup x_j - y_j$ is a minimum vertex separator between x_i and o. Indeed, if we remove $S_i - y_j$, any path from x_i to o must contain y_j, which in turn must contain x_j, since y_j is a dedicated sensor for x_j. As a result, $S_i \cup x_j - y_j$ is a vertex separator between x_i and o in $f(\mathcal{G})$ and this vertex separator is minimal. An inductive argument shows we can construct $S_i^* \subset \mathcal{X}$. We can then proceed as in the proof of 5.17 by attacking S_i^* to obtain the final result. □

Considering agent only attacks reduces the structural requirements once more. Observed agents no longer need $p + 1$ disjoint paths to the set of outputs and we only need a maximum matching

for n choose q graphs instead of $n+m$ choose q graphs. Again, verifying condition D1 appears to be combinatorial. Once more, we can simplify required analysis if we consider the instance where each agent has a self-loop.

Corollary 5.10. *Suppose each agent has a self-loop. A system with actuator attacks and dedicated sensors is not discreetly attackable iff the minimum vertex separator S_i between each unobserved agent x_i and o has size $|S_i| \geq q+1$.*

Proof. It is sufficient to show that the self-loop condition implies ci for all feasible attacks. WLOG, consider an arbitrary feasible attack. We know that there exists $F \subset \mathcal{X}$ such that $\theta(\mathcal{U}^a, F) = |\mathcal{U}^a|$. Construct a maximum linking $\mathcal{L}$ from $\mathcal{U}^a$ to $\mathcal{Y}$. Since each unobserved agent has $q+1$ paths to o, we know $\rho(\mathcal{U}^a, \mathcal{Y}) = |\mathcal{U}^a|$ from Theorem 5.16. Let $\mathcal{X}_\mathcal{L}$ be the set of vertices in $\mathcal{X}$ belonging to $\mathcal{L}$. $\mathcal{L}$ gives a maximum set of $v-$ disjoint edges from $\mathcal{U}^a \cup \mathcal{X}_\mathcal{L}$ to $\mathcal{X}_\mathcal{L} \cup \mathcal{Y}$. Thus, $\theta(\mathcal{U}^a \cup \mathcal{X}_\mathcal{L}, \mathcal{X}_\mathcal{L} \cup \mathcal{Y}) = |\mathcal{X}_\mathcal{L}| + |\mathcal{U}^a|$. Since each agent has a self-loop, $\theta(\mathcal{X} \backslash \mathcal{X}_\mathcal{L}, \mathcal{X} \backslash \mathcal{X}_\mathcal{L}) = |\mathcal{X} \backslash \mathcal{X}_\mathcal{L}|$. Therefore, $\theta(\mathcal{U}^a \cup \mathcal{X}, \mathcal{X} \cup \mathcal{Y}) = n + |\mathcal{U}^a|$. □

If we introduce the additional assumption that the attack inputs are dedicated, the prior results analyzing discreetly attackable systems with agent/actuator attacks still hold. Specifically, we have the following.

Corollary 5.11. *A system with actuator attacks, dedicated inputs and dedicated sensors is not discreetly attackable iff:*

D1 For all $\mathcal{T} \subset \mathcal{X}$ with $|\mathcal{T}| = q$, $\theta(\mathcal{X}, (\mathcal{X} \cup \mathcal{Y}) \backslash \mathcal{T}) = n$.

D2 For all unobserved agents x_i the minimum vertex separator S_i between (x_i, o) in $f(\mathcal{G})$ has size $|S_i| \geq q+1$.

Moreover, if each agent has a self-loop, a system with actuator attacks, dedicated inputs, and dedicated sensors is not discreetly attackable iff the minimum vertex separator S_i between each unobserved agent x_i and o has size $|S_i| \geq q+1$.

5.2.5 System Verification

If $f(\mathcal{G})$ has self-loops at each agent, we can efficiently determine if a system is discreetly attackable in a system with both sensor and actuator attacks. We do not require the self-loop assumption to determine if a system is perfectly attackable. To determine if a fixed agent (x_i, o) has minimum vertex separator S_i of size $q+1$ (or q), we solve a $0-1$ maximum flow problem. We consider a graph $h^i(f(\mathcal{G})) = (\mathcal{V}_{H_i}, \mathcal{E}_{H_i})$, where $|\mathcal{V}_{H_i}| = 2|\mathcal{V}|$ and $|\mathcal{E}_{H_i}| \leq |\mathcal{E}'| + |\mathcal{V}| - 1$. First, all self-loops can be eliminated. Then, every $v \in \mathcal{V} \backslash x_i$ is converted to a pair of nodes, v_{in} and v_{out}, where $N^I_{v_{in}} = N^I_v$, $N^O_{v_{in}} = v_{out}$, $N^I_{v_{out}} = v_{in}$, $N^O_{v_{out}} = N^O_v$. Moreover, all incoming edges to x_i are removed. All edges in $\mathcal{E}_{H_i}$ have capacity 1. (x_i, o) has minimum vertex separator S_i of size at least $q+1$ (or q) if and only if the maximum flow from source x_i to sink o in $h^i(f(\mathcal{G}))$ is at least $q+1$ (or q). Using Dinic's algorithm, [74, 75] this can be determined in $O((2|\mathcal{V}|)^{\frac{1}{2}}(|\mathcal{E}'| + |\mathcal{V}| - 1))$ time. Since, we must verify $|S_i| \geq q+1$ (or $|S_i| \geq q$) for each of n agents, the worst case computational complexity is $O(n(2|\mathcal{V}|)^{\frac{1}{2}}(|\mathcal{E}'| + |\mathcal{V}| - 1))$. This outperforms algebraic methods based on the matrix pencil [65] and graphical methods based on Lemma 5.3 which verify a system's strong observability/left invertibility for fixed attack nodes, which is a combinatorial task.

The problem of system verification becomes simpler in a system with actuator only attacks and dedicated sensors. In this case we have the following proposition.

Lemma 5.6. *Let $f'(\mathcal{G})$ be constructed from $\mathcal{G}$ by removing all nodes in $\mathcal{Y}$, adding a node o, and adding directed edges from each observed node to o. An unobserved node x_i has minimum vertex separator to o of size r in $f(\mathcal{G})$ if and only if unobserved node x_i has minimum vertex separator to o of size r in $f'(\mathcal{G})$.*

Proof. We first observe that if S is a vertex separator between x_i and o in $f'(\mathcal{G})$, it is also a vertex separator of x_i and o in $f(\mathcal{G})$. Suppose Not. Then, after deleting S in $f(\mathcal{G})$, there is a path x_i, P, y_j, o in $f(\mathcal{G})$. However, this means there is a path x_i, P, o in $f'(\mathcal{G})$, which is a contradiction. As a result, the size of a minimum vertex separator in S_i between x_i and o in $f(\mathcal{G})$ is less than or equal to the size of a minimum vertex separator S'_i between x_i and o in $f'(\mathcal{G})$. That is $|S_i| \leq |S'_i|$.

Let S_i be a minimum vertex separator between x_i and o in $f(\mathcal{G})$. If y_j observing x_j is in S_i, we argue that $S_i \cup x_j - y_j$ is a minimum vertex separator. Indeed if one removes $S_i - y_j$ from $f(\mathcal{G})$, there must must be a path from x_i to o, which must contain y_j. However any path to y_j must also contain x_j since y_j is a dedicated observer. Thus, $S_i \cup x_j - y_j$ is a minimum vertex separator. Consequently, without loss of generality, we can assume $S_i \subset \mathcal{X}$. Suppose we remove S_i from $f'(\mathcal{G})$. Then, there would be no path from x_i to o in $f'(\mathcal{G})$. If such a path x_i, P, o existed, then we would be able to construct a path x_i, P, y_k, o in $f(\mathcal{G})$ after removing S_i. Thus, S_i is a vertex separator between x_i and o in $f'(\mathcal{G})$. As such, the size of minimum separator between x_i and o in $f(\mathcal{G})$ is greater than or equal to the size of a minimum vertex separator S_i' between x_i and o in $f'(\mathcal{G})$. That is $|S_i| \geq |S_i'|$. The result follows. □

Thus, we can solve maximum flow problems on a smaller graph to determine if a system is perfectly or discreetly attackable. Moreover, we only need to solve a maximum flow problem at most once for each unobserved node. More specifically, in the case of actuator only attacks and dedicated sensors, we consider a graph $h_i'(f'(\mathcal{G})) = (\mathcal{V}'_{H_i}, \mathcal{E}'_{H_i})$, where $|\mathcal{V}'_{H_i}| = 2|\mathcal{X}|$ and $|\mathcal{E}_{H_i}|' \leq |\mathcal{E}| + |\mathcal{X}| - 1$ for each unobserved x_i. First, all self-loops are eliminated. Then, every $v \in \mathcal{X} \backslash x_i$ is converted to a pair of nodes, v_{in} and v_{out}, where $N^I_{v_{in}} = N^I_v$, $N^O_{v_{in}} = v_{out}$, $N^I_{v_{out}} = v_{in}$, $N^O_{v_{out}} = N^O_v$. Moreover, all incoming edges to x_i are removed. All edges in $\mathcal{E}'_{H_i}$ have capacity 1. For unobserved x_i, (x_i, o) has minimum vertex separator S_i of size at least $q+1$ (or q) if and only if the maximum flow from source x_i to sink o in $h_i'(f'(\mathcal{G}))$ is at least $q+1$ (or q). Using Dinic's algorithm, [74, 75] this can be determined in $O((2|\mathcal{X}|)^{\frac{1}{2}}(|\mathcal{E} + |\mathcal{X}| - 1))$ time. Since, we must verify $|S_i| \geq q+1$ (or $|S_i| \geq q$) for each of $n-m$ unobserved agents, the worst case computational complexity is $O((n-m)(2|\mathcal{X}|)^{\frac{1}{2}}(|\mathcal{E}| + |\mathcal{X}| - 1))$.

5.3 Robust Structural Design of Distributed Control Systems

Distributed control systems (DCSs) have become prevalent in today's world. A DCS is a system where components such as sensors, actuators, and controllers are separated over a large network. DCSs allows operators to control multiple local environments while simultaneously meeting various global objectives. The ability of a DCS to meet society's demands for large scale control has made such systems common in a variety of applications including sensor networks, the smart grid, vehicular systems, and manufacturing.

We consider the setting of DCS where no more than q agents and/or sensors may be compromised. Here we formulate and solve optimization problems which minimize sensing and communication in DCS while ensuring resilience to undetectable attacks. We first consider an unconstrained minimization problem, where there are no restrictions on which agents may communicate or be observed. For a fixed number of observers, we find the minimum number of communication links that can guarantee perfect detectability. Furthermore, we completely characterize the subset of networks which solve the optimization problem and contain no cycles among unobserved agents. We then show the problem of jointly minimizing the number of sensors and communication links strictly depends upon the cost of sensing and communicating. This work is the extended to the constrained case where a set of agents are not able to communicate. We consider the optimal design of these systems as a means of active detection. In particular, we introduce systems which are design fundamentally to ensure adversarial behavior is detectable.

5.3.1 System Model

Graphical Model: We will model a Distributed Control System (DCS) both graphically and algebraically. We assume there are n agents, $\mathcal{X} \triangleq \{x_1, \cdots, x_n\}$ that communicate with each other and are observed by m sensors, $\mathcal{Y} \triangleq \{y_1, \cdots, y_m\}$ where we assume $m \leq n$. We model interactions using a directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with vertices $\mathcal{V} \triangleq \mathcal{X} \cup \mathcal{Y}$. The edges $\mathcal{E} \subset \mathcal{V} \times \mathcal{V}$ capture agent/sensor interactions. If $(x_i, x_j) \in \mathcal{E}$, agent x_i sends messages to x_j. If sensor y_j

measures state x_i, $(x_i, y_j) \in \mathcal{E}$. Each agent $x_i \in \mathcal{X}$ has a self-loop, therefore $(x_i, x_i) \in \mathcal{E}$.

Algebraic Model: We assume each agent x_i has a scalar time dependent state $\mathbf{x}_i(k)$ with dynamics given as follows:

$$\mathbf{x}_i(k+1) = a_{ii}\mathbf{x}_i(k) + \mathbf{u}_i(k). \tag{5.41}$$

The input $\mathbf{u}_i(k)$ is a linear function of the states of x_i's incoming neighbors and a centrally known input $\mathbf{u}_i^{ff}(k)$ so

$$\mathbf{u}_i(k) = \mathbf{u}_i^{ff}(k) + \sum_{j \neq i} a_{ij}\mathbf{x}_j(k), \tag{5.42}$$

where $x_j \notin N_{x_i}^I \cap \mathcal{X} \implies a_{ij} = 0$. Without loss of generality $\mathbf{u}_i^{ff}(k) = 0$. Each agent is assumed to have a scalar state. From a notational perspective, in this section on robust structural design, we again write the discrete time index k as an argument of the states, inputs, and measurements as opposed to a subscript in order to distinguish vertices in graphs from numerical parameters.

Remark 5.1. *The state $\mathbf{x}_i(k)$ can refer to a physical quantity such as temperature or simply a quantity for distributed processing (e.g. consensus). While it is assumed each agent has a scalar state in this subsection, similar tools for DCS analysis and design can be incorporated in the vector case. Examining the vector case is a subject of current research.*

A set of dedicated sensors $\mathcal{Y}$ measure the state of a subset of agents. The outputs are sent to a central operator for estimation and detection. A dedicated sensor measures the state of one agent and no two sensors measure the state of the same agent. The output of sensor y_i measuring x_j at time k is

$$\mathbf{y}_i(k) = \mathbf{x}_j(k). \tag{5.43}$$

Remark 5.2. *The assumption of dedicated sensors is based on the fact that the system is distributed and it's likely that sensors would not have the physical access to measure multiple agents accurately. While we assume there are no redundant sensors, in practice multiple sensors can be added to measure a single state in the physical system for robustness. However, for our treatment, it is likely that if an attacker can manipulate one sensor that measures a state x_j, he has the ability to access*

and manipulate all sensors that measure the state x_j, especially if the hardware itself is redundant. As a result, for the purposes of modeling attacks, we can consider redundant sensors as a single node.

For simplicity, we concatenate state and output vectors

$$\mathbf{x}(k) \triangleq \begin{bmatrix} \mathbf{x}_1(k) \cdots \mathbf{x}_n(k) \end{bmatrix}^T, \ \mathbf{y}(k) \triangleq \begin{bmatrix} \mathbf{y}_1(k) \cdots \mathbf{y}_m(k) \end{bmatrix}^T,$$

so that the dynamics of the full control system are given by

$$\mathbf{x}(k+1) = A\mathbf{x}(k), \ \ \mathbf{y}(k) = C\mathbf{x}(k). \tag{5.44}$$

The pair (A, C) is assumed to be observable. Letting $\mathbb{I}$ be the indicator function, A and C can be defined entrywise:

$$A(i,j) = a_{ij}, \ \ C(i,j) = \mathbb{I}_{(x_j, y_i) \in \mathcal{E}}.$$

Since (A, C) is observable, the state can be estimated using a linear filter.

$$\hat{\mathbf{x}}(k+1) = (A - KCA)\hat{\mathbf{x}}(k) + K\mathbf{y}(k+1), \tag{5.45}$$

$$\mathbf{z}(k+1) = \mathbf{y}(k+1) - CA\hat{\mathbf{x}}(k). \tag{5.46}$$

Here, K is chosen so $(A - KCA)$ is Schur stable. The residue $\mathbf{z}(k)$ can be used to perform detection. As we have seen. smaller residues are often indicative of normal behavior while larger residues are associated with faulty or malicious behavior.

5.3.2 DCS Attack Model

Graphical Model: We now define our DCS model under attack. At time 0 an unknown subset of the agents and sensors F are compromised. No more than q agents and sensors can be corrupted. In other words, the operator would like the system to be resilient to up to q malicious failures. The set of all feasible sets of attacked sensor and agent nodes is given by $\mathcal{F}_{xy}$:

$$\mathcal{F}_{xy} = \{F \subset \mathcal{V} : \ |F| \leq q\}. \tag{5.47}$$

It may be the case an adversary is only able to perform agent only attacks. For example, suppose the information broadcast by an agent to its neighbor is the same information that is sent to a central monitor. In this scenario, there can be no deviation between an agents measured state and broadcasted state. The set of all feasible attacks on agents is given by $\mathcal{F}_x$:

$$\mathcal{F}_x = \{F \subset \mathcal{X} : \ |F| \leq q\}. \tag{5.48}$$

We define the graph $\mathcal{G}_F^a = (\mathcal{V}_F^a, \mathcal{E}_F^a)$ of a DCS when a set F of agents/sensors is compromised.

$$F = \{x_{l_1}, \cdots, x_{l_p}, y_{l_{p+1}}, \cdots, y_{l_{q'}}\}, \ \ (q' \leq q)$$

We introduce attack input vertices $\mathcal{U}_F^a = \{u_1^a, \cdots, u_{q'}^a\}$. We assume there exists directed edges from $\mathcal{U}_F^a$ to F given by

$$\mathcal{E}_{\mathcal{U}_F^a, \mathcal{X}} \triangleq \{(u_1^a, x_{l_1}), \cdots, (u_p^a, x_{l_p})\}$$
$$\mathcal{E}_{\mathcal{U}_F^a, \mathcal{Y}} \triangleq \{(u_{p+1}^a, y_{l_{p+1}}), \cdots, (u_{q_*}^a, y_{l_{q'}})\}$$

We then define $\mathcal{E}_F^a \triangleq \mathcal{E} \cup \mathcal{E}_{\mathcal{U}_F^a, \mathcal{X}} \cup \mathcal{E}_{\mathcal{U}_F^a, \mathcal{Y}}$ and $\mathcal{V}_F^a \triangleq \mathcal{V} \cup \mathcal{U}_F^a$. In the case of agent only attacks $\mathcal{E}_{\mathcal{U}_F^a, \mathcal{Y}}$ is empty.

Algebraic Model: We let $\mathbf{x}_i^a(k)$ represent the state of x_i under attack. If $(u_l^a, x_i) \in \mathcal{E}_F^a$, then the dynamics are

$$\mathbf{x}_i^a(k+1) = a_{ii}\mathbf{x}_i^a(k) + \sum_{j \neq i} a_{ij}\mathbf{x}_j^a(k) + \mathbf{u}_l^a(k), \tag{5.49}$$

where $\mathbf{u}_l^a(k)$ is an input from node u_l^a at time k. If x_i is secure then $\mathbf{u}_l^a(k) = 0$. We define $\mathbf{y}_i^a(k)$ as the output of y_i at time k under attack. If $(u_l^a, y_i) \cup (x_j, y_i) \subset \mathcal{E}_F^a$, then

$$\mathbf{y}_i^a(k) = \mathbf{x}_j^a(k) + \mathbf{u}_l^a(k). \tag{5.50}$$

If y_i is secure then in (5.50), $\mathbf{u}_l^a(k) = 0$. Concatening $\mathbf{x}_i^a(k)$, $\mathbf{y}_i^a(k)$, and $\mathbf{u}_i^a(k)$ into $\mathbf{x}^a(k), \mathbf{y}^a(k)$, and $\mathbf{u}^a(k)$, we have :

$$\mathbf{x}^a(k+1) = A\mathbf{x}^a(k) + B_F^a\mathbf{u}^a(k), \ \ \mathbf{x}^a(0) = \mathbf{x}(0), \tag{5.51}$$

$$\mathbf{y}^a(k) = C\mathbf{x}^a(k) + D_F^a\mathbf{u}^a(k), \tag{5.52}$$

with $B_F^a(i,j) \triangleq \mathbb{I}_{(u_j^a, x_i) \in \mathcal{E}_{\mathcal{U}_F^a, \mathcal{X}}}$, $D_F^a(i,j) \triangleq \mathbb{I}_{(u_j^a, y_i) \in \mathcal{E}_{\mathcal{U}_F^a, \mathcal{Y}}}$. Again, in the case of agent only attacks D_F^a is empty. We assume the attacker knows (A, B_F^a, C, D_F^a). The estimator policy remains unchanged during an attack.

$$\hat{\mathbf{x}}^a(k+1) = (A - KCA)\hat{\mathbf{x}}^a(k) + K\mathbf{y}^a(k+1), \tag{5.53}$$

$$\mathbf{z}^a(k+1) = \mathbf{y}^a(k+1) - CA\hat{\mathbf{x}}^a(k). \tag{5.54}$$

5.3.3 Optimal Unconstrained Network Design of DCS for Detection

We will now consider the minimal design of robust DCS to ensure that such systems are not discreetly attackable. The case for perfectly attackable systems follows similarly. Specifically the posed optimization problems to minimally design DCS to avoid zero dynamics attacks coming from q malicious nodes is equivalent to minimally designing DCS to avoid perfect attacks coming from $q+1$ malicious nodes We consider this a technique of active detection, in that we are intelligently designing our system in order to ensure the detectability of attacks. Unlike our previously discussed methods, this approach for active detection takes place offline.

Communication Design: Agent and Sensor Attacks

We first assume the structure of C, or $[C]$ is given. Here in order to ensure the system is not discreetly attackable, we have at least $q+1$ dedicated sensors. That is, $m \geq q+1$. Let S_i be a minimal vertex separator between (x_i, o) in $f(\mathcal{G})$. We have:

$$\begin{aligned} &\underset{[A]}{\text{minimize}} && \|A\|_0 \\ &\text{subject to} && |S_i| \geq q+1,\ [A](i,i) \neq 0,\ i \in \{1, \ldots, n\}, \end{aligned} \tag{5.55}$$

The objective function represents the number of communication links in our system. The constraint ensures that the system is not discreetly attackable.

Theorem 5.19. *The optimal solution to problem* (5.55) *is* $\|A\|_0^* = m(q+1) + (n-m)(q+2) = n(q+2) - m$.

Proof. We begin by showing that $n(q+2)-m$ is a lower bound of the optimal solution $\|A\|_0^*$. Without loss of generality, assume that $\{x_1, \cdots, x_m\}$ are the set of agents which are observed by $\mathcal{Y}$. Then,

$$\begin{aligned}\|A\|_0^* &= \sum_{k=1}^{m}(|N_{x_k}^O| - 1) + \sum_{k=m+1}^{n} |N_{x_k}^O|, \\ &\geq m(q+1) + (n-m)(q+2). \end{aligned} \tag{5.56}$$

The first equality is obtained by noting that the number of nonzero entries in each row i of A is equal to the out-degree of x_i if the agent x_i is unobserved and equal to the out-degree of x_i minus 1 if it is observed. The last inequality is obtained from the necessary conditions for a system to not be discreetly attackable from Corollary 5.9. Thus $nq+2n-m$ is a lower bound for $\|A\|_0^*$.

We now show that $nq+2n-m$ is an upper bound for $\|A\|_0^*$ by constructing a feasible $[A]$ with a minimal number of edges. To do this we consider the following lemma.

Lemma 5.7. *Consider a realization $([A],[C])$ of a DCS with graph $\mathcal{G}$ where every nontrivial cycle (cycles not containing self-loops) contains an observed agent. Assume there are at least $q+1$ dedicated sensors. Then $([A],[C])$ is an optimal solution to problem (5.55) if and only if each agent x_i has out-degree $q+2$ with $(x_i, x_i) \in \mathcal{E},\ i = 1, \cdots, n$.*

We first show there exists a graph $\mathcal{G}$ which satisfies these assumptions. WLOG we assume that agents $\{x_1, \cdots, x_m\}$ are observed so that there exists a directed edge from x_j to y_j for $j \in \{1, \cdots, m\}$. Next for $j \in \{1, \cdots, m\}$, we have $|N_{x_j}^O| = q+2$ and $N_{x_j}^O \subset \{y_j, x_1, \cdots, x_m\}$. Thus, each observed agent has $q+2$ outgoing edges, 1 to its observer, q edges to other observed agents, and 1 to itself. Finally, for $j \in \{m+1, \cdots, n\}$, we have $d_{x_j}^O = q+2$ and $N_{x_j}^O \subset \{x_1, \cdots, x_m, x_j\}$. Each unobserved agent has $q+1$ neighbors besides itself, all of which are observed. Thus, there are no cycles which only contain unobserved agents.

We next show $\mathcal{G}$ in Lemma 5.7 satisfies the constraints of problem (5.55). By inspection, the graph immediately satisfies $[A](i,i) \neq 0$. We must verify that $|S_i| \geq q+1$. Suppose there exists a vertex separator S_i of (x_i, o) in $f(\mathcal{G})$ where $|S_i| < q+1$. Suppose we remove all vertices S_i

Algorithm 1 Find Path from x_i to o in $f(\mathcal{G})$ when S_i is removed

1: **function** FIND PATH($f(\mathcal{G}), x_i$)
2: $z = x_i, \quad P = x_i.$
3: **if** $(z, y_j) \in \mathcal{E}$ for some $y_j \in \mathcal{V} \setminus S_i$ **then**
4: $P = P, y_j, o.$ **return** P
5: break
6: **end if**
7: **if** $\exists\, x_j, y_k \in \mathcal{V} \setminus S_i$ such that $(z, x_j), (x_j, y_k) \in \mathcal{E}$ **then**
8: $P = P, x_j, y_k, o.$ **return** P
9: break
10: **end if**
11: Find $x_l \in \mathcal{X} \setminus S_i$ such that $(z, x_l) \in \mathcal{E}$ and $(x_l, y_k) \notin \mathcal{E},\ \forall y_k \in \mathcal{Y}$.
12: $z = x_l, \quad P = P, x_l.$
13: Proceed to step 7.
14: **end function**

from $f(\mathcal{G})$. We can construct a path from x_i to o even when vertices from S_i are removed using Algorithm 1.

Since $|S_i| < q+1$ and x_i has $q+1$ outgoing neighbors besides itself, x_i must either be observed by a node $y_j \notin S_i$, have outgoing edge to an observed agent $x_j \notin S_i$ with observer $y_k \notin S_i$, or have outgoing edge to unobserved agent $x_l \notin S_i$. The algorithm terminates successfully if either of the first two conditions hold. Otherwise, the path is extended to the unobserved agent x_l. Since x_l is unobserved, the algorithm proceeds to step 7. The same argument holds for x_l. This process will eventually encounter an observed agent $x_j \notin S_i$ with observer $y_k \notin S_i$ in step 7 since $\mathcal{G}$ is finite and every cycle must contain an observed agent. Consequently, this process will eventually terminate and give a path P from x_i to o. Thus, S_i is not a vertex separator and $\mathcal{G}$ is feasible.

We now show $\mathcal{G}$ constructed with the rules presented in Lemma 5.7 is optimal. We note that each agent has out-degree $q+2$. Thus, from (5.56) we have $\|A\|_0 = nq+2n-m$. Since $nq+2n-m$ is a lower bound for the number of edges in an optimal solution, $\mathcal{G}$ is an optimal solution. □

We can see in an optimal solution that each observed agent has q agent neighbors besides itself and each unobserved agent has $q+1$ agent neighbors besides itself. This is clearly required for each agent to have $q+1$ disjoint paths to the output. The prior result however also shows that no

additional edges are required.

Communication Design: Agent Attacks

We now consider the scenario of agent only attacks and obtain similar results to the agent and sensor attack case. Again, assume the structure of C, or $[C]$ is given. Here in order to ensure the system is not discreetly attackable, we have at least $q+1$ dedicated sensors. That is, $m \geq q+1$. Let S_i be a minimal vertex separator between (x_i, o) in $f'(\mathcal{G})$ as considered in the previous section where x_i is unobserved. We have:

$$\begin{aligned} \underset{[A]}{\text{minimize}} \quad & \|A\|_0 \\ \text{subject to} \quad & [A](i,i) \neq 0, \ i \in \{1, \ldots, n\}, \\ & |S_i| \geq q+1, \ \ \forall\, i \text{ s.t. } \theta(x_i, \mathcal{Y}) = 0. \end{aligned} \tag{5.57}$$

The objective function represents the number of communication links in our system. The constraint ensures that the system is not discreetly attackable. Here, we make use of the results in Lemma 5.6 which state that minimum vertex separator between unobserved x_i and o in $f(\mathcal{G})$ has size r if and only if that minimum vertex separator between x_i and o in $f'(\mathcal{G})$ has size r.

Theorem 5.20. *The optimal solution to problem* (5.57) *is* $\|A\|_0^* = m + (n-m)(q+2)$.

Proof. We begin by showing that $m+(n-m)(q+2)$ is a lower bound of the optimal solution $\|A\|_0^*$. Without loss of generality, assume that $\{x_1, \cdots, x_m\}$ are the set of agents which are observed by $\mathcal{Y}$. Then,

$$\begin{aligned} \|A\|_0^* &= \sum_{k=1}^{m} (|N_{x_k}^O| - 1) + \sum_{k=m+1}^{n} |N_{x_k}^O|, \\ &\geq m + (n-m)(q+2). \end{aligned} \tag{5.58}$$

The first equality is obtained by noting that the number of nonzero entries in each row i of A is equal to the out-degree of x_i if the agent x_i is unobserved and equal to the out-degree of x_i minus

1 if it is observed. The last inequality is obtained from the necessary conditions for a system to not be discreetly attackable from Corollary 5.11. Thus $m + (n-m)(q+2)$ is a lower bound for $\|A\|_0^*$.

We now show that $m + (n-m)(q+2)$ is an upper bound for $\|A\|_0^*$ by constructing a feasible $[A]$ with a minimal number of edges. To do this we consider the following lemma.

Lemma 5.8. *Consider a realization* $([A], [C])$ *of a DCS with graph* $\mathcal{G}$ *where every nontrivial cycle (cycles not containing self-loops) contains an observed agent or an agent with a directed edge to an observed agent. Assume there are at least* $q+1$ *dedicated sensors. Then* $([A], [C])$ *is an optimal solution to problem* (5.57) *if and only if each unobserved agent* x_i *has out-degree* $q+2$ *and each observed agent* x_i *has out-degree equal to* 2 *with* $(x_i, x_i) \in \mathcal{E},\ i = 1, \cdots, n$.

We first show there exists a graph $\mathcal{G}$ which satisfies these assumptions. WLOG we assume that agents $\{x_1, \cdots, x_m\}$ are observed so that there exists a directed edge from x_j to y_j for $j \in \{1, \cdots, m\}$. Next for $j \in \{1, \cdots, m\}$, we have $|N_{x_j}^O| = 2$ and $N_{x_j}^O = \{y_j, x_j\}$. Thus, each observed agent has 2 outgoing edges, 1 to its observer and 1 to itself. Finally, for $j \in \{m+1, \cdots, n\}$, we have $d_{x_j}^O = q+2$ and $N_{x_j}^O \subset \{x_1, \cdots, x_m, x_j\}$. Each unobserved agent has $q+1$ neighbors besides itself, all of which are observed. Thus, there are no nontrivial cycles.

We next show $\mathcal{G}$ in Lemma 5.8 satisfies the constraints of problem (5.57). By inspection, the graph immediately satisfies $[A](i,i) \neq 0$. We must verify that $|S_i| \geq q+1$ for unobserved agents x_i in $f'(\mathcal{G})$. Suppose there exists a vertex separator S_i of (x_i, o) in $f'(\mathcal{G})$ where $|S_i| < q+1$. Suppose we remove all vertices S_i from $f'(\mathcal{G})$. We can construct a path from x_i to o even when vertices from S_i are removed using Algorithm 2.

Since $|S_i| < q+1$ and x_i has $q+1$ outgoing neighbors besides itself, x_i must either have an outgoing edge to an observed node $x_j \notin S_i$, have an outgoing edge to an unobserved agent $x_j \notin S_i$ that has outgoing edge to observed agent $x_k \notin S_i$, or have outgoing edge to unobserved agent $x_l \notin S_i$ that has no outgoing edge to an observed agent. The algorithm terminates successfully if either of the first two conditions hold. Otherwise, the path is extended to the unobserved agent x_l. Since x_l is unobserved, and has no outgoing edge to an observed agent the algorithm proceeds

Algorithm 2 Find Path from x_i to o in $f'(\mathcal{G})$ when S_i is removed

function FIND PATH($f'(\mathcal{G}), x_i$)
  $z = x_i, \quad P = x_i.$
  if $(z, x_j) \in \mathcal{E}$ for some observed $x_j \in \mathcal{X} \setminus S_i$ **then**
    $P = P, x_j, o.$ **return** P
    break
  end if
  if $\exists\, x_j, x_k \in \mathcal{X} \setminus S_i$ such that $(z, x_j), (x_j, x_k) \in \mathcal{E}$ and x_k is observed **then**
    $P = P, x_j, x_k, o.$ **return** P
    break
  end if
  Find $x_l \in \mathcal{V} \setminus S_i$ such that $(z, x_l) \in \mathcal{E}$ and $(x_l, x_k) \notin \mathcal{E},\ \forall$ observed $x_k \in \mathcal{X}$.
  $z = x_l, \quad P = P, x_l.$
  Proceed to step 7.
end function

to step 7. The same argument holds for x_l. This process will eventually encounter an unobserved agent $x_j \notin S_i$ with outgoing edge to observed agent $x_k \notin S_i$ in step 7 since $\mathcal{G}$ is finite and every cycle must contain an observed agent or an unobserved agent with directed edge to an observed agent. Consequently, this process will eventually terminate and give a path P from x_i to o. Thus, S_i is not a vertex separator and $\mathcal{G}$ is feasible.

We now show $\mathcal{G}$ constructed with the rules presented in Lemma 5.7 is optimal. We note that each unobserved agent has out-degree $q+2$ and each observed agent has out-degree 2. Thus, from (5.58) we have $\|A\|_0 = m + (n-m)(q+2)$. Since $m + (n-m)(q+2)$ is a lower bound for the number of edges in an optimal solution, $\mathcal{G}$ is an optimal solution. □

As expected, in a system with agent only attacks, fewer communication edges are required. Observed agents only need to communicate to sensors while unobserved agents will have exactly $q+1$ neighbors besides themselves. We note that while the general graphical solution to both (5.55) and (5.57) is unknown, Lemmas 5.7 and 5.8 gives us the structure for optimal graphs in specially defined scenarios.

5.3.4 Optimal Unconstrained Joint Design of DCS for Detection

Instead of fixing the number of sensors m under consideration, the number of sensors can be a design variable which is chosen concurrently with the network. We can alter the optimization problem to consider this as follows.

Joint Design: Agent and Sensor Attacks

Let S_i be a minimal vertex separator between x_i and o in $f(\mathcal{G})$. The joint design problem is given as follows.

$$\begin{aligned} \underset{[A],[C]}{\text{minimize}} \quad & \alpha_1 \|A\|_0 + \alpha_2 m \\ \text{subject to} \quad & |S_i| \geq q+1,\ [A](i,i) \neq 0,\ i \in \{1,\dots,n\}, \\ & C \in \mathbb{R}^{m \times n},\ m \in \{q+1, \cdots, n\}, \\ & \|C_j\|_0 \leq 1,\ j \in \{1,\dots,n\}, \\ & \|C^t\|_0 = 1,\ t \in \{1,\dots,m\}. \end{aligned} \tag{5.59}$$

The last three constraints convey that $[C]$ implements a set of m dedicated sensors where $m \in \{q+1, \cdots, n\}$.

Theorem 5.21. *Consider problem (5.59). If $\alpha_1 > \alpha_2$. Then every agent should be observed ($m = n$). Alternatively, if $\alpha_2 > \alpha_1$, then $m = q+1$. Finally, if $\alpha_1 = \alpha_2$, then m can be chosen arbitrarily from $\{q+1, \cdots, n\}$*

Proof. For a fixed set of dedicated sensors, we can solve (5.55) to obtain the joint solution. Since $\|A\|_0^* = (q+2)n - m$ for a fixed set of sensors, the optimal value of (5.59) is $(\alpha_2 - \alpha_1)m + \alpha_1(q+2)n$. The result follows. □

If $\alpha_1 > \alpha_2$, so that communication is more costly than sensing, it is optimal to observe all agents. If $\alpha_2 > \alpha_1$ so sensing is more costly than communication, it is optimal to observe the fewest number of sensors that enable a robust solution, which is $q+1$. Roughly speaking the prior

result is based on the idea that in sensor and agent attacks, the combined number of communication edges and sensing nodes is fixed and equal to $(q+2)n$. Thus, if sensing cost more, we want to have as few sensors as possible contribute to this fixed quantity. Likewise if communication costs more, we want to have as few links as possible contribute to this fixed quantity.

An optimal graphical solution can be obtained by first selecting an arbitrary set of observed nodes and then constructing a feasible graphical solution, for instance as is described in the proof of Lemma 5.7.

Joint Design: Agent Attacks

In the case of agent only attacks, it can be shown that sensing must be significantly more costly than communication to justify additional communication links. The joint design problem is given by:

$$\begin{aligned} \underset{[A],[C]}{\text{minimize}} \quad & \alpha_1 \|A\|_0 + \alpha_2 m \\ \text{subject to} \quad & [A](i,i) \neq 0,\ i \in \{1,\dots,n\}, \\ & |S_i| \geq q+1,\ \ \forall\, i \text{ s.t. } \theta(x_i, \mathcal{Y}) = 0, \\ & C \in \mathbb{R}^{m \times n},\ m \in \{q+1, \cdots, n\}, \\ & \|C_j\|_0 \leq 1,\ j \in \{1,\dots,n\}, \\ & \|C^t\|_0 = 1,\ t \in \{1,\dots,m\}. \end{aligned} \tag{5.60}$$

where S_i is a vertex separator between unobserved x_i and o in $f'(\mathcal{G})$.

Theorem 5.22. *Consider problem* (5.59). *If* $(q+1)\alpha_1 > \alpha_2$. *Then every agent should be observed* $(m = n)$. *Alternatively, if* $\alpha_2 > (q+1)\alpha_1$, *then* $m = q+1$. *Finally, if* $(q+1)\alpha_1 = \alpha_2$, *then* m *can be chosen arbitrarily from* $\{q+1, \cdots, n\}$

Proof. For a fixed set of dedicated sensors, we can solve (5.60) to obtain the joint solution. Since $\|A\|_0^* = (q+2)(n-m) + m$ for a fixed set of sensors, the optimal value of (5.60) is $(\alpha_2 - (q+1)\alpha_1)m + \alpha_1(q+2)n$. The result follows. □

Unlike the joint sensor and agent attacks, if all agents are observed, no additional communication is required. Indeed, in this case $\|A\|_0^* = n$, where the remaining edges are the self loops. The significant difference in agent only attacks is that the combined number of links and sensors is no longer fixed. Rather, an observed agent only has a self loop and an edge to a sensor, while an unobserved has a self loop and $q+1$ other agent neighbors. Thus, one must determine which is more expensive, $q+1$ communication links or 1 sensor to determine an optimal joint solution.

An optimal graphical solution can be obtained by first selecting an arbitrary set of observed nodes and then constructing a feasible graphical solution, for instance as is described in the proof of Lemma 5.8.

5.3.5 Constrained Optimization of Communication of DCS for Detection

In the previous subsection, we found minimal designs of systems which prevent all possible zero dynamics attacks. In these problems, we assumed that there were no restrictions among which agents can communicate. In practice, due to physical constraints, certain agents may not be able to communicate. We assume constraints on communication are encoded into $[\bar{A}]$ where agent x_i can speak to agent x_j if and only if $[\bar{A}](j,i) \neq 0$. Given a set of observers $[C]$, we formulate a problem to robustly minimize the amount of communication among agents subject to constraints given by $[\bar{A}]$. We demonstrate that introducing these constraints does not change the optimal number of links in a system.

Constrained Design: Agent and Sensor Attacks

Let S_i be a vertex separator between x_i and o in $f(\mathcal{G})$.

$$\begin{aligned} \underset{[A]}{\text{minimize}} \quad & \|A\|_0 \\ \text{subject to} \quad & |S_i| \geq q+1,\ [A](i,i) \neq 0,\ i \in \{1,\dots,n\}, \\ & [\bar{A}](u,v) = 0 \implies [A](u,v) = 0,\ u,v \in \{1,\dots,n\}. \end{aligned} \tag{5.61}$$

Algorithm 3 Constrained Optimization of DCS

1: **function** OPTIMIZATION(($[\bar{A}], [C]$))
2: Let graph $\mathcal{G}$ be generated from $[\bar{A}], [C]$, $[A] = [\bar{A}]$.
3: **while** $\|A\|_0 > nq + 2n - m$ **do**
4: Find an edge $(x_i, x_{i'})$ whose removal still ensures there are no zero dynamics attacks on $\mathcal{G}$.
5: $\mathcal{G} = \mathcal{G} - (x_i, x_{i'}),\ [A]_{i'i} = 0$.
6: **end while**
7: **return** $[A]$
8: **end function**

We now obtain the following result related to problem (5.61) which states that if the problem is feasible, there always exists a solution to problem (5.61) which is also a solution to the unconstrained optimization problem (5.55).

Theorem 5.23. *Suppose there exists a feasible solution to problem* (5.61). *Then, the optimal solution to problem 5.61 satisfies* $\|A\|_0^* = nq + 2n - m$.

Proof. We argue that Algorithm 3 can be used to obtain an optimal solution to problem (5.61). It suffices to show step 4 is feasible for an arbitrary $\mathcal{G}$ which is not discreetly attackable and is non-minimal. To do this, we observe there must exist an agent x_i with out-degree $|N_{x_i}^O| > q + 2$ if the system is non-minimal. Since the system is not discreetly attackable, there exists at least $q + 1$ disjoint paths from x_i to o. Because x_i has out-degree greater than $q + 2$, there exists an edge $(x_i, x_{i'})$ whose removal ensures x_i still has $q + 1$ disjoint paths to o so that $|S_i| \geq q + 1$ in $f(\mathcal{G}) - (x_i, x_{i'})$. Indeed, construct $q + 1$ disjoint paths from x_i to o. Without loss of generality assume x_i is not in one of these $q + 1$ disjoint paths. There must exists an outgoing neighbor of x_i, which is not in these disjoint paths. If this neighbor is an agent x_j, one can delete (x_i, x_j) and still have $q + 1$ disjoint paths to o. If the only remaining neighbor (not in this set of $q + 1$ paths) is an observer , one can remove an arbitrary agent neighbor and there will still be $q + 1$ disjoint paths to o.

Now consider arbitrary x_j not equal to x_i in $\mathcal{G}$. We must show that $|S_j| \geq q + 1$ where S_j is a minimum vertex separator of x_j and o in $f(\mathcal{G}) - (x_i, x_{i'})$. Suppose $|S_j| < q + 1$. We

observe that $\{S_j, x_i\}$ is a vertex separator of x_j and o in $f(\mathcal{G})$. Since $\mathcal{G}$ is not discreetly attackable, $|\{S_j, x_i\}| \geq q+1$. Consequently, $x_i \notin S_j$, $|S_j| = q$, and $\{S_j, x_i\}$ is a minimal vertex separator of (x_j, o) in $f(\mathcal{G})$.

Lets remove S_j from $f(\mathcal{G}) - (x_i, x_{i'})$. We first argue there must still be a path from x_j to x_i. Suppose instead that removing S_j from $f(\mathcal{G}) - (x_i, x_{i'})$ deletes all paths from x_j to x_i. Then, removing S_j from $f(\mathcal{G})$ deletes all paths from x_j to x_i in $f(\mathcal{G})$. However, since $\{S_j, x_i\}$ is a minimal vertex separator of x_j and o in $f(\mathcal{G})$, removing S_j from $f(\mathcal{G})$ would mean there still exists a path from x_j to o containing x_i. By contradiction, there must still be a path from x_j to x_i after deleting S_j from $f(\mathcal{G}) - (x_i, x_{i'})$.

We now show there exists a path from x_i to o after removing S_j from $f(\mathcal{G}) - (x_i, x_{i'})$. By assumption, there are at least $q+1$ disjoint paths from x_i to o in $f(\mathcal{G}) - (x_i, x_{i'})$. Deleting S_j, which has q vertices, can remove at most q paths. Thus, there is still a path from x_i to o.

As a result, even after deleting S_j from $f(\mathcal{G}) - (x_i, x_{i'})$, there exists a path from x_j to x_i and a path from x_i to o. Consequently, there exists a path from x_j to o so that S_j is not a vertex separator. Thus, by contradiction, any vertex separator S_j of (x_j, o) in $f(\mathcal{G}) - (x_i, x_{i'})$ satisfies $|S_j| \geq q+1$. Therefore, $\mathcal{G} - (x_i, x_{i'})$ is not discreetly attackable and step 4 is feasible. □

Theorem 5.23 shows we can obtain a minimal network resilient to zero dynamics attacks even with constraints on communication. While Algorithm 3 gives a method to construct such an optimal communication network, the method and complexity of this approach is unclear. Nonetheless, if we can compute a maximum set of vertex disjoint paths from a vertex x_i to o, we can determine outgoing neighbors of agent x_i which should not be deleted. In particular, we should keep edges from x_i to $q+1$ neighbors through which there exists $q+1$ disjoint paths to o, with the condition that none of these paths should contain x_i as an intermediate vertex. We use Dinic's algorithm in Algorithm 4 to solve problem (5.61). The worst case complexity is less than $O(n(2|\mathcal{V}|)^{\frac{1}{2}}(|\mathcal{E}'| + |\mathcal{V}| - 1))$ where $\mathcal{V}$ and $\mathcal{E}'$ are associated with matrices $[\bar{A}], [C]$. While not considered here, it will be interesting to evaluate the scenario where links are not restricted to have the same cost. We might not be able to

Algorithm 4 Practical Solution to Constrained Optimization of DCS

function OPTIMIZATION($([\bar{A}], [C])$)
  Let graph $\mathcal{G}$ be generated from $[\bar{A}]$, $[C]$, $[A] = [\bar{A}]$.
  for $i = 1 : n$ **do**
    if $|N_{x_i}^O| > q + 2$ **then**
      Solve maximum flow by using Dinic's algorithm on $h^i(f(\mathcal{G}))$ from source x_i to sink o
      If x_i is observed (or unobserved), keep q (or respectively $q+1$) neighbors in $\mathcal{X}$ through which $\exists$ a maximum flow. Delete edges to other outgoing neighbors in $\mathcal{X} - x_i$
      Update $\mathcal{G}$, $[A]$
    end if
  end for
return $[A]$
end function

solve such a problem optimally, though some sort of greedy algorithms may be used.

Constrained Design: Agent Attacks

Let S_i be a vertex separator between x_i and o in $f'(\mathcal{G})$.

$$\begin{aligned} \underset{[A]}{\text{minimize}} \quad & \|A\|_0 \\ \text{subject to} \quad & [A](i,i) \neq 0, \ i \in \{1,\dots,n\}, \\ & |S_i| \geq q+1, \ \ \forall\, i \text{ s.t. } \theta(x_i, \mathcal{Y}) = 0, \\ & [\bar{A}](u,v) = 0 \implies [A](u,v) = 0, \ u,v \in \{1,\dots,n\}. \end{aligned} \tag{5.62}$$

We now obtain the following result related to problem (5.62) which states that if the problem is feasible, there always exists a solution to problem (5.62) which is also a solution to the unconstrained optimization problem (5.57).

Theorem 5.24. *Suppose there exists a feasible solution to problem* (5.62). *Then, the optimal solution to problem 5.62 satisfies* $\|A\|_0^* = m + (n-m)(q+2)$.

Proof. We argue that Algorithm 5 can be used to obtain an optimal solution to problem (5.62). It suffices to show step 4 is feasible for an arbitrary $\mathcal{G}$ which is not discreetly attackable and is

Algorithm 5 Constrained Optimization of DCS

function OPTIMIZATION($([\bar{A}], [C])$)
  Let graph $\mathcal{G}$ be generated from $[\bar{A}]$, $[C]$, $[A] = [\bar{A}]$.
  while $\|A\|_0 > m + (n-m)(q+2)$ **do**
    Find an edge $(x_i, x_{i'})$ whose removal still ensures there are no zero dynamics attacks on $\mathcal{G}$.
    $\mathcal{G} = \mathcal{G} - (x_i, x_{i'}),\ [A]_{i'i} = 0.$
  end while
return $[A]$
end function

non-minimal. To do this, we observe there must exist an unobserved agent x_i with out-degree $|N_{x_i}^O| > q+2$ or an observed agent x_i with out-degree $|N_{x_i}^O| > 2$ if the system is non-minimal. Select such an x_i. If x_i is unobserved there exists at least $q+1$ disjoint paths from x_i to o. Because x_i has out-degree greater than $q+2$, there exists an edge $(x_i, x_{i'})$ whose removal ensures x_i still has $q+1$ disjoint paths to o so that $|S_i| \geq q+1$ in $f'(\mathcal{G}) - (x_i, x_{i'})$. If x_i is observed and has out-degree greater than 2, we can delete an arbitrary edge $(x_i, x_{i'})$.

Now consider arbitrary unobserved x_j not equal to x_i in $\mathcal{G}$. We must show that $|S_j| \geq q+1$ where S_j is a minimum vertex separator of x_j and o in $f'(\mathcal{G}) - (x_i, x_{i'})$. Suppose $|S_j| < q+1$. We observe that $\{S_j, x_i\}$ is a vertex separator of x_j and o in $f'(\mathcal{G})$. Since $\mathcal{G}$ is not discreetly attackable, $|\{S_j, x_i\}| \geq q+1$. Consequently, $x_i \notin S_j$, $|S_j| = q$, and $\{S_j, x_i\}$ is a minimal vertex separator of (x_j, o) in $f'(\mathcal{G})$.

Lets remove S_j from $f'(\mathcal{G}) - (x_i, x_{i'})$. We first argue there must still be a path from x_j to x_i. Suppose instead that removing S_j from $f'(\mathcal{G}) - (x_i, x_{i'})$ deletes all paths from x_j to x_i. Then, removing S_j from $f'(\mathcal{G})$ deletes all paths from x_j to x_i in $f'(\mathcal{G})$. However, since $\{S_j, x_i\}$ is a minimal vertex separator of x_j and o in $f'(\mathcal{G})$, removing S_j from $f'(\mathcal{G})$ would mean there still exists a path from x_j to o containing x_i. By contradiction, there must still be a path from x_j to x_i after deleting S_j from $f'(\mathcal{G}) - (x_i, x_{i'})$.

We now show there exists a path from x_i to o after removing S_j from $f'(\mathcal{G}) - (x_i, x_{i'})$. If x_i is unobserved there are at least $q+1$ disjoint paths from x_i to o in $f'(\mathcal{G}) - (x_i, x_{i'})$. Deleting S_j,

Algorithm 6 Practical Solution to Constrained Optimization of DCS with Agent Attacks

function OPTIMIZATION($([\bar{A}], [C])$)
  Let graph $\mathcal{G}$ be generated from $[\bar{A}]$, $[C]$, $[A] = [\bar{A}]$.
  for $i = 1 : n$ **do**
    if x_i is unobserved **then**
      if $|N^O_{x_i}| > q + 2$ **then**
        Solve maximum flow by using Dinic's algorithm on $h^i_l(f'(\mathcal{G}))$ from source x_i to sink o
        Keep $q + 1$ neighbors in $\mathcal{X}$ through which $\exists$ a maximum flow from x_i to o. Delete edges to other outgoing neighbors in $\mathcal{X} - x_i$
        Update $\mathcal{G}$, $[A]$
      end if
    end if
    if x_i is observed **then**
      if $|N^O_{x_i}| > 2$ **then**
        Delete edges to outgoing neighbors in $\mathcal{X} - x_i$
      end if
    end if
  end for
return $[A]$
end function

which has q vertices, can remove at most q paths. Thus, there is still a path from x_i to o. If x_i is observed, there is a directed edge from x_i to o.

As a result, even after deleting S_j from $f'(\mathcal{G}) - (x_i, x_{i'})$, there exists a path from x_j to x_i and a path from x_i to o. Consequently, there exists a path from x_j to o so that S_j is not a vertex separator. Thus, by contradiction, any vertex separator S_j of (x_j, o) in $f'(\mathcal{G}) - (x_i, x_{i'})$ satisfies $|S_j| \geq q + 1$ if x_j is unobserved. Therefore, $\mathcal{G} - (x_i, x_{i'})$ is not discreetly attackable and step 4 is feasible. □

Theorem 5.24 shows we can obtain a minimal network resilient to zero dynamics attacks even with constraints on communication. We can again use Dinic's algorithm in Algorithm 6 to solve problem (5.62). The worst case complexity is less than $O((n-m)(2|\mathcal{X}|)^{\frac{1}{2}}(|\mathcal{E}| + |\mathcal{V}| - 1))$ where $\mathcal{X}$ and $\mathcal{E}$ are associated with matrices $[\bar{A}]$, $[C]$.

5.3.6 Joint Constrained Optimization of DCS for Detection

Since the constrained optimal solution, is also unconstrained optimal if it exists, the results in the joint constrained case are similar to the joint unconstrained case. A significant deviation however occurs when a solution calls us to minimize the number of sensors.

Joint Constrained Design: Agent and Sensor Attacks

Let S_i be a minimal vertex separator between x_i and o in $f(\mathcal{G})$. The joint constrained design problem is given as follows.

$$\begin{aligned} \underset{[A],[C]}{\text{minimize}} \quad & \alpha_1 \|A\|_0 + \alpha_2 m \\ \text{subject to} \quad & |S_i| \geq q+1,\ [A](i,i) \neq 0,\ i \in \{1,\dots,n\}, \\ & [\bar{A}](u,v) = 0 \implies [A](u,v) = 0,\ u,v \in \{1,\dots,n\}, \\ & C \in \mathbb{R}^{m \times n},\ m \in \{q+1,\cdots,n\}, \\ & \|C_j\|_0 \leq 1,\ j \in \{1,\dots,n\}, \\ & \|C^t\|_0 = 1,\ t \in \{1,\dots,m\}. \end{aligned} \tag{5.63}$$

Theorem 5.25. *Consider problem (5.63). Suppose there exists a feasible solution. That is $[\bar{A}], [C]$ is not discreetly attackable. If $\alpha_1 > \alpha_2$. Then every agent should be observed $(m = n)$. Alternatively, if $\alpha_2 > \alpha_1$, then $m = q^*$ where q^* is the fewest number of sensors for which Problem (5.61) is feasible. Finally, if $\alpha_1 = \alpha_2$, then m can be chosen arbitrarily from $\{q^*, \cdots, n\}$*

Proof. For a fixed set of dedicated sensors, we can solve (5.61) to obtain the joint solution. Since $\|A\|_0^* = (q+2)n - m$ for a fixed set of sensors, the optimal value of (5.59) is $(\alpha_2 - \alpha_1)m + \alpha_1(q+2)n$. The result follows. □

Again, if $\alpha_1 > \alpha_2$, so that communication is more costly the sensing, it is optimal to observe all sensors. If $\alpha_2 > \alpha_1$, we must first obtain a set of dedicated sensors $[C^*]$ with $C^* \in \mathbb{R}^{q^* \times n}$ which makes Problem (5.61) feasible. Given C^*, Problem (5.63) can be solved using Problem (5.61).

We note that determining q^* is a combinatorial problem. Future work aims to discover efficient solutions.

Joint Constrained Design: Agent Attacks

In the case of agent only attacks, we have

$$\begin{aligned}
\underset{[A]}{\text{minimize}} \quad & \alpha_1 \|A\|_0 + \alpha_2 m \\
\text{subject to} \quad & [A](i,i) \neq 0,\ i \in \{1,\dots,n\}, \\
& |S_i| \geq q+1,\ \ \forall\, i \text{ s.t. } \theta(x_i, \mathcal{Y}) = 0, \\
& [\bar{A}](u,v) = 0 \implies [A](u,v) = 0,\ u,v \in \{1,\dots,n\}, \\
& C \in \mathbb{R}^{m \times n},\ m \in \{q+1, \cdots, n\}, \\
& \|C_j\|_0 \leq 1,\ j \in \{1,\dots,n\}, \\
& \|C^t\|_0 = 1,\ t \in \{1,\dots,m\}.
\end{aligned} \tag{5.64}$$

where S_i is a vertex separator between unobserved x_i and o in $f'(\mathcal{G})$.

Theorem 5.26. *Consider problem* (5.64). *If* $(q+1)\alpha_1 > \alpha_2$. *Then every agent should be observed* $(m = n)$. *Alternatively, if* $\alpha_2 > (q+1)\alpha_1$, *then* $m = q^*$ *where* q^* *is the fewest number of sensors for which Problem* (5.62) *is feasible. Finally, if* $(q+1)\alpha_1 = \alpha_2$, *then* m *can be chosen arbitrarily from* $\{q^*, \cdots, n\}$

Proof. For a fixed set of dedicated sensors, we can solve (5.62) to obtain the joint solution. Since $\|A\|_0^* = (q+2)(n-m) + m$ for a fixed set of sensors, the optimal value of (5.64) is $(\alpha_2 - (q+1)\alpha_1)m + \alpha_1(q+2)n$. The result follows. □

Again, if $\alpha_2 > (q+1)\alpha_1$, we must first obtain a set of dedicated sensors $[C^*]$ with $C^* \in \mathbb{R}^{q^* \times n}$ which makes Problem (5.62) feasible. Given C^*, Problem (5.64) can be solved using Problem (5.62). Again determining q^* is a combinatorial problem.

In the case that $q = 0$ for zero dynamics attacks, we can efficiently determining the minimum number of sensors required and thus obtain an efficient solution. Note that $q = 0$ for zero dynamics attacks corresponds to solving the problem for $q = 1$ and perfect attacks. Thus, this solution is relevant.

Strongly Connected Component Decomposition

We first consider the graph $\mathcal{G}^{\mathcal{X}} = (\mathcal{X}, \mathcal{E}_{\mathcal{X},X})$ obtained by removing all observers and only considering the structural system associated with $[\bar{A}]$. The digraph $\mathcal{G}^{\mathcal{X}}$ is strongly connected if there is a path between any pair of vertices. Moreover, a strongly connected component (SCC) is a maximum subgraph of $\mathcal{G}^{\mathcal{X}}$, that is strongly connected.

It is noted that any digraph can be uniquely decomposed into disjoint SCCs. Moreover, we can represent such a decomposition using a directed acyclic graph (DAG), that is, a graph without cycles [76]. A supernode in such a graph corresponds to a single SCC and there exists a directed edge between two SCCs if and only if there exists an edge between vertices belonging to the corresponding SCCs. We say that an SCC is non-bottom linked if there is no outgoing directed edge from that SCC to another SCC. Otherwise it is bottom linked. Let $\mathcal{G}_S^{\mathcal{X}} = (\mathcal{V}_S, \mathcal{E}_S)$ denote the DAG obtained from the SCC decomposition of $\mathcal{G}^{\mathcal{X}}$. We can obtain the DAG in $O(|\mathcal{X}| + |\mathcal{E}_{\mathcal{X},X}|)$ time complexity [77].

Case q = 1

Given the SCC decomposition of $\mathcal{G}^{\mathcal{X}}$ we can characterize the number of observers needed to ensure structural left invertibility when the defender must be resilient to $q = 1$ attackers. In particular we have the following result.

Theorem 5.27. *The minimum number of observers needed for $[\bar{A}]$ to avoid being perfectly attackable when $q = 1$ is given by the number of non-bottom linked SCCs in $\mathcal{G}_S^{\mathcal{X}}$.*

Proof. To avoid perfect attacks when $q = 1$, there must exist at least one directed path from every node to an observer. This holds both for agent attacks, and agent and sensor attacks. We first argue that each non-bottom linked SCC requires one unique observer. Suppose instead that a non-bottom linked SCC $X_1 \in \mathcal{V}_S$ does not have an observer. Let $x_i \in X_1$. There must be a directed path from x_i to an observer. However, since X_1 has no outgoing edges to another SCC, such a path can not exist. Thus, the number of non-bottom linked SCCs in $\mathcal{G}_S^{\mathcal{X}}$ is a lower bound on the number of observers needed.

We next show that there exists a system which is not perfectly attackable with a number of observers equal to the number of non-bottom linked SCCs in $\mathcal{G}_S^{\mathcal{X}}$. To do this, we arbitrarily assign an observer to each non-bottom linked SCC. Suppose x_i is in a non-bottom linked SCC. Since the SCC is strongly connected, there exists a path from x_i to an observer. Suppose instead that x_i is in a bottom linked SCC. We observe that in $\mathcal{G}_S^{\mathcal{X}}$, there must exist a path from a bottom linked SCC $X_j \in \mathcal{V}_S$ to a non-bottom linked SCC $X_l \in \mathcal{V}_S$. If not, $\mathcal{G}_S^{\mathcal{X}}$ contains a cycle. However, by construction [76], $\mathcal{G}_S^{\mathcal{X}}$ is an acyclic graph. Thus, there exists a path from a bottom linked SCC to a non-bottom linked SCC. This implies that there exists a directed path from x_i to an observer. As a result, the system is not perfectly attackable. □

In the case that $q = 1$ for perfect attacks, the above theorem states that the fewest number of observers needed is equal to the number of non-bottom linked SCCs. This allows us to solve the joint constrained problem for design in (5.63) and (5.64) when $q = 1$ for perfect attacks by first obtaining a minimum sensor placement and then solving (5.61) or (5.62).

5.3.7 Examples

Illustrative Example

We provide an illustrative example which shows how we obtain the solution of Problem 5.61 based on Algorithm 4. Consider a 6-state system measured by 3 sensors, as depicted in Fig. 5.2. The graphical representation of the constraint matrix $[\bar{A}]$ is depicted in Fig. 5.2(a) with self loops

n	r	$\|[A]\|_0$	q	m	$\|A\|_0^*$	Runtime (sec)
100	0.15	732	1	10	290	425.58
100	0.2	1080	1	10	290	776.31
100	0.3	2120	1	10	290	1766.97
100	0.2	1070	2	15	385	768.49
100	0.2	1038	3	20	480	682.13
50	0.2	232	1	10	140	25.11
150	0.2	2536	1	10	440	1.1430×10^4

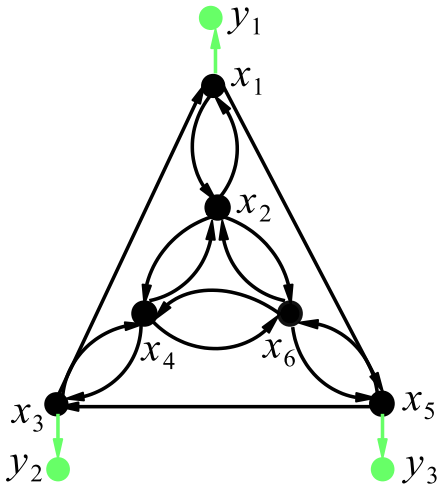
Table 5.1: Runtime of Algorithm 4 for different n, q, r parameters to obtain Minimal Constrained DCS Network Design.

abstracted away. If $[\bar{A}](u, v)$ is not a fixed zero, there exists an edge (x_v, x_u). Suppose the goal is to design an optimal communication network which prevents all perfect attacks when $q = 2$ and all zero dynamics attacks when $q = 1$. Recalling Algorithm 4, we start with the digraph associated with $[\bar{A}]$, and for each of the state vertices x_i we keep enough outgoing agent neighbors to ensure the size of the minimum vertex separator between (x_i, o) is $q + 1$ (to ensure the system is not discreetly attackable) or q (to ensure the system is not perfectly attackable). Figs. 5.2(b)-5.2(d) show the results of these iterations.

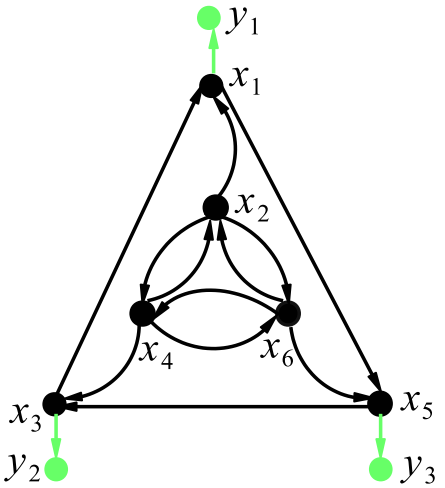
Formation Control

Consider a multi-agent system with n agents, where the agents are able to locally communicate with each other. The goal of formation control could be organizing the agents according to certain 2-D formations. In the simulation, we generated an $n \times 2$ matrix of random variables under uniform distribution $U[0, 1]$, which represent the initial location of n agents.

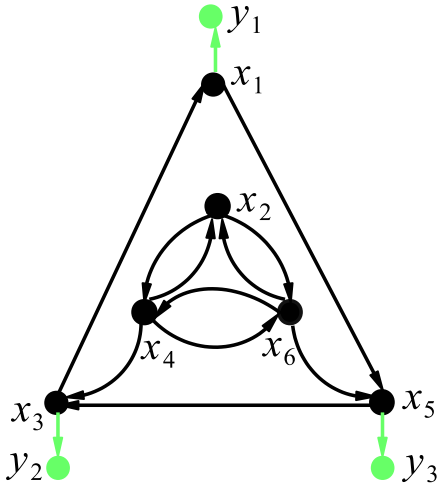
We again consider problem (5.61). Due to communication cost and noise, the communications between agents are restricted to a certain radius r. As a result, we can compute the constraint matrix $[\bar{A}]$ by enumerating the distance between every pair of agents. More precisely, if the distance between the i-th agent and j-th agent is less than r, then $[\bar{A}](i, j) = [\bar{A}](i, j) \neq 0$. Otherwise, $[\bar{A}](i, j) = [\bar{A}](i, j) = 0$. Under such a constraint matrix, the goal is to design a minimum communication network $[A]$, which prevents zero dynamics attacks (the defender does



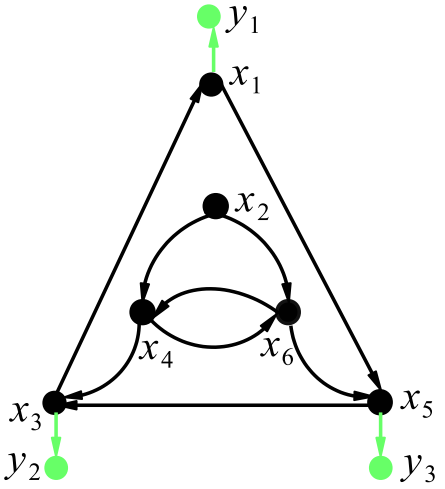
(a) Original graph, i.e., the constraint matrix



(b) For x_1, x_3 and x_5, delete edge to x_2, x_4, and x_6, respectively.



(c) For x_2, delete edge to x_1.



(d) For x_4 and x_6, delete edge to x_2.

Figure 5.2: Process of Algorithm 4, starting with the constraint matrix in (a). This obtains minimum constrained DCS Network Design.

not know the initial state) from q malicious sensors and actuators. To generate $[C]$, we apply graph clustering [78] to the graph associated with $[\bar{A}]$ and group the vertices into five clusters. In each cluster, we assign $q+1$ sensors to q arbitrary state vertices.

Note that the structural system $([\bar{A}], [C])$ constructed based on the previous discussion is not necessarily left invertible and strongly observable with respect to q attacks. In other words, feasible

solutions may not exist for some of the randomly generated pairs $([\bar{A}], [C])$. The following results only consider those $([\bar{A}], [C])$ pairs with a feasible solution. Table 5.1 lists the simulation results, where we consider different values of n, q and r, and record the runtime of Algorithm 4 using a Macbook Pro running Ubuntu Linux with a 2.7 GHz Intel Core i5 processor. In order to compute $q + 1$ essential neighbors of x_i, we incorporate the toolbox TOMLAB/CPLEX.

Chapter 6

Information Flow for Attack Detection

In the previous chapters we have discussed several methods for performing active detection. Specifically, we have introduced physical watermarking, the moving target, and robust structural design. In this chapter, we introduce a formal architecture which can help guide a defender in the analysis and design of systems that are resilient to undetectable attacks. In particular, we introduce the notion of information flows as a means to characterize detectability of attacks. Here, we borrow nomenclature from the study of information flow in software security, which was used to characterize properties of secrecy. Roughly speaking, a designer's goal in such systems was to restrict the flow of information to ensure sensitive information was available to only the appropriate parties. In our work, our goal is to in fact design systems that elicit the flow of information from an attacker's actions to the defender's outputs. We propose the KL divergence as a suitable metric to characterize this information flow and demonstrate its effectiveness in characterizing the stealthiness of an attacker. We conclude by offering a design methodology, which can aid a defender in designing systems that can actively detect an adversary, and investigating the information flow generated by several attack and defense strategies. The results in this chapter are largely based on [79] and [80].

6.1 Active Detection as Causal Information Flow

6.1.1 System Model

We consider a control system with discrete linear time varying model given below.

$$x_{k+1} = A_k x_k + B_k u_k + w_k, \;\; y_k = C_k x_k + v_k. \tag{6.1}$$

As done previously, $x_k \in \mathbb{R}^n$ is the state, $u_k \in \mathbb{R}^p$ is the set of control inputs and $y_k \in \mathbb{R}^m$ is the set of sensor outputs. Once more, $w_k \sim \mathcal{N}(0, Q)$ and $v_k \sim \mathcal{N}(0, R)$ are independent and identically distributed (IID) process and IID measurement noise respectively where $R > 0$. We will additionally consider a version of the system where A_k, B_k, C_k are constant and given by A, B, C respectively. We assume here that $(A, Q^{\frac{1}{2}})$ is stabilizable. We consider a finite horizon T. We will also consider a deterministic setup where

$$x_{k+1} = A_k x_k + B_k u_k, \;\; y_k = C_k x_k. \tag{6.2}$$

We let $\mathcal{I}_k$ be the information available to the defender at time k after making a measurement. In the stochastic setting, the initial state x_0 is unknown. However, the defender knows that $f(x_0|\mathcal{I}_{-1}) = \mathcal{N}(\hat{x}_{0|-1}, P_{0|-1})$. The defender at time -1 is aware of the system model $\mathcal{M} = \{\{A_k, B_k, C_k\}, Q, R, \hat{x}_{0|-1}, P_{0|-1}\}$. In the deterministic setting, if the defender knows x_0, we have $\mathcal{M} = \{\{A_k, B_k, C_k\}, x_0\}$ If the defender does not know x_0, we have $\mathcal{M} = \{\{A_k, B_k, C_k\}\}$. In total the defender's information at time k is given by

$$\mathcal{I}_k = \{y_{0:k}, u_{0:k-1}, \mathcal{M}\}. \tag{6.3}$$

Thus, $\mathcal{I}_{-1} = \mathcal{M}$. Therefore, the defender is a central entity having cumulative knowledge of the dynamics of the system and the history of outputs and inputs. We now aim to define an admissible defender strategy.

Definition 6.1. *An admissible defender strategy is a sequence of deterministic functions* $\{\mathcal{U}_{-1}, \mathcal{U}_0, \cdots, \mathcal{U}_{T-1}\}$ *which take the defender's information and maps it to a feasible action on the system. The set of feasible defense strategies are denoted by* $\mathbb{U}_T = \{\mathcal{U}_{-1:T-1} | \mathcal{U}_{-1:T-1}$ *is feasible* $\}$

We keep the definition of an admissible defender strategy general to account for the various actions the defender can introduce. Physical watermarking at the control input, either through additive Gaussian inputs or intentional packet drops represent a viable defender action. In this case, for $k \geq 0$, $u_k = \mathcal{U}_k(\mathcal{I}_k)$. The action can be represented as a deterministic function since the watermark is in fact a deterministic function of the seed of a pseudo random number generator, which is known to the defender. Other possible actions can include a change in system parameters according a moving target approach. In this case $\mathcal{U}_{-1}(\mathcal{I}_{-1}) = \{A_k, B_k, C_k\}$. Since $\{A_k, B_k, C_k\}$ is chosen exclusively using a pseudorandom number generator, it is a deterministic function of the defender's information. A one time change in the system matrices as dictated in our chapter on robust structural design can also be captured using $\mathcal{U}_{-1}$. For instance, we can have $\mathcal{U}_{-1}(\mathcal{I}_{-1}) = \{A, C\}$. In this chapter, we assume $u_k = \mathcal{U}_k(\mathcal{I}_k)$ for $k \geq 0$. In addition, if the defender has available degrees of freedom, $\mathcal{U}_{-1}(\mathcal{I}_{-1}) = \{A_k, B_k, C_k\}$.

We assume that the defender implements some passive bad data detector to determine whether the system is operating normally, denoted by a null hypothesis $\mathcal{H}_0$, or if there exist an abnormality (or possible attack), denoted by a state of $\mathcal{H}_1$. We define an admissible detector as follows.

Definition 6.2. *An admissible defender detector strategy is a sequence of deterministic functions* $\{\Psi_0, \Psi_1, \cdots, \Psi_T\}$*, which take the defender's information and maps it to a binary decision* $\mathcal{H}_0$ *or* $\mathcal{H}_1$.

Thus, at each time k, the defender intelligently constructs a function Ψ_k which maps the defender's available information to a decision about the state of the system, whether it is operating normally or has faulty and/or malicious behavior. The probability of detection β_k and false alarm α_k are given by

$$\beta_k = \Pr(\Psi_k(\mathcal{I}_k) = \mathcal{H}_1 | \mathcal{H}_1), \quad \alpha_k = \Pr(\Psi_k(\mathcal{I}_k) = \mathcal{H}_1 | \mathcal{H}_0). \tag{6.4}$$

6.1.2 Adversarial Model

We now aim to model an attacker in a CPS. The attacker's actions are a function of the adversarial information. Unlike the defender, the adversary in our setting has potentially two opportunities to act on the system. The attacker can first act given information $\mathcal{I}_k^{a-}$ which potentially includes the control input u_{k-1} sent by the defender, but does not include the true output of the system. The attacker can again act when the true measurement is received. We refer to the attacker's information at this point to be $\mathcal{I}_k^a$. With this, we can define an admissible attack strategy.

Definition 6.3. *An admissible attack strategy is a sequence of deterministic functions* $\{\mathcal{U}_0^a, \mathcal{U}_1^{a-}, \mathcal{U}_1^a, \cdots, \mathcal{U}_T^{a-}, \mathcal{U}_T^a\}$ *which take the attacker's information and maps it to a feasible action on the system.*

The given definition can capture very general attack strategies. For instance, a topology attack time k can be represented as $\mathcal{U}_k^a(\mathcal{I}_k^a) = \{A_k\}$. A denial of service attack can introduce packet drops at the input or output. For instance a drop at the control input can can be characterized as $u_{k-1} = \mathcal{U}_k^{a-}(\mathcal{I}_k^{a-}) = 0$. However, as considered in the rest of this thesis, we focus on integrity attacks. In this case, the system model is given by

$$x_{k+1} = A_k x_k + B_k u_k + B^a u_k^a + w_k, \tag{6.5}$$

$$y_k = C_k x_k + D^a d_k^a + v_k. \tag{6.6}$$

We make the assumption here that an attacker is restricted to manipulate a fixed set of inputs and outputs described by the matrices B^a and D^a. We assume $B^a \in \mathbb{R}^{n \times p_*}$. The sensor attack matrix $D^a \in \mathbb{R}^{m \times m_*}$ is constructed under the assumption that some set m_* sensors can be modified. Note for simplicity we assume B^a and D^a are constant matrices with no nontrivial null space. However, we can also consider the time varying case.

Here, we assume at a minimum that the adversary is aware of his or her own input history. Moreover, the adversary may have the ability to read a subset of control inputs u_k or sensor outputs y_k from the defender. For instance, if the attacker can modify channels, he may also be able to

intercept signals sent along these channels, thereby utilizing a man in the middle attack. The portion of inputs and outputs the attacker and defender can read are public and are denoted u_k^{pu}, y_k^{pu}. Finally, the adversary may have some imperfect prior knowledge of the plant $\hat{\mathcal{M}}$, the controller $\hat{\mathcal{C}}$, and the detector $\hat{\mathcal{D}}$. The adversary's information is

$$\mathcal{I}_k^{a-} = \{u_{0:k-2}^a, d_{0:k-1}^a, u_{0:k-1}^{pu}, y_{0:k-1}^{pu}, \hat{\mathcal{M}}, \hat{\mathcal{C}}, \hat{\mathcal{D}}\}. \tag{6.7}$$

$$\mathcal{I}_k^{a} = \{u_{0:k-1}^a, d_{0:k-1}^a, u_{0:k-1}^{pu}, y_{0:k}^{pu}, \hat{\mathcal{M}}, \hat{\mathcal{C}}, \hat{\mathcal{D}}\}. \tag{6.8}$$

As a result, in this setting $u_{k-1}^a = \mathcal{U}_k^{a-}(\mathcal{I}_k^{a-})$ and $d_k^a = \mathcal{U}_k^a(\mathcal{I}_k^a)$.

6.1.3 Information Flow: Background

In this subsection we introduce the notion of information flows as it relates to software security and propose its use for characterizing detectability in control systems.

Definition 6.4 ([81])**.** *An information flow exists from object x to object y whenever information stored in x is transferred to, or used to derive information transferred to y.*

Information flow has been traditionally used to restrict flows of information to ensure properties of secrecy. For instance, [82] considered a lattice based security structure with a finite number of security classes having a partial ordering. Each object such as a variable, array, or file has a security class. In US intelligence, examples of security classes includes unclassified, confidential, secret, and top secret. Information can flow from unclassified objects to top secret objects but not vice versa. [81] proposed program certifications to ensure only valid information flows existed.

Goguen and Meseguer [83] demonstrated that information flow can be used to express very general security policies including multi-level security, capability passing, confinement, discretionary access control, downgrading, and channel control. These policies can be implemented using noninterference assertions.

Definition 6.5. *An object H is non-interfering with an object L if the behavior of H has no effect on the information of L. Thus, there is no information flow from H to L.*

Significant research has investigated how to design systems with non-interference properties [84], [85], [86]. We however wish to design systems where there does in fact exist interference. Specifically, it has been recently noted that information flow can be related to the problem of detection [87]. For instance, aspects of digital watermarking to detect copyright infringement and traitor tracing to detect stolen keys make use of information flow analysis in order to detect malicious flows of information.

Moreover, the authors propose information flow experiments in order to detect information flows. The specific application considered is web data usage detection where the aim is to determine if the information collected by a user at a website impacts the treatment of the user on that and affiliated websites. During an information flow experiment subjects are randomly assigned to either an experimental group or control group. The differences between each group is carefully controlled. In the case of web data usage detection, the experimental group interacted with websites while the control group remained idle. A hypothesis test is used to determine if the resulting website treatment (as determined by ads) is different.

In the CPS security case, the experimental group is the true system which may or may not be under attack, while the control group is represented by a model. The measurements of a system are compared to the expected outputs to determine if there exists an information flow from a potential adversary to the sensor measurements. A hypothesis test in particular is leveraged to detect an information flow coming from an adversary. If an adversarial strategy does not generate measurements that can be distinguished from the measurements of a system under normal operation, we would say no information flow exists. Such an attack can not be detected. On the other hand, if the attacker's actions generate measurements that can be distinguished from the expected behavior of the system operating normally, then an information flow has been generated. In this case, an attack can be detected.

In stochastic systems, we would like a metric to characterize attack detectability. Quantitative information flow has been studied in the past. Most quantitative measures have been associative [88]. Associative measures of information flow, which quantify correlation, attempt to evaluate

how much information is leaked between two objects and thus provide utility in secrecy and privacy applications. More recently, causal measures for information flow have been explored [89]. Causal measures quantify the extent to which changing an input changes the system output. This is suitable for detection as we aim to characterize how much an attacker's influence perturbs the sensor measurements in our system.

6.1.4 The KL Divergence as a measure of Information Flow

Deterministic Systems

In this section, we propose metrics to measure the information flow introduced by the attacker's inputs. In deterministic systems, we assume w_k and v_k are 0. Detection in deterministic systems is also deterministic. As a result, we propose a binary measure of information flow below.

Definition 6.6. *Assume x_0 is known to the defender. The deterministic information flow IF_T^D from the attacker's inputs $(\mathcal{U}_{0:T}^a, \mathcal{U}_{1:T}^{a-})$ to the defender's outputs $y_{0:T}$ at time T is 0 if for $0 \leq k \leq T$*

$$y_k(x_0, \mathcal{U}_{-1:k-1}, 0, 0) = y_k(x_0, \mathcal{U}_{-1:k-1}, \mathcal{U}_{0:k}^a, \mathcal{U}_{1:k}^{a-}).$$

Otherwise $IF_T^D = \infty$.

By construction, the control inputs are a deterministic function of the prior inputs and outputs, the initial state, and the known sequence of system matrices. As such, the sequence of sensor outputs is entirely deterministic and known to the defender, thus validating our measure of detectability.

Definition 6.7. *Assume x_0 is not known to the defender. The deterministic information flow IF_T^D from the attacker's inputs $(\mathcal{U}_{0:T}^a, \mathcal{U}_{1:T}^{a-})$ to the defender's outputs $y_{0:T}$ at time T is 0 if for $0 \leq k \leq T$ there exists $x_0' \in \mathbb{R}^n$*

$$y_k(x_0', \mathcal{U}_{-1:k-1}, 0, 0) = y_k(x_0, \mathcal{U}_{-1:k-1}, \mathcal{U}_{0:k}^a, \mathcal{U}_{1:k}^{a-}).$$

Otherwise $IF_T^D = \infty$.

By construction, the control inputs are a deterministic function of the prior inputs and outputs and the known sequence of system matrices. As such, for a given initial state, the sequence of sensor outputs is entirely deterministic and known to the defender, thus validating our measure of detectability.

Stochastic Systems

We propose the KL divergence as a measure of information flow in the stochastic setting. We introduced the KL divergence in Chapter 2, but revisit here. For definiteness, we assume that all discrete time stochastic processes of interest considered hereafter induce (joint) distributions on the path space that are absolutely continuous with respect to Lebesgue measure. Thus, they possess densities in the usual sense. The KL divergence between a distribution with probability density function $p(x)$ and a distribution with probability density function $q(x)$ over a sample space X is given by

$$D_{KL}(p(x)||q(x)) = \int_X \log\left(\frac{p(x)}{q(x)}\right) p(x)dx. \tag{6.9}$$

The above definition can be generalized to probability measures. The KL divergence has the following properties

1. $D_{KL}(p(x)||q(x)) \geq 0$.
2. $D_{KL}(p(x)||q(x)) = 0$ if and only if $p(x) = q(x)$ almost everywhere.
3. $D_{KL}(p(x)||q(x)) \neq D_{KL}(q(x)||p(x))$.

We now use the KL divergence to define information flows in a physical system. To begin, denote the conditional distribution of the output based on apriori information as follows.

$$\mathbb{D}_{y_{0:k}}^{\mathcal{I}_{-1},\mathcal{U}_{-1:k-1},\mathcal{U}_{0:k}^{a},\mathcal{U}_{1:k}^{a-}} = f(y_{0:k}|\mathcal{I}_{-1},\mathcal{U}_{-1:k-1},\mathcal{U}_{0:k}^{a},\mathcal{U}_{1:k}^{a-}).$$

Definition 6.8. *The information flow from the attacker's inputs* $(\mathcal{U}^a_{0:T}, \mathcal{U}^{a-}_{1:T})$ *to the defender's outputs* $y_{0:T}$ *at time* T *is*

$$IF_T = \frac{1}{T+1} D_{KL}(\mathbb{D}^{\mathcal{I}_{-1},\mathcal{U}_{-1:T-1},\mathcal{U}^a_{0:T},\mathcal{U}^{a-}_{1:T}}_{y_{0:T}} || \mathbb{D}^{\mathcal{I}_{-1},\mathcal{U}_{-1:T-1},0,0}_{y_{0:T}}).$$

The proposed definition of information flows, both in the deterministic and stochastic settings, has many desirable properties, which make it compatible with existing measures of information flow in cyber security. First, our measures allows us to recover the property of noninterference in deterministic systems and probabilististic noninterference [90] in stochastic systems. There exists interference from a high level object H to a low level object L if changing the behavior of H changes the information of L.

In our model, the low level inputs are the defender's actions, the high level inputs are the attacker's actions, and the low level outputs are the defender's outputs $y_{0:k}$. In a deterministic system, with known initial state, if an adversary's actions change the output $y_{0:k}$, the information flow is infinite, reflecting the fact that there is interference. However, if the output $y_{0:k}$ is the same when the system is operating normally and under attack, indicating noninterference, the deterministic information flow is 0. There exists probabilistic interference from a high level user to a low level user if changing high level inputs measurably alters the distribution of low level outputs. We see $IF_T = 0$ if and only if there exists probabilistic noninterference.

While for simplicity, we consider a fixed window T for detection, we can consider the information flow generated at each T as a means to characterize time to detection. For instance, in a deterministic setting, if the deterministic information flow is 0 from time 0 to $k-1$ but is infinite at time k, then the time to detection is k.

We can use prior work to relate our measure of information flow to properties of detectability. This brings us to the following theorem from [91] and [92].

Theorem 6.1. *Let* $\epsilon > 0$. *Suppose* $\limsup_{k\to\infty} IF_k > \epsilon$. *Then there exists* $0 < \delta < 1$ *and a*

detector $\{\Psi_k\}$ *such that* $\beta_k \geq \delta$ *for all* $k \geq 0$ *and*

$$\limsup_{k\to\infty} -\frac{1}{k+1}\log(\alpha_k) > \epsilon.$$

In addition suppose the outputs generated under attack are ergodic. Suppose $\lim_{k\to\infty} IF_k \leq \epsilon$. *For any detector* $\{\Psi_k\}$ *that satisfies* $\beta_k \geq \delta$ *for all* $k \geq 0$, *where* $0 < \delta < 1$, *we have*

$$\limsup_{k\to\infty} -\frac{1}{k+1}\log(\alpha_k) \leq \epsilon.$$

Finally $IF_T = 0$ *for all* $T \geq 0$ *if and only if there is no* $k \geq 0$ *and detector satisfying* $\beta_k > \alpha_k$.

The information flow is essentially equivalent to the optimal decay rate in the probability of false alarm. As a result, information flow allows us to generically evaluate and compare the detectability of different attack policies as a function of the defender's active detection strategy. However unlike other potential measures such as β_k, the KL divergence can, in many case, be efficiently characterized.

We note that it may be difficult to compute the KL divergence of the outputs $y_{0:T}$ directly. For instance, if a control policy includes nonlinear feedback, the Gaussian property of the output is destroyed, which likely removes the ability to obtain closed form distributions of the output. We can instead consider the normalized residue z_k, obtained from a Kalman filter. The Kalman filter is given by:

$$\begin{aligned}
\hat{x}_{k+1|k} &= A_k\hat{x}_{k|k} + B_k u_k,\ \hat{x}_{k|k} = (I - K_kC_k)\hat{x}_{k|k-1} + K_k y_k,\\
P_{k+1|k} &= A_kP_{k|k-1}A_k^T + Q - A_kP_{k|k-1}C_k^T(C_kP_{k|k-1}C_k^T + R)^{-1}C_kP_{k|k-1}A_k^T,\\
K_k &= P_{k|k-1}C_k^T(C_kP_{k|k-1}C_k^T + R)^{-1},\ \ z_k = (C_kP_{k|k-1}C_k^T + R)^{-\frac{1}{2}}(y_k - C_k\hat{x}_{k|k-1}).
\end{aligned}$$

Recall that the normalized residue z_k is a normalized measure of the difference between the defender's outputs and the expected outputs derived from the state estimate. We now have the following result.

Lemma 6.1. *Consider the stochastic setting. The set of residues* $f(z_{0:k}|\mathcal{I}_{-1},\mathcal{U}_{-1:k-1}) = \mathcal{N}(0,I)$ *when the system is operating normally. Moreover given* $\mathcal{I}_{-1}$ *and an admissible defense strategy* $\mathcal{U}_{-1:k-1}$, $z_{0:k}$ *is an invertible function of* $y_{0:k}$.

Proof. We know $f(z_{0:k}|\mathcal{I}_{-1},\mathcal{U}_{-1:k-1}) = \mathcal{N}(0,I)$ from [93]. We use the fact the time varying system matrices are known to the defender due to $\mathcal{I}_{-1}$ and the fact the residue is independent of the control input.

We now prove that $z_{0:k}$ is an invertible function of $y_{0:k}$ for an admissible control strategy by induction on k. We can trivially obtain $z_{0:k}$ from $y_{0:k}$ using a Kalman Filter so we focus on obtaining $y_{0:k}$ from the residues $z_{0:k}$.

1) Case $k=0$: $y_0 = C_0\hat{x}_{0|-1} + (C_0P_{0|-1}C_0^T + R)^{\frac{1}{2}}z_0$ and the result holds.

2) Case $k=j$: We assume $z_{0:j}$ is an invertible function of $y_{0:j}$.

3) Case $k=j+1$, we observe that

$$\hat{x}_{j+1|j} = A_j\hat{x}_{j|j-1} + A_jK_j(C_jP_{j|j-1}C_j^T + R)^{\frac{1}{2}}z_j + B_ju_j.$$

First, $u_l = \mathcal{U}_l(\mathcal{I}_l) = \mathcal{U}_l(y_{0:l}, u_{0:l-1}, \mathcal{M})$ for $l \leq j$. Given $z_{0:l}$, by our induction assumption, we can obtain $y_{0:l}$ and consequently compute $u_{0:l}$ for $l \leq j$. As such, given $z_{0:j}$ we can compute $\hat{x}_{j+1|j}$. We then see that $y_{j+1} = (C_{j+1}P_{j+1|j}C_{j+1}^T + R)^{\frac{1}{2}}z_{j+1} + C_{j+1}\hat{x}_{j+1|j}$ which concludes the proof. □

Because the residues and outputs are related by an invertible mapping, we can show their KL divergences are equal [94].

Theorem 6.2. *The KL divergence between sensor outputs and between residues are equivalent.*

$$D_{KL}(\mathbb{D}_{y_{0:T}}^{\mathcal{I}_{-1},\mathcal{U}_{-1:T-1},\mathcal{U}_{0:T}^a,\mathcal{U}_{1:T}^{a-}} || \mathbb{D}_{y_{0:T}}^{\mathcal{I}_{-1},\mathcal{U}_{-1:T-1},0,0}) = D_{KL}(\mathbb{D}_{z_{0:T}}^{\mathcal{I}_{-1},\mathcal{U}_{-1:T-1},\mathcal{U}_{0:T}^a,\mathcal{U}_{1:T}^{a-}} || \mathbb{D}_{z_{0:T}}^{\mathcal{I}_{-1},\mathcal{U}_{-1:T-1},0,0})$$

Due to Theorem 6.2, we can analyze the residues operating normally and under attack instead of the system output when computing the information flow. Residues under normal operation have a known zero-mean Gaussian distribution. If the distribution of the residue under attack remains

Gaussian, a closed form solution exists for the KL divergence. The KL divergence between two Gaussian distributions $\mathcal{N}_1 = \mathcal{N}_1(\mu_1, \Sigma_1)$ and $\mathcal{N}_0 = \mathcal{N}_0(\mu_0, \Sigma_0)$ with $\mu_1 \in \mathbb{R}^l$ is [95]

$$D_{KL}(\mathcal{N}_1||\mathcal{N}_0) = -\frac{l}{2} + \frac{1}{2}\text{tr}(\Sigma_0^{-1}\Sigma_1) + \frac{1}{2}\log\det\left(\Sigma_0\Sigma_1^{-1}\right) + \frac{1}{2}(\mu_1 - \mu_0)^T\Sigma_0^{-1}(\mu_1 - \mu_0).$$

Even if the attacker's policy preserves the Gaussianity of the residues, it may still be difficult to compute the KL divergence of $z_{0:k}$ since it is a growing sequence. Fortunately, we can leverage the independence of the residues to obtain the following bound.

Theorem 6.3. *The information flow generated by an adversary can be lower bounded by the sum of the residue-based KL divergences generated at each time step.*

$$IF_T \geq \sum_{k=0}^{T} \frac{D_{KL}(\mathbb{D}_{z_k}^{\mathcal{I}_{-1},\mathcal{U}_{-1:k-1},\mathcal{U}_{0:k}^a,\mathcal{U}_{1:k}^{a-}}||\mathbb{D}_{z_k}^{\mathcal{I}_{-1},\mathcal{U}_{-1:k-1},0,0})}{T+1}.$$

Proof. By Theorem 6.2 and Bayes rule we know

$$IF_T = \sum_{k=0}^{T} \frac{D_{KL}(\mathbb{D}_{z_k|z_{0:k-1}}^{\mathcal{I}_{-1},\mathcal{U}_{-1:k-1},\mathcal{U}_{0:k}^a,\mathcal{U}_{1:k}^{a-}}||\mathbb{D}_{z_k}^{\mathcal{I}_{-1},\mathcal{U}_{-1:k-1},0,0})}{T+1}.$$

Thus, we observe

$$IF_T - IF_T^{LB} = \sum_{k=0}^{T} \frac{I_{z_k,z_{0:k-1}}^{\mathcal{I}_{-1},\mathcal{U}_{-1:k-1},\mathcal{U}_{0:k}^a,\mathcal{U}_{1:k}^{a-}}}{T+1}.$$

where IF_T^{LB} is the obtained lower bound and $I_{z_k,z_{0:k-1}}$ is the mutual information between z_k and $z_{0:k-1}$ which is nonnegative. □

Instead of computing the KL divergence of vectors $z_{0:k} \in R^{mk}$, which in general requires us to store and compute the determinant of a matrix in $\mathbb{R}^{mk\times mk}$, we can instead obtain a recursive lower bound by computing the sum of T divergences for vectors $z_k \in \mathbb{R}^m$. Moreover, note that the gap between the lower bound and IF_T is the scaled sum of mutual informations between z_k and $z_{0:k-1}$ so that if attack residues are independent, the gap is 0.

6.1.5 A Methodology for Design

In this section, we provide loose guidelines for the design of resilient CPS in order to detect adversaries actively. To do this, we first introduce the following definitions.

Definition 6.9. *An attack strategy $\{\mathcal{U}^a_{0:T}, \mathcal{U}^{a-}_{1:T}\}$ generates an unconditional $\epsilon-$weak information flow at time T if for all feasible defense strategies $\mathcal{U}_{-1:T-1}$ in $\mathbb{U}_T$, we have $IF_T \leq \epsilon$.*

Definition 6.10. *Let $\mathbf{U}_T \subset \mathbb{U}_T$. An attack strategy $\{\mathcal{U}^a_{0:T}, \mathcal{U}^{a-}_{1:T}\}$ generates an $\mathbf{U}_T$ conditional $\epsilon-$weak information flow at time T if for all feasible defense strategies $\mathcal{U}_{-1:T-1}$ in $\mathbf{U}_T$, we have $IF_T \leq \epsilon$.*

Definition 6.11. *A defense strategy $\mathcal{U}_{-1:T-1}$ generates an $\mathbf{U}^a_T$ conditional $\epsilon-$strong information flow at time T if for all feasible attack strategies $\{\mathcal{U}^a_{0:T}, \mathcal{U}^{a-}_{1:T}\}$ in $\mathbf{U}^a_T$, we have $IF_T > \epsilon$.*

Note in deterministic systems, we will consider the metric IF^D_T instead of IF_T in the above definitions. Consider a defender who has degrees of freedom represented by $\mathbb{U}_T$ and a current defense policy given by $\mathcal{U}_{-1:T}$. Suppose the defender wishes to detect an attack $\{\mathcal{U}^a_{0:T}, \mathcal{U}^{a-}_{1:T}\}$ with $IF_T > \epsilon$. We first require the defender categorize the information flow. If the defense strategy $\mathcal{U}_{-1:T-1}$ generates a $\{\mathcal{U}^a_{0:T}, \mathcal{U}^{a-}_{1:T}\}$ conditional $\epsilon-$strong information flow at time T, then adequate detection performance is obtained and no further actions need to be taken.

On the other hand, suppose $\{\mathcal{U}^a_{0:T}, \mathcal{U}^{a-}_{1:T}\}$ generates an $\mathbf{U}_T$ conditional $\epsilon-$weak information flow at time T where $\mathcal{U}_{-1:T-1} \subset \mathbf{U}_T$, but $\mathbf{U}_T$ is a strict subset of $\mathbb{U}_T$. Then, we must try to search for defense policies (active detection strategies) in $\mathbb{U}_T$ which allow us to generate a $\{\mathcal{U}^a_{0:T}, \mathcal{U}^{a-}_{1:T}\}$ conditional $\epsilon-$strong information flow. Such a policy, in addition to security, must balance performance and cost constraints. Moreover, ideally we can find such a defense policy (active detection strategy) that retains our ability to detect other realistic attack vectors. For instance, if we desire that $IF_T > \epsilon$ for all feasible attack vectors in $\mathbf{U}^a_T$, we need $\mathcal{U}_{-1:T-1}$ to generate an $\mathbf{U}^a_T$ conditional $\epsilon-$strong information flow.

Finally, suppose $\{\mathcal{U}^{a}_{0:T}, \mathcal{U}^{a-}_{1:T}\}$ generates an unconditional $\epsilon-$weak information flow at time T. In this case, regardless of the chosen active detection policy, we are unable to obtain adequate levels of detection. In this case, it might be necessary to increase the defender's degrees of freedom (i.e. expand $\mathbb{U}_T$) in order to achieve adequate detection of a given attack vector. The design methodology is illustrated in Fig. 6.1.

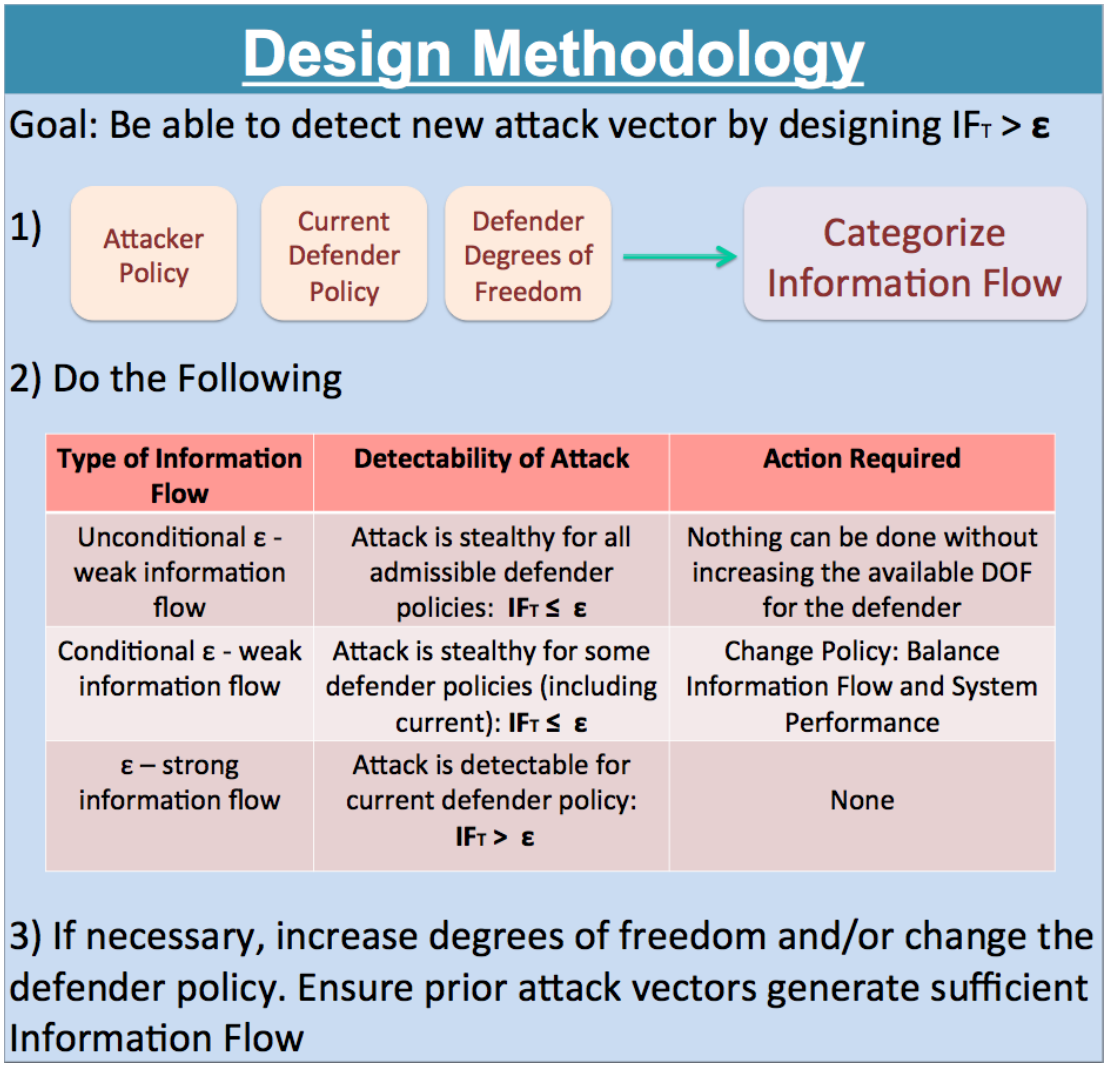


Figure 6.1: A simple design methodology for introducing adequate information flows via active detection

6.1.6 Examples

In this section, we consider several simple examples which illustrate the utility of our information flow metric for detection.

False Data Injection Attacks

Assume that the adversary injects additive inputs which are independent of the defender's system outputs. Thus, we assume

$$\begin{aligned} u^a_{k-1} &= \mathcal{U}^{a-}_k(u^a_{0:k-2}, d^a_{0:k-1}, \hat{\mathcal{M}}, \hat{\mathcal{C}}, \hat{\mathcal{D}}), \\ d^a_k &= \mathcal{U}^a_k(u^a_{0:k-1}, d^a_{0:k-1}, \hat{\mathcal{M}}, \hat{\mathcal{C}}, \hat{\mathcal{D}}). \end{aligned} \tag{6.10}$$

Such attacks are known as false data injection attacks. We now have the following result, allowing us to compute the information flow generated by false data injection attacks.

Theorem 6.4. *Consider an admissible adversarial policy which satisfies* (6.10)*. Then,*

$$IF_T = \frac{1}{2(T+1)} \Delta z^T_{0:T} \Delta z_{0:T}, \tag{6.11}$$

where Δz_k *satisfies* $\Delta e_{-1} = 0$ *and*

$$\begin{aligned} \Delta e_{k+1} &= (A_k - K_{k+1}C_{k+1}A_k)\Delta e_k + (I - K_{k+1}C_{k+1})B^a u^a_k - K_{k+1}D^a d^a_{k+1} \\ \Delta z_k &= (C_k P_{k|k-1} C^T_k + R)^{-\frac{1}{2}} \left(C_k A_{k-1} \Delta e_{k-1} + C_k B^a u^a_{k-1} + D^a d^a_k \right). \end{aligned} \tag{6.12}$$

Proof. Let z_k be the normalized residue under attack and z^s_k be the normalized residue under normal operation. Similar to the proof of Theorem 5.8, we know that

$$z_k = z^s_k + \Delta z_k.$$

Moreover, from an inductive argument, we see that Δz_k is a deterministic variable since $\mathcal{U}^a_{0:k}$ and $\mathcal{U}^{a-}_{1:k}$ are known functions of deterministic variables. As a result, $\mathbb{D}^{\mathcal{I}_{-1}, \mathcal{U}_{-1:k-1}, \mathcal{U}^a_{0:k}, \mathcal{U}^{a-}_{1:k}}_{z_{0:k}} = \mathcal{N}(\Delta z_{0:k}, I)$. Finally, from Lemma 6.1, we know $\mathbb{D}^{\mathcal{I}_{-1}, \mathcal{U}_{-1:k-1}, 0, 0}_{z_{0:k}} = \mathcal{N}(0, I)$. The result follows by computing the KL divergence between normal distributions. □

We remark that Δe_k is the bias introduced in the estimation error of our Kalman filter due to the false data injection attack.

There exists scenarios where a false data injection attack can be incredibly effective. Suppose A_k, B_k, C_k are constant matrices and the Kalman filter has converged to a fixed gain K. Moreover, define $\mathbb{U}_T$ so that all defense strategies can only alter the control input. We have the following result from [43].

Theorem 6.5. *Given some $\underline{B} > 0$, there exists a sensor attack satisfying $\|\Delta z_k\|_2 \leq \underline{B}$ for all $k \geq 0$ while $\limsup_{k\to\infty} \|\Delta e_k\|_2 = \infty$ if and only if A has an unstable eigenvalue with corresponding eigenvector v satisfying*

1. $Cv = \mathit{Image}(D^a)$
2. *v is a reachable state of $\Delta e_k = (A - KCA)\Delta e_{k-1} - KD^a d_k^a$.*

This result can be framed in the context of information flows.

Corollary 6.1. *There exists a destabilizing sensor false data injection which can generate an unconditional $\epsilon-$weak information flow for all time $T \geq 0$ if A has an unstable eigenvalue with corresponding eigenvector v satisfying*

1. $Cv = \mathit{Image}(D^a)$
2. *v is a reachable state of $\Delta e_k = (A - KCA)\Delta e_{k-1} - KD^a d_k^a$.*

This result is easily seen by taking $\underline{B} = \sqrt{2\epsilon}$. We remark that such an attack generates an unconditional $\epsilon-$weak information flow since $\mathbb{U}_T$ only allows a defender to provide active detection through the control input. By increasing the feasible design space, such an attack may only generate a $\mathcal{U}_{-1:T-1}$ conditional $\epsilon-$weak information flow.

Intelligent active detection can allow us to elicit a strong information flow. For example, we can utilize the hybrid moving target proposed in Chapter 4. In this case $\mathcal{U}_{-1}(\mathcal{I}_{-1}) = \{A_k, C_{k+1}\}$. Recall

from Theorem 4.9 that if a defender uses a moving target defense leveraging the design recommendations listed in Chapter 4.2, then $\limsup_{k\to\infty} \|\Delta e_k\|_2 = \infty \implies \limsup_{k\to\infty} \|\Delta z_k\|_2 = \infty$ with probability 1. In other words, a destabilizing attacks results in an unbounded residue. We remark that $\limsup_{k\to\infty} \|\Delta z_k\|_2 = \infty$ on its own does not allow us to make any definitive statements about the information flow as currently defined. However, if we redefine the time that $k = 0$, then we can obtain unbounded information flow. In particular, we have the following.

Corollary 6.2. *Define $\{\mathcal{U}_k\}$ so that the defender uses a moving target defense leveraging the design recommendations listed in Chapter 4.2. Suppose, using a sensor only attack, $\limsup_{k\to\infty} \|\Delta e_k\|_2 = \infty$. Then, with probability 1, for any $\epsilon > 0$, there exists $k' \geq k \geq 0$ which satisfy*

$$\frac{1}{k'-k+1} D_{KL}(\mathbb{D}_{y_{k:k'}}^{\mathcal{I}_{-1},\mathcal{U}_{-1:k'-1},\mathcal{U}_{0:k'}^{a},\mathcal{U}_{1:k'}^{a-}} || \mathbb{D}_{y_{k,k'}}^{\mathcal{I}_{-1},\mathcal{U}_{-1:k'-1},0,0}) > \epsilon.$$

Another effective technique for active detection was the robust design of systems in Chapter 5. For example, suppose we design our system (A, C) so that $(A, [B^a \ 0_{n\times m_*}], C, [0_{m\times p_*} \ D^a])$ is left invertible and strongly observable for all feasible B^a and D^a. As a result, $\mathcal{U}_{-1}(\mathcal{I}_{-1}) = \{A, C\}$. From Theorem 5.4 and Corollary 5.3, if $\limsup_{k\to\infty} \|\Delta e_k\|_2 = \infty$, then $\limsup_{k\to\infty} \|\Delta z_k\|_2 = \infty$. Again, if we redefine the time that $k = 0$, we can obtain a strong information flow.

Corollary 6.3. *Define $\{\mathcal{U}_k\}$ so that the defender designs (A, C) so $(A, [B^a \ 0_{n\times m_*}], C, [0_{m\times p_*} \ D^a])$ is left invertible and strongly observable. Suppose, $\limsup_{k\to\infty} \|\Delta e_k\|_2 = \infty$. Then, for any $\epsilon > 0$, there exists $k' \geq k \geq 0$ which satisfy*

$$\frac{1}{k'-k+1} D_{KL}(\mathbb{D}_{y_{k:k'}}^{\mathcal{I}_{-1},\mathcal{U}_{-1:k'-1},\mathcal{U}_{0:k'}^{a},\mathcal{U}_{1:k'}^{a-}} || \mathbb{D}_{y_{k,k'}}^{\mathcal{I}_{-1},\mathcal{U}_{-1:k'-1},0,0}) > \epsilon.$$

Perfect Attacks and Zero Dynamics Attacks

Consider an LTI control system under attack defined by system matrices (A, B, C) and attack matrices B^a and D^a. Here, we assume B^a and D^a have full column rank. Suppose $(A, [B^a \ 0_{n\times m_*}], C, [0_{m\times p_*} \ D^a])$ is not left invertible. Moreover, assume the defender's strategy

$\mathcal{U}_{-1:k-1}$ only changes the control input. From Theorem 5.2 and Theorem 5.6, we know we can construct a destabilizing perfect attack. In the deterministic setting we know that for all $k \geq 0$,

$$y_k(x_0, \mathcal{U}_{-1:k-1}, 0, 0) = y_k(x_0, \mathcal{U}_{-1:k-1}, \mathcal{U}^a_{0:k}, \mathcal{U}^{a-}_{1:k}).$$

As a result $IF^D_T = 0$ for all T. Such an attacker fails to generate an information flow. Perfect attacks in a stochastic setting are equally effective. Since, a perfect attack introduces no net bias into the sensor measurements, we know that

$$f(y_{0:k}|\mathcal{I}_{-1}, \mathcal{U}_{-1:k-1}, \mathcal{U}^a_{0:k}, \mathcal{U}^{a-}_{1:k}) = f(y_{0:k}|\mathcal{I}_{-1}, \mathcal{U}_{-1:k-1}, 0, 0)$$

As such, $IF_T = 0$ for all k in a perfect attack.

Suppose the defender does not know x_0 in a deterministic setting. Suppose (A, C) is observable but $(A, [B^a \ 0_{n\times m_*}], C, [0_{m\times p_*} \ D^a])$ is not strongly observable. Moreover, assume the defender's strategy $\mathcal{U}_{-1:k-1}$ only changes the control input. From Theorem 5.4, we know we can construct a zero dynamics attack. Here, there exists $x'_0 \in \mathbb{R}^n$ such that for all $k \geq 0$

$$y_k(x'_0, \mathcal{U}_{-1:k-1}, 0, 0) = y_k(x_0, \mathcal{U}_{-1:k-1}, \mathcal{U}^a_{0:k}, \mathcal{U}^{a-}_{1:k}).$$

As a result $IF^D_T = 0$ for all T. Here, in the absence of initial state information, the adversary avoids generating an information flow. We can compute the information flow generated by a zero dynamics attack in the stochastic setting. Here, we assume the Kalman gain K_k and error covariance $P_{k|k-1}$ have converged to K and P respectively. We have the following result.

Theorem 6.6. *Suppose an attacker performs a zero dynamics attacks with the sequence of attack inputs determined by (5.13). Thus, the zero dynamics attack is associated with initial state deviation δx_0. The information flow IF_T generated by a zero dynamics attack is given by*

$$IF_T = \frac{1}{2(T+1)} \sum_{k=0}^{T} \delta x_0^T ((A - AKC)^k)^T C^T (CPC^T + R)^{-1} C(A - AKC)^k \delta x_0. \tag{6.13}$$

Moreover, $\lim_{T\to\infty} IF_T = 0$.

Proof. We observe (6.13) is an immediate consequence of Theorem 5.8 and Theorem 6.4. Next, let $\Sigma = \lim_{T\to\infty} \sum_{k=0}^{T}((A-AKC)^k)^T C^T (CPC^T+R)^{-1} C(A-AKC)^k$. Since $(A-AKC)$ is Schur stable, Σ is the finite solution to the following Lyapunov equation

$$\Sigma = (A-AKC)^T \Sigma (A-AKC) + C^T (CPC^T+R)^{-1} C.$$

As a result, we have $\lim_{T\to\infty} IF_T = \lim_{T\to\infty} \frac{1}{2(T+1)} \delta x_0^T \Sigma \delta x_0 = 0$. □

If the defender only has the degrees of freedom, as defined by $\mathbb{U}_T$, to change the control input, we see that a perfect attack generates an unconditional 0-weak information flow both in the deterministic and stochastic settings. In addition, if the defender does not know the initial state, a zero dynamics attack generates an unconditional 0-weak information flow in the deterministic setting. Finally, for any $\epsilon > 0$, there exists a $T' > 0$ such that a zero dynamics attack generates an unconditional ϵ-weak information flow at time $T \geq T'$ in the stochastic setting.

Increasing the degrees of freedom $\mathbb{U}_T$ can allow a defender to actively detect these integrity attacks and elicit a strong information flow.

Theorem 6.7. *Suppose the defender designs (A, C) so that $(A, [B^a \; 0_{n\times m_*}], C, [0_{m\times p_*} D^a])$ is left invertible. Thus, $\mathcal{U}_{-1}(\mathcal{I}_{-1}) = (A, C)$. Assume x_0 is known to the defender. Without loss of generality assume that either d_0^a or u_0^a is nonzero. Then for $T \geq n$, $IF_T^D = \infty$.*

Proof. We prove by contradiction. Suppose for some $T \geq n$, we have

$$y_k(x_0, \mathcal{U}_{-1:k-1}, 0, 0) = y_k(x_0, \mathcal{U}_{-1:k-1}, \mathcal{U}_{0:k}^a, \mathcal{U}_{1:k}^{a-}),$$

for $0 \leq k \leq T$. From Theorem 5.1, this implies

$$\delta x_{k+1} = A\delta x_k + B^a u_k^a, \quad 0 \leq k \leq T-1, \qquad \delta x_0 = 0, \tag{6.14}$$

$$0 = \delta y_k = C\delta x_k + D^a d_k^a, \quad 0 \leq k \leq T. \tag{6.15}$$

From, Corollary 1 of [96], since the system is left invertible, it is n delay invertible. Thus, we can uniquely recover input u_0^a, d_0^a given $\delta y_0, \cdots \delta y_n$. when $\delta x_0 = 0$. We see that $\delta y_k = 0$ for $0 \leq k \leq n$. As a result, this implies both $u_0^a = 0$ and $d_0^a = 0$, which is a contradiction. □

Let $\epsilon > 0$. In the deterministic setting, for $T \geq n$, we know that $\mathcal{U}_{-1:T-1}$ generates an $(\mathcal{U}^a_{0:T}, \mathcal{U}^{a-}_{1:T})$ conditional $\epsilon-$strong information flow at time T if the attacker's strategy dictates d^a_0 or u^a_0 is nonzero. We can obtain similar results by designing strongly observable systems.

Theorem 6.8. *Suppose the defender designs (A, C) so that $(A, [B^a \ 0_{n\times m_*}], C, [0_{m\times p_*} \ D^a])$ is left invertible and strongly observable. Thus, $\mathcal{U}_{-1}(\mathcal{I}_{-1}) = (A, C)$. Assume x_0 is not known to the defender. Without loss of generality assume that either d^a_0 or u^a_0 is nonzero. Then for $T \geq n$, $IF^D_T = \infty$.*

Proof. We prove by contradiction. Suppose for some $T \geq n$, we have

$$y_k(x_0, \mathcal{U}_{-1:k-1}, 0, 0) = y_k(x'_0, \mathcal{U}_{-1:k-1}, \mathcal{U}^a_{0:k}, \mathcal{U}^{a-}_{1:k}),$$

for $0 \leq k \leq T$. From Theorem 5.3, this implies

$$\delta x_{k+1} = A\delta x_k + B^a u^a_k, \quad 0 \leq k \leq T-1, \qquad \delta x_0 \in \mathbb{R}^n, \tag{6.16}$$

$$0 = \delta y_k = C\delta x_k + D^a d^a_k, \quad 0 \leq k \leq T. \tag{6.17}$$

From, Theorem 4 of [97], since the system is strongly observable, we can uniquely recover input δx_0 given $\delta y_0, \cdots \delta y_n$ and unknown inputs. Since $\delta y_0, \cdots \delta y_n$ are all zero, we know $\delta x_0 = 0$. Since the system is left invertible, $\delta y_k = 0$ for $0 \leq k \leq n$, and $\delta x_0 = 0$, we can uniquely recover u^a_0 and d^a_0. This implies both $u^a_0 = 0$ and $d^a_0 = 0$, which is a contradiction. □

Again in the deterministic setting, for $T \geq n$, we know that $\mathcal{U}_{-1:T-1}$ generates an $(\mathcal{U}^a_{0:T}, \mathcal{U}^{a-}_{1:T})$ conditional $\epsilon-$strong information flow at time T if the attacker's strategy dictates d^a_0 or u^a_0 is nonzero.

Replay Attacks

Consider a stochastic LTI control system with system matrices A, B, C. In a replay attack, the adversary observes a sequence of measurements from y_{-N} to y_{-N+T-1}. Then, without loss of generality, at time 0, the attacker replays these measurements. Here, we will assume $N \geq T$ is

large so that the adversary has an adequate buffer and that the replayed outputs are independent of the current outputs. Moreover we assume the system at time $-N$ is in steady state. In the ensuing asymptotic results, we will make the assumption that N and T approach ∞. We first argue that a replay attack generates a conditional ϵ-weak information flow for common control policies $\mathcal{U}_{0:k-1}$. For instance, consider a defender that uses state feedback with gain L so $\mathcal{U}_k(\mathcal{I}_k) = L\hat{x}_{k|k}$. Let $\mathcal{A} \triangleq (A+BL)(I-KC)$ and $\bar{P} \triangleq (CPC^T + R)$. It has been shown that [27]

$$z_k = z_{k-N} - \bar{P}^{-\frac{1}{2}} C\mathcal{A}^k(\hat{x}_{0|-1} - \hat{x}_{-N|-N-1}). \tag{6.18}$$

If $\mathcal{A}$ is Schur stable, the second term converges to 0. Therefore, we have the following result.

Theorem 6.9. *Suppose that our control system (2.1), (2.2) with state feedback control is under replay attack, where $\mathcal{A}$ is Schur stable. Then, $\lim_{T\to\infty} IF_T = 0$.*

Proof. We observe from (6.18) that under replay attack

$$z_{0:k} \sim \mathcal{N}(\mu_r, \Sigma_r), \tag{6.19}$$

$$\mu_r(jm : jm+m-1) = -\bar{P}^{-\frac{1}{2}} C\mathcal{A}^k \hat{x}_{0|-1}, \tag{6.20}$$

$$\Sigma_r(jm : jm+m-1, lm : lm+m-1) = \bar{P}^{-\frac{1}{2}} C\mathcal{A}^j \mathcal{W} (\mathcal{A}^l)^T C^T \bar{P}^{-\frac{1}{2}} + \delta(l-m)I, \tag{6.21}$$

where $\mathcal{W}$ is the steady state covariance of $\hat{x}_{k|k-1}$ and δ refers to the discrete delta dirac function. From Theorem 6.2, and Sylvester's determinant theorem we have

$$D_{KL}(\mathbb{D}_{y_{0:k}}^{\mathcal{I}_{-1},\mathcal{U}_{-1:k-1},\mathcal{U}_{0:k}^a,\mathcal{U}_{1:k}^{a-}} || \mathbb{D}_{y_{0:k}}^{\mathcal{I}_{-1},\mathcal{U}_{-1:k-1},0,0}) = \frac{c_1 + c_2 + c_3}{2}$$

where

$$c_1 = \text{tr}\left(\sum_{j=0}^{k} \bar{P}^{-\frac{1}{2}} C\mathcal{A}^j \mathcal{W} (\mathcal{A}^j)^T C^T \bar{P}^{-\frac{1}{2}} \right),$$

$$c_2 = \sum_{j=0}^{k} \hat{x}_{0|-1}^T (\mathcal{A}^j)^T C^T \bar{P}^{-1} C\mathcal{A}^j \hat{x}_{0|-1},$$

$$c_3 = -\log\det\left(I + \sum_{j=0}^{k} \mathcal{W}^{\frac{1}{2}} (\mathcal{A}^j)^T C^T \bar{P}^{-1} C\mathcal{A}^j \mathcal{W}^{\frac{1}{2}} \right).$$

Let X_1 and X_2 be given by

$$X_1 = \sum_{j=0}^{\infty} \mathcal{A}^j \mathcal{W} (\mathcal{A}^j)^T = \mathcal{A} X_1 \mathcal{A}^T + \mathcal{W},$$

$$X_2 = \sum_{j=0}^{\infty} (\mathcal{A}^j)^T C^T \bar{P}^{-1} C \mathcal{A}^j = \mathcal{A}^T X_2 \mathcal{A} + C^T \bar{P}^{-1} C.$$

From Lyapunov's equation and since $\mathcal{A}$ is stable, the matrices X_1 and X_2 exist and are bounded. Since c_1, c_2, and $|c_3|$ are monotonic in k, we have for all k

$$c_1 \leq \text{tr}\left(\bar{P}^{-\frac{1}{2}} C X_1 C^T \bar{P}^{-\frac{1}{2}}\right),\ c_2 \leq \hat{x}_{0|-1}^T X_2 \hat{x}_{0|-1}^T,\ |c_3| \leq \log\det\left(I + \mathcal{W}^{\frac{1}{2}} X_2 \mathcal{W}^{\frac{1}{2}}\right).$$

Consequently, for all k there exists M^* satisfying

$$D_{KL}(\mathbb{D}_{y_{0:k}}^{\mathcal{I}_{-1}, \mathcal{U}_{-1:k-1}, \mathcal{U}_{0:k}^a, \mathcal{U}_{1:k}^{a-}} || \mathbb{D}_{y_{0:k}}^{\mathcal{I}_{-1}, \mathcal{U}_{-1:k-1}, 0, 0}) \leq M^*,$$

Dividing by $k+1$, the result follows. □

If $\mathcal{A}$ is stable, the adversary's actions are asymptotically undetectable since the information flow is 0. This result agrees closely with Theorem 2.1, but is obtained independently using our new measure of information flow.

In this example, the defender's control strategy $\mathcal{U}_{-1:T-1}$ of state feedback, leaves the system vulnerable to a replay attack. We have observed that watermarking can be effective against replay attacks. It is our argument that introducing a physical watermark can allow us to obtain a strong information flow. We assume $u_k = \mathcal{U}_k(\mathcal{I}_k) = L\hat{x}_{k|k} + \Delta u_k$ where $\Delta u_k \sim \mathcal{N}(0, \mathcal{J})$ is an IID watermark. Note that while the watermark is random, it can be predetermined offline so that $\mathcal{U}_k(\mathcal{I}_k)$ remains a deterministic function. We now show watermarking creates a strong information flow.

Theorem 6.10. *Suppose the system (2.1), (2.2) with state feedback control and watermarking is under replay attack, where $\rho(\mathcal{A}) < 1$. Then, almost surely $\lim_{T\to\infty} IF_T \geq \epsilon$, where*

$$\epsilon = \frac{\text{tr}\left(\bar{P}^{-1} C \Sigma C^T\right)}{2}, \quad \Sigma = \mathcal{A}\Sigma\mathcal{A}^T + B\mathcal{J}B^T.$$

Proof. When under a replay attack, we have [27]

$$z_k = z_{k-N} - \bar{P}^{-\frac{1}{2}} C\mathcal{A}^k(\hat{x}_{0|-1} - \hat{x}_{-N|-N-1}) - \bar{P}^{-\frac{1}{2}} C \sum_{j=0}^{k-1} \mathcal{A}^{k-1-j} B\left(\Delta u_j - \Delta u_{j-N}\right),$$

where N is some unknown, but large delay between the replayed sequence and the true sequence. Thus, under attack $z_k \sim \mathcal{N}(\mu_k, \Sigma_k + I)$ with

$$\mu_k = \bar{P}^{-\frac{1}{2}} C\mathcal{A}^k \hat{x}_{0|-1} + \bar{P}^{-\frac{1}{2}} C \sum_{j=0}^{k-1} \mathcal{A}^{k-1-j} B\Delta u_j,$$

$$\Sigma_k = \bar{P}^{-\frac{1}{2}} C[\mathcal{A}^k \mathcal{W} \mathcal{A}^{k\ T} + \sum_{j=0}^{k-1} \mathcal{A}^j B\mathcal{J}B^T \mathcal{A}^{j\ T}] C^T \bar{P}^{-\frac{1}{2}}.$$

Thus, the KL divergence between z_k under attack and under normal operation is given by

$$D_{KL}(\mathbb{D}_{z_k}^{\mathcal{I}_{-1}, \mathcal{U}_{-1:k-1}, \mathcal{U}_{0:k}^a, \mathcal{U}_{1:k}^{a-}} || \mathbb{D}_{z_k}^{\mathcal{I}_{-1}, \mathcal{U}_{-1:k-1}, 0, 0}) = \frac{c_k^1 + c_k^2 + c_k^3}{2}, \tag{6.22}$$

where

$$c_k^1 = \mu_k^T \mu_k, \quad c_k^2 = -\log\det\left(I + \Sigma_k\right), \quad c_k^3 = \text{tr}(\Sigma_k).$$

From (2.45), it is known that

$$c_k^2 + c_k^3 \geq 0. \tag{6.23}$$

Furthermore, by the law of large numbers, we know

$$\lim_{T\to\infty} \frac{1}{T+1} \sum_{k=0}^{T} c_k^1 \overset{a.s.}{\to} \text{tr}\left(\bar{P}^{-1} C\Sigma C^T\right). \tag{6.24}$$

Using (6.22), (6.23) and (6.24), we know almost surely that

$$\lim_{T\to\infty} \sum_{k=0}^{T} \frac{D_{KL}(\mathbb{D}_{z_k}^{\mathcal{I}^{-1}, \mathcal{U}_{-1:k-1}, \mathcal{U}_{0:k}^a, \mathcal{U}_{1:k}^{a-}} || \mathbb{D}_{z_k}^{\mathcal{I}_{-1}, \mathcal{U}_{-1:k-1}, 0, 0})}{T+1} \geq \epsilon. \tag{6.25}$$

By Theorem 6.3, the result immediately follows. □

From the theorem above, the defender can make the asymptotic information flow from an adversarial input arbitrarily large (generating a strong information flow) by increasing $\text{tr}\left(\bar{P}^{-1} C\Sigma C^T\right)$ which is a linear function of the watermark covariance $\mathcal{J}$.

In fact, previous work on IID watermarking [27] does aim to design watermarks by maximizing $\text{tr}\left(\mathcal{P}^{-1}C\Sigma C^T\right)$ subject to constraints on control performance in the system. Thus, our results motivate the choice of this objective function. The use of information flows also allow us to extend previous results to analyze optimal detection of replay attacks under watermarking scenarios.

Corollary 6.4. *Consider a system with state feedback control and IID Gaussian watermarking under a replay attack, where* $\rho(\mathcal{A}) < 1$. *Let* $\epsilon > 0$. *Then for some* $0 > \delta > 1$ *there exists a detector such that* $\beta_k \geq \delta,\ \forall\ k \geq 0$ *and*

$$\limsup_{k\to\infty} -\frac{1}{k+1}\log(\alpha_k) > \frac{1}{2}\,\text{tr}\left(\mathcal{P}^{-1}C\Sigma C^T\right) - \epsilon. \tag{6.26}$$

Proof. The result follows from Theorems 6.10 and 6.1. □

6.1.7 Conclusions: Information Flow

In this chapter, we proposed a measure to characterize the detectability of an attack strategy as a function of the defender's strategy, motivated by the study of causal information flow. We then briefly discussed a design methodology to ensure attacks generate adequate information flow. Finally, we demonstrated how techniques for active detection allow us to increase the information flow generated by a given adversary. This captures fundamentally the idea of active detection. Specifically, active detection enables us to design systems where we can better distinguish between adversarial and normal output sequences. As such, our passive algorithms for detection are more effective. In general the notion of information flow was not necessary to obtain the mathematical results in this section. However, using this terminology provides two main advantages as we consider future work. A language of security in terms of information flow will allow scientists to begin to develop unifying theories that incorporate both control system security and software/cyber security. In addition, recognizing the parallels between the proposed work and other methods that leverage information flow analysis could eventually result in an exchange of tools, techniques, and ideas which can advance the field of CPS security.

Chapter 7

Summary and Conclusions

In this dissertation, we proposed the technique of active detection for the purposes of designing resilient cyber-physical systems that can detect integrity attacks generated by resourceful, intelligent, and powerful attackers. Here, we were motivated by the fact that standard or passive detection can fail to distinguish between normal and adversarial outputs. Consequently, the defender needed to leverage available degrees of freedom to design resilient systems and controllers. Through such intelligent design, we have shown we are able to detect otherwise stealthy attacks. We proposed several mechanisms, which allow us to achieve active detection.

To begin, we considered physical watermarking where the defender introduces a noisy Gaussian signal into the control input. We designed stationary watermarks, which allowed us to detect replay attacks. Moreover, we designed robust watermarks to counter a class of attackers with model knowledge. We then considered environmental watermarks, which occur naturally within a control system and evaluated how the inherent randomness of these phenomena can allow us to detect replay and simulation attacks. In particular, we considered packet drops at the control input. We then considered the design of a Gaussian watermark that is composed with (possibly) intentional packet drops.

Next, motivated by the idea that physical watermarking can fail against classes of model aware adversaries, we proposed the moving target approach. Here, we attempted to introduce time varying

system dynamics in order to limit an attacker's understanding of the dynamic behavior of a system and thus limit his/her ability to construct stealthy attacks. An authenticating subsystem approach, where an additional time varying system is introduced to the plant, was considered. Here, we obtained bounds characterizing the effectiveness of an adversary that attacks the CPS. In addition, we evaluated a hybrid system approach where a system switched among multiple modes. In this case, we offered design recommendations which enable a defender to detect and identify sensor attacks in a control system.

As an alternative to the moving target, we considered the robust offline design of systems to limit the presence of stealthy and harmful attacks. Here, we characterized the effectiveness of both perfect attacks and zero dynamics attacks. We demonstrated that designing systems that are left invertible and strongly observable for all feasible attack strategies enable us to detect all stealthy attacks in deterministic systems, eliminate destabilizing stealthy attacks in stochastic systems, and possibly perform attack identification and resilient estimation. We used structural system theory to arrive at graphical conditions which allow us to design left invertible and strongly observable systems and then solved optimization problems which allowed us to achieve minimal robust design in distributed control systems.

We concluded by providing brief investigations into the concept of information flow. We argued that information flow can be used to characterize attack detectability in CPS, proposed a measure of information flow, and suggested a fundamental design methodology based on this measure.

Significant tasks remain in our goal to achieve resilient cyber-physical systems. This thesis in large part assumes model knowledge was available to the defender. However, this may not be the case. It is our argument that active techniques for detection remain effective even when the model is uncertain or unknown. Here, it is only important to understand how the system responds to the perturbations a defender introduces and to ensure attacks can not mimic these perturbations. Passive detection, must however be tuned to recognize the deviations between normal and malicious outputs in the absence of precise model knowledge.

In addition, this dissertation largely focused on the task of detection. We note that attack

detection is a critical first step towards responding to an adversary. If an attacker can remain stealthy for long periods of time, he or she can maximize their impact and cause significant damage without having to worry about defender interference. However, once an attack is detected, the defender must respond in order to assure the graceful degradation of cyber-physical systems.

A first task upon detection is to identify the malicious nodes in a CPS. Identifying malicious nodes allows a defender to design attack specific countermeasures that can allow a system to recover. For instance, upon identifying malicious sensors, a defender can construct resilient estimators that bypass misleading output nodes. Additional techniques for active identification can be explored in future work. Upon attack identification, it is also imperative to design actions that lead to system recovery. Ideally, to save time, such actions should be obtained automatically upon identification. Of course, in general the number of possible failure modes can be extremely large in a CPS and as such developing countermeasures in all scenarios may be intractable. Thus, an important research problem is to consider how to incorporate risk when designing mechanisms for response and recovery.

This dissertation explored the process of securing cyber-physical systems from a largely system theoretic viewpoint. However, a complete study of secure cyber-physical systems must be able to simultaneously consider both system theoretic security and cyber security. An important problem to consider is compositional security. We need to understand how properties of security are preserved or compromised when we combine traditional control systems with communication networks, software systems, and cryptographic primitives.

Finally, our current treatment does not specialize in challenges which may arise in specific applications such as transportation systems, the smart grid, health care, and water distribution. For the results we developed in this dissertation to have a direct impact on society, one has to investigate how the presented tools can be applied to real world scenarios. While it is unavoidable to reach different conclusions based on more specific practical models, the mathematical frameworks in this dissertation offer methods to formally handle many problems that emerge when securing cyber-physical systems.

Bibliography

[1] A. A. Cardenas, T. Roosta, and S. Sastry, "Rethinking security properties, threat models, and the design space in sensor networks: A case study in SCADA systems," *Ad Hoc Networks*, vol. 7, no. 8, pp. 1434–1447, 2009. 2

[2] L. Xie, Y. Mo, and B. Sinopoli, "False data injection attacks in electricity markets," in *International Conference on Smart Grid Communications*. IEEE, 2010, pp. 226–231. 2

[3] B. DeBruhl, S. Weerakkody, B. Sinopoli, and P. Tague, "Is your commute driving you crazy?: A study of misbehavior in vehicular platoons," in *Conference on Security & Privacy in Wireless and Mobile Networks*. ACM, 2015, pp. 22:1–22:11. 2

[4] D. P. Fidler, "Was stuxnet an act of war? decoding a cyberattack," *IEEE Security & Privacy*, vol. 9, no. 4, pp. 56–59, 2011. 3

[5] T. Chen, "Stuxnet, the real start of cyber warfare?[editor's note]," *IEEE Network*, vol. 24, no. 6, pp. 2–3, 2010. 3

[6] J. Slay and M. Miller, "Lessons learned from the Maroochy water breach," in *International Conference on Critical Infrastructure Protection*. Springer, 2007, pp. 73–82. 3, 42

[7] T. Pultarova, "Cyber security-Ukraine grid hack is wake-up call for network operators [news briefing]," *Engineering & Technology*, vol. 11, no. 1, pp. 12–13, 2016. 3

[8] "Analysis of the cyber attack on the ukrainian power grid," *Electricity Information Sharing and Analysis Center (E-ISAC)*, 2016. 3

[9] L. Schenato, B. Sinopoli, M. Franceschetti, K. Poolla, and S. S. Sastry, "Foundations of control and estimation over lossy networks," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 163–187, 2007. 5, 62, 63, 64, 65, 89

[10] B. Sinopoli, L. Schenato, M. Franceschetti, K. Poolla, M. I. Jordan, and S. S. Sastry, "Kalman filtering with intermittent observations," *IEEE Transactions on Automatic Control*, vol. 49, no. 9, pp. 1453–1464, 2004. 5, 62

[11] Y. Mo, S. Weerakkody, and B. Sinopoli, "Physical authentication of control systems: Designing watermarked control inputs to detect counterfeit sensor outputs," *IEEE Control Systems*, vol. 35, no. 1, pp. 93–109, 2015. 11

[12] S. Weerakkody, Y. Mo, and B. Sinopoli, "Detecting integrity attacks on control systems using robust physical watermarking," in *53rd IEEE Conference on Decision and Control*. IEEE, 2014, pp. 3757–3764. 11

[13] R. M. Needham and M. D. Schroeder, "Using encryption for authentication in large networks of computers," *Communications of the ACM*, vol. 21, no. 12, pp. 993–999, 1978. 12

[14] P. Kumar and P. Varaiya, *Stochastic Systems: Estimation, Identification, and Adaptive Control*. Prentice Hall, 1986. 13

[15] Y. Mo and B. Sinopoli, "Secure control against replay attacks," in *47th Allerton Conference on Communication, Control, and Computing*. IEEE, 2009, pp. 911–918. 14, 22, 24, 35, 42, 43, 76, 90

[16] R. K. Mehra and J. Peschon, "An innovations approach to fault detection and diagnosis in dynamic systems," *Automatica*, vol. 7, no. 5, pp. 637–640, 1971. 17

[17] P. E. Greenwood and M. S. Nikulin, *A guide to chi-squared testing*. John Wiley & Sons, 1996. 17

[18] R. Langner, "To kill a centrifuge: A technical analysis of what stuxnet's creators tried to achieve," Langner Communications, Tech. Rep., November 2013. [Online]. Available: www.langner.com/en/wp-content/uploads/2013/11/To-kill-a-centrifuge.pdf 18

[19] A. Teixeira, D. Pérez, H. Sandberg, and K. H. Johansson, "Attack models and scenarios for networked control systems," in *Proceedings of the 1st international conference on High Confidence Networked Systems*. ACM, 2012, pp. 55–64. 19, 79

[20] A. Teixeira, I. Shames, H. Sandberg, and K. H. Johansson, "A secure control framework for resource-limited adversaries," *Automatica*, vol. 51, pp. 135–148, 2015. 19

[21] R. S. Smith, "Covert misappropriation of networked control systems: Presenting a feedback structure," *IEEE Control Systems*, vol. 35, no. 1, pp. 82–92, 2015. 19, 99

[22] L. L. Scharf and C. Demeure, *Statistical Signal Processing: Detection, Estimation And Time Series Analysis*. Addison-Wesley Pub. Co., 1991. 28

[23] S. Kullback and R. Leiber, "On information and sufficiency," *The Annals of Mathematical Statistics*, vol. 22, no. 1, 1951. 28

[24] S. Eguchi and J. Copas, "Interpreting Kullback–Leibler divergence with the Neyman–Pearson lemma," *Journal of Multivariate Analysis*, vol. 97, no. 9, 2006. 29

[25] T. Chonavel and J. Ormrod, *Statistical Signal Processing: Modelling and Estimation*, ser. Advanced Textbooks in Control and Signal Processing. Springer Verlag GmbH, 2002. 32, 252

[26] P. Delsarte, Y. Genin, and Y. Kamp, "Orthogonal polynomial matrices on the unit circle," *IEEE Transactions on Circuits and Systems*, vol. 25, no. 3, pp. 149–160, 1978. 32, 252

[27] Y. Mo, R. Chabukswar, and B. Sinopoli, "Detecting integrity attacks on SCADA systems," *IEEE Transactions on Control Systems Technology*, vol. 22, no. 4, pp. 1396–1407, 2014. 35, 43, 50, 55, 58, 80, 90, 228, 230, 231, 254

[28] M. Grant and S. Boyd, "CVX: Matlab software for disciplined convex programming, version 2.0 beta," www.cvxr.com/cvx, Sep. 2013. 35

[29] ——, "Graph implementations for nonsmooth convex programs," in *Recent Advances in Learning and Control*, ser. Lecture Notes in Control and Information Sciences, V. Blondel, S. Boyd, and H. Kimura, Eds. Springer-Verlag Limited, 2008, vol. 371, pp. 95–110. 35

[30] R. Chabukswar, Y. Mo, and B. Sinopoli, "Detecting integrity attacks on SCADA systems," in *18th IFAC World Congress*, Milan, Italy, 2011, pp. 11 239–11 244. 43, 67, 70, 71

[31] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Journal of Fluids Engineering*, vol. 82, no. 1, pp. 35–45, 1960. 51

[32] C. DeSouza, M. Gevers, and G. Goodwin, "Riccati equations in optimal filtering of nonstabilizable systems having singular state transition matrices," *IEEE Transactions on Automatic Control*, vol. 31, no. 9, pp. 831–838, 1986. 51

[33] L. Scharf, *Statistical Signal Processing*. Prentice Hall, 1990. 54, 138

[34] P. Lancaster and L. Rodman, *Algebraic Riccati Equations*. Oxford University Press, 1995. 56

[35] O. Ozel, S. Weerakkody, and B. Sinopoli, "Physical watermarking for securing cyber physical systems via packet drop injections," in *2017 IEEE International Conference on Smart Grid Communications*. IEEE, 2017, pp. 271–276. 61

[36] S. Weerakkody, O. Ozel, and B. Sinopoli, "A Bernoulli-Gaussian physical watermark for detecting integrity attacks in control systems," in *55th Annual Allerton Conference on Communication Control and Computing*. IEEE, 2017, pp. 966–973. 61

[37] V. Gungor, B. Lu, and G. Hancke, "Opportunities and challenges of wireless sensor networks in smart grid," *IEEE Transactions on Industrial Electronics*, vol. 57, no. 10, pp. 3557 – 3564, October 2010. 62

[38] L. Zheng, N. Lu, and L. Cai, "Reliable wireless communication networks for demand response control," *IEEE Transactions on Smart Grid*, vol. 4, no. 1, pp. 133 – 140, March 2013. 62

[39] K. H. Johansson, "The quadruple-tank process: A multivariable laboratory process with an adjustable zero," *IEEE Transactions on Control Systems Technology*, vol. 8, no. 3, pp. 456–465, 2000. 67, 115

[40] M. Grebeck, "A comparison of controllers for the quadruple tank system," *Department of Automatic Control, Lund Institute of Technology, Lund, Sweden, Tech. Rep*, 1998. 67, 115

[41] A. R. Bergen, *Power systems analysis*. Pearson Education India, 2009. 70

[42] Y. Mo, E. Garone, and B. Sinopoli, "LQG control with Markovian packet loss," in *European Control Conference*. IEEE, 2013, pp. 2380–2385. 75, 83, 84, 88, 245, 247, 249

[43] Y. Mo and B. Sinopoli, "False data injection attacks in cyber physical systems," in *First Workshop on Secure Control Systems*, Stockholm, Sweden, 2010. 79, 223

[44] F. Pasqualetti, F. Dörfler, and F. Bullo, "Attack detection and identification in cyber-physical systems," *IEEE Transactions on Automatic Control*, vol. 58, no. 11, pp. 2715–2729, 2013. 79, 119, 161

[45] F. Miao, M. Pajic, and G. J. Pappas, "Stochastic game approach for replay attack detection," in *52nd IEEE Conference on Decision and Control*. IEEE, 2013, pp. 1854–1859. 90

[46] S. Weerakkody and B. Sinopoli, "Detecting integrity attacks on control systems using a moving target approach," in *54th IEEE Conference on Decision and Control*. IEEE, 2015, pp. 5820–5826. 99

[47] ——, "A moving target approach for identifying malicious sensors in control systems," in *54th Annual Allerton Conference on Communication Control and Computing*. IEEE, 2016, pp. 1149–1156. 99

[48] L. Zuo, R. Niu, and P. K. Varshney, "Conditional posterior Cramer - Rao lower bounds for nonlinear sequential Bayesian estimation," *IEEE Transactions on Signal Processing*, vol. 59, no. 1, pp. 1–14, 2011. 110, 111, 112, 113

[49] H. L. Van Trees, *Detection Estimation and Modulation Theory*. New York: Wiley, 1968, vol. 1. 110

[50] L. Zuo, *Conditional posterior Cramer-Rao lower bound and distributed target tracking in sensor networks*. Syracuse University, 2011. 111

[51] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," *IEEE Transactions on Signal Processing*, vol. 50, no. 2, pp. 174–188, 2002. 113

[52] Y. Zheng, O. Ozdemir, R. Niu, and P. K. Varshney, "New conditional posterior Cramer - Rao lower bounds for nonlinear sequential Bayesian estimation," *IEEE Transactions on Signal Processing*, vol. 60, no. 10, pp. 5549–5556, 2012. 113

[53] P. Tichavsky, C. H. Muravchik, and A. Nehorai, "Posterior Cramer - Rao bounds for discrete-time nonlinear filtering," *IEEE Transactions on Signal Processing*, vol. 48, no. 2, pp. 1386–1395, 1998. 113

[54] H. Fawzi, P. Tabuada, and S. Diggavi, "Secure estimation and control for cyber-physical systems under adversarial attacks," *IEEE Transactions on Automatic Control*, vol. 59, no. 6, pp. 1454–1467, 2014. 120

[55] Y. Nakahira and Y. Mo, "Dynamic state estimation in the presence of compromised sensory data," in *54th IEEE Conference on Decision and Control*. IEEE, 2015, pp. 5808–5813. 123

[56] P. D. Lax, *Linear Algebra and its Applications*. NJ: Wiley-Interscience, 2007. 130

[57] R. A. Horn and C. R. Johnson, *Matrix analysis*. Cambridge university press, 1990. 130

[58] W. F. Mascarenhas, "The structure of the eigenvectors of sparse matrices," *Linear algebra and its applications*, vol. 207, pp. 1–20, 1994. 132

[59] J. R. Gilbert, "Predicting structure in sparse matrix computations," *SIAM Journal on Matrix Analysis and Applications*, vol. 15, no. 1, pp. 62–79, 1994. 132

[60] S.-L. Sun and Z.-L. Deng, "Multi-sensor optimal information fusion Kalman filter," *Automatica*, vol. 40, no. 6, pp. 1017–1023, 2004. 136

[61] Q. Gan and C. J. Harris, "Comparison of two measurement fusion methods for Kalman-filter-based multisensor data fusion," *IEEE Transactions on Aerospace and Electronic systems*, vol. 37, no. 1, pp. 273–279, 2001. 136

[62] S. Weerakkody, X. Liu, S. H. Son, and B. Sinopoli, "A graph theoretic characterization of perfect attackability and detection in distributed control systems," in *2016 American Control Conference*. IEEE, 2016, pp. 1171–1178. 145

[63] ——, "A graph-theoretic characterization of perfect attackability for secure design of distributed control systems." *IEEE Transactions on Control of Network Systems*, vol. 4, no. 1, pp. 60–70, 2017. 145

[64] S. Weerakkody, X. Liu, and B. Sinopoli, "Robust structural analysis and design of distributed control systems to prevent zero dynamics attacks," in *56th IEEE Conference on Decision and Control*. IEEE, 2017, pp. 1356–1361. 145

[65] H. L. Trentelman, A. A. Stoorvogel, and M. Hautus, *Control theory for linear systems*. Springer Science & Business Media, 2012. 148, 150, 151, 152, 181

[66] Y. Mo and B. Sinopoli, "Integrity attacks on cyber-physical systems," in *Proceedings of the 1st international conference on High Confidence Networked Systems*. ACM, 2012, pp. 47–54. 159

[67] J. van der Woude, "The generic number of invariant zeros of a structured linear system," *SIAM Journal on Control and Optimization*, vol. 38, no. 1, pp. 1–21, 1999. 166, 169

[68] ——, "A graph-theoretic characterization for the rank of the transfer matrix of a structured system," *Mathematics of Control, Signals and Systems*, vol. 4, no. 1, pp. 33–40, 1991. 166, 169

[69] C. Commault, J.-M. Dion, and J. W. van der Woude, "Characterization of generic properties of linear structured systems for efficient computations," *Kybernetika*, vol. 38, no. 5, pp. 503–520, 2002. 168

[70] K. Menger, "Zur allgemeinen kurventheorie," *Fundamenta Mathematicae*, vol. 10, no. 1, pp. 96–115, 1927. 169

[71] J.-M. Dion, C. Commault, and J. van der Woude, "Generic properties and control of linear structured systems: a survey," *Automatica*, vol. 39, no. 7, pp. 1125–1144, 2003. 169

[72] T. Boukhobza, F. Hamelin, and S. Martinez-Martinez, "State and input observability for structured linear systems: A graph-theoretic approach," *Automatica*, vol. 43, no. 7, pp. 1204–1210, 2007. 173

[73] T. Boukhobza and F. Hamelin, "State and input observability recovering by additional sensor implementation: A graph-theoretic approach," *Automatica*, vol. 45, no. 7, pp. 1737–1742, 2009. 173

[74] E. A. Dinic, "Algorithm for solution of a problem of maximum flow in networks with power estimation," in *Soviet Math. Doklady*, vol. 11, 1970, pp. 1277–1280. 181, 182

[75] S. Even and R. E. Tarjan, "Network flow and testing graph connectivity," *SIAM journal on computing*, vol. 4, no. 4, pp. 507–518, 1975. 181, 182

[76] X. Liu, S. Pequito, S. Kar, B. Sinopoli, and A. P. Aguiar, "Minimum sensor placement for robust observability of structured complex networks," 2015. [Online]. Available: http://arxiv.org/pdf/1507.07205v1.pdf 203, 204

[77] T. H. Cormen, C. Stein, R. L. Rivest, and C. E. Leiserson, *Introduction to Algorthms*, 2nd ed. McGraw-Hill Higher Education, 2001. 203

[78] S. E. Schaeffer, "Survey: Graph clustering," *Computer Science Review*, vol. 1, no. 1, pp. 27–64, 2007. 206

[79] S. Weerakkody, B. Sinopoli, S. Kar, and A. Datta, "Information flow for security in control systems," in *55th IEEE Conference on Decision and Control*. IEEE, 2016, pp. 5065–5072. 208

[80] A. Datta, S. Kar, B. Sinopoli, and S. Weerakkody, "Accountability in cyber-physical systems," in *Science of Security for Cyber-Physical Systems Workshop*. IEEE, 2016, pp. 1–3. 208

[81] D. E. Denning and P. J. Denning, "Certification of programs for secure information flow," *Communications of the ACM*, vol. 20, no. 7, pp. 504–513, 1977. 212

[82] D. E. Denning, "A lattice model of secure information flow," *Communications of the ACM*, vol. 19, no. 5, pp. 236–243, 1976. 212

[83] J. A. Goguen and J. Meseguer, "Security policies and security models," in *IEEE Symposium on Security and Privacy*. IEEE, 1982, pp. 11–20. 212

[84] D. Volpano, C. Irvine, and G. Smith, "A sound type system for secure flow analysis," *Journal of computer security*, vol. 4, no. 2-3, pp. 167–187, 1996. 213

[85] G. Barthe, P. D'Argenio, and T. Rezk, "Secure information flow by self-composition," in *17th IEEE Computer Security Foundations Workshop*. IEEE, 2004, pp. 100–114. 213

[86] V. N. Venkatakrishnan, W. Xu, D. C. DuVarney, and R. Sekar, "Provably correct runtime enforcement of non-interference properties," in *International Conference on Information and Communications Security*. Springer, 2006, pp. 332–351. 213

[87] M. C. Tschantz, A. Datta, A. Datta, and J. M. Wing, "A methodology for information flow experiments," *arXiv preprint arXiv:1405.2376*, 2014. 213

[88] G. Smith, "On the foundations of quantitative information flow," in *International Conference on Foundations of Software Science and Computational Structures*. Springer, 2009, pp. 288–302. 213

[89] A. Datta, S. Sen, and Y. Zick, "Algorithmic transparency via quantitative input influence: Theory and experiments with learning systems," in *IEEE Symposium on Security and Privacy*. IEEE, 2016, pp. 598–617. 214

[90] D. Volpano and G. Smith, "Probabilistic noninterference in a concurrent language," *Journal of Computer Security*, vol. 7, no. 2-3, pp. 231–253, 1999. 216

[91] C.-Z. Bai, F. Pasqualetti, and V. Gupta, "Security in stochastic control systems: Fundamental limitations and performance bounds," in *American Control Conference (ACC), 2015*. IEEE, 2015, pp. 195–200. 216

[92] ——, "Data-injection attacks in stochastic control systems: Detectability and performance tradeoffs," *Automatica*, vol. 82, pp. 251–260, 2017. 216

[93] R. K. Mehra and J. Peschon, "An innovations approach to fault detection and diagnosis in dynamic systems," *Automatica (Journal of IFAC)*, vol. 7, no. 5, pp. 637–640, 1971. 218

[94] S. Kullback, *Information theory and statistics*. Courier Corporation, 1997. 218

[95] T. M. Cover and J. A. Thomas, *Elements of information theory*. John Wiley & Sons, 2012. 219

[96] A. Willsky, "On the invertibility of linear systems," *IEEE Transactions on Automatic Control*, vol. 19, no. 3, pp. 272–274, 1974. 226

[97] S. Sundaram and C. N. Hadjicostis, "Distributed function calculation via linear iterations in the presence of malicious agents - part II: Overcoming malicious behavior," in *2008 American Control Conference,*. IEEE, 2008, pp. 1356–1361. 227

Appendix A

Proof of Theorem 2.6

Proof. We first define function $\tilde{\Gamma} : \mathbb{Z} \to \mathbb{R}^{p\times p}$ as

$$\tilde{\Gamma}(d) \triangleq \bar{\rho}^{-|d|}\Gamma(d). \tag{A.1}$$

From the constraints of the optimization problem, we observe $\tilde{\Gamma}$ is an autocovariance function of a stationary Gaussian process. The proof is divided into steps.

Step 1 Rewrite the objective function and the contraint of Problem (2.52) in terms of the Fourier transform $\tilde{\nu}$ of $\tilde{\Gamma}$.

Consider a partition of $[0, 1/2]$ into disjoint intervals $\mathcal{I}_1, \mathcal{I}_2, \ldots, \mathcal{I}_q$, where

$$\mathcal{I}_i \bigcap \mathcal{I}_j = \emptyset, \bigcup_{i=1}^{q} \mathcal{I}_i = [0, \frac{1}{2}].$$

Define σ as the maximum length of interval $\mathcal{I}_i$s. By Riemann-Stieltjes integral and Theorem 2.5, $\tilde{\Gamma}(d)$ can be written as

$$\tilde{\Gamma}(d) = \lim_{\sigma\to 0} 2\,\mathfrak{Re}\left[\sum_{i=1}^{q} \exp(2\pi j d\omega_i)\tilde{\nu}(\mathcal{I}_i)\right],$$

where $\omega_i \in \mathcal{I}_i$. By (2.39) and (A.1),

$$\begin{aligned}\Sigma &= \lim_{\sigma\to 0} C \sum_{i=1}^{q} 2\,\mathfrak{Re}\left\{2\sum_{d=0}^{\infty}\mathrm{sym}\left[\exp(2\pi j d\omega_i)\,(\bar{\rho}\mathcal{A})^d\,\mathcal{L}_2(\tilde{\nu}(\mathcal{I}_i))\right] - \mathcal{L}_2(\tilde{\nu}(\mathcal{I}_i))\right\} C^T,\\ &= \lim_{\sigma\to 0} C \sum_{i=1}^{q} 2\,\mathfrak{Re}\left\{2\,\mathrm{sym}\left[(I-\exp(2\pi j\omega_i)\bar{\rho}\mathcal{A})^{-1}\mathcal{L}_2(\tilde{\nu}(\mathcal{I}_i))\right] - \mathcal{L}_2(\tilde{\nu}(\mathcal{I}_i))\right\} C^T,\\ &= \lim_{\sigma\to 0} C \sum_{i=1}^{q} \mathcal{F}_2(\omega_i, \tilde{\nu}(I_i)) C^T.\end{aligned}$$

Notice that the order of summation and limit changes, which is feasible as $\mathcal{A}$ is stable. As a result,

$$\mathrm{tr}(\Sigma\bar{P}^{-1}) = \lim_{\sigma\to 0}\sum_{i=1}^{q}\mathrm{tr}\left[\mathcal{F}_2(\omega_i,\tilde{\nu}(I_i))C^T\bar{P}^{-1}C\right]. \tag{A.2}$$

Similarly,

$$\Delta J = \lim_{\sigma\to 0}\sum_{i=1}^{q}\mathcal{F}_1(\omega_i,\tilde{\nu}(I_i)). \tag{A.3}$$

Step 2 Prove that the upper bound for Problem (2.52) is the optimal value of the objective function of Problem (2.57).

Since ΔJ and Σ are always nonnegative, for all $\omega \in [0, 1/2]$ and H positive semidefinite,

$$\mathcal{F}_1(\omega, H) \geq 0,\ \mathcal{F}_2(\omega, H) \geq 0. \tag{A.4}$$

Suppose that the optimal solution of (2.57) is $\omega_*,\ H_*$ and the optimal value of the objective function is φ. Since $\mathcal{F}_1$ and $\mathcal{F}_2$ are linear with respect to H, it can be shown that

$$\mathcal{F}_1(\omega_*, H_*) = \delta.$$

Hence, for all $\tilde{\nu}(I_i)$ and $\omega_i \in [0, 1/2]$

$$\mathrm{tr}\left[\mathcal{F}_2(\omega_i,\tilde{\nu}(I_i))C^T\bar{P}^{-1}C\right] \leq \frac{\varphi}{\delta}\mathcal{F}_1(\omega_i,\tilde{\nu}(I_i)). \tag{A.5}$$

By (A.2)-(A.5), for all watermark signals $\{\Delta u_k\}$ with $\Delta J \leq \delta$,

$$\mathrm{tr}(\Sigma\bar{P}^{-1}) \leq \varphi.$$

Step 3 Prove that the upper bound is tight.

Consider the point mass measure $\tilde{\nu}_*$,

$$\tilde{\nu}_*(S_B) = H_* \mathbb{I}_{\{\omega_* \in S_B\}} + \overline{H_*} \mathbb{I}_{\{-\omega_* \in S_B\}},$$

where $\mathbb{I}$ is the indicator function. It can be shown that $\Gamma_*(d)$ is generated by $\tilde{\nu}_*$. Furthermore, by (A.2) and (A.3), the corresponding $\Delta J = \delta$ and $\mathrm{tr}(\Sigma \bar{P}^{-1}) = \varphi$. Hence, $\Gamma_*(d)$ achieves the upper bound of Problem (2.52). Now it only remains to prove that $\Gamma_*(d)$ can be generated by an HMM with $\rho(A_\omega) \leq \bar{\rho}$.

Notice that the boundary of the cone of positive semidefinite Hermitian matrices is of the form hh^H. Furthermore, since $\mathcal{F}_1$ and $\mathcal{F}_2$ are linear with respect to H, for fixed ω, the optimization problem (2.57) attains its maximum on the boundary of the cone (through it is possible that an interior point is also optimal), which proves (2.60). As a result,

$$H_* = (h_r + jh_i)(h_r^T - jh_i^T) = h_r h_r^T + h_i h_i^T - j(h_r h_i^T - h_i h_r^T).$$

It can be shown that the watermark signal $\{\Delta u_k\}$ generated by the HMM (2.61) follows (2.56), which proves that (2.56) is the optimal autocovariance function for Problem (2.52).

□

Appendix B

Proof of Lemma 3.2

Proof. We begin with the following Lemma.

Lemma B.1. *Assume $\{\eta_k\}$ is a stationary Markovian drop process. Suppose $\alpha > 0$ and $\beta > 0$ are chosen so that the system has finite cost $J_{(m)}$ [42][Theorem 3] in the absence of a Gaussian watermark. Consider $\bar{x}_{k+1} = (A + \eta_k BL_{(m)})\bar{x}_k$ and $\bar{x}'_{k+1} = (A + \eta_k BL_{(m)})\bar{x}'_k$,*

$$\bar{x}_0 = \begin{cases} x_0^0 & \eta_{-1} = 0 \\ x_0^1 & \eta_{-1} = 1 \end{cases}, \quad \bar{x}'_0 = \begin{cases} x_0^{0'} & \eta_{-1} = 0 \\ x_0^{1'} & \eta_{-1} = 1 \end{cases}. \tag{B.1}$$

Then, we have

$$\lim_{k\to\infty} \mathbb{E}[\bar{x}_k^i \bar{x}_k^{j\prime} | \eta_{k-1} = 0] = 0, \quad \lim_{k\to\infty} \mathbb{E}[\bar{x}_k^i \bar{x}_k^{j\prime} | \eta_{k-1} = 1] = 0,$$
$$\lim_{k\to\infty} \mathbb{E}[\bar{x}_k^i \bar{x}_k^{j\prime}] = 0, \quad \lim_{k\to\infty} \mathbb{E}[(\bar{x}_k^i)^2] = 0,$$

where $i, j \in \{1, \cdots, n\}$ and $\bar{x}_k^i$ is the ith element of $\bar{x}_k$ and $\bar{x}_k^{j\prime}$ is the jth element of $\bar{x}'_k$.

Proof. Define $\mathcal{J}_N \triangleq \sum_{k=0}^N \epsilon \mathbb{E}[\bar{x}_k^T \bar{x}_k]$ where $\epsilon > 0$ is chosen so $\epsilon I \leq W$. Moreover the cost to go function is defined as $\bar{V}_j(\bar{x}_j) \triangleq \sum_{k=j}^N \epsilon \mathbb{E}[\bar{x}_k^T \bar{x}_k | \eta_{-1:j-1}]$. We show that

$$\bar{V}_k(\bar{x}_k) = \begin{cases} \mathbb{E}[\bar{x}_k^T \bar{S}_k \bar{x}_k | \eta_{-1:k-1}] & (\eta_{k-1} = 0) \\ \mathbb{E}[\bar{x}_k^T \bar{R}_k \bar{x}_k | \eta_{-1:k-1}] & (\eta_{k-1} = 1) \end{cases}, \tag{B.2}$$

where

$$\bar{S}_k = g_1(\bar{S}_{k+1}, \bar{R}_{k+1}), \bar{R}_k = g_2(\bar{S}_{k+1}, \bar{R}_{k+1}),$$
$$g_1(X,Y) \triangleq \epsilon I + (1-\alpha)A^T X A + \alpha F^T Y F, \;\; g_2(X,Y) \triangleq \epsilon I + \beta A^T X A + (1-\beta)F^T Y F,$$

and $F = A + BL_{(m)}$, $\bar{S}_N = \epsilon I$, $\bar{R}_N = \epsilon I$. The proof is by induction. (B.2) holds for $k = N$. Assume it holds for $k = t+1$. We next show (B.2) holds for $k = t$. Conditioned on $\eta_{t-1} = 0$, we have

$$\begin{aligned}
\bar{V}_t(\bar{x}_t) &= \mathbb{E}[\epsilon \bar{x}_t^T \bar{x}_t + \bar{V}_{t+1}(\bar{x}_{t+1}) | \eta_{-1:t-1}], \\
&= \mathbb{E}[\epsilon \bar{x}_t^T \bar{x}_t + \bar{x}_t^T ((1-\alpha)A^T \bar{S}_{t+1} A + \alpha F^T \bar{R}_{t+1} F)\bar{x}_t | \eta_{-1:t-1}], \\
&= \mathbb{E}[\bar{x}_t^T g_1(\bar{S}_{t+1}, \bar{R}_{t+1})\bar{x}_t | \eta_{-1:t-1}] = \mathbb{E}[\bar{x}_t^T \bar{S}_t \bar{x}_t | \eta_{-1:t-1}].
\end{aligned}$$

The case when $\eta_{t-1} = 1$ is similar. Thus,

$$\mathcal{J}_N = \mathbb{E}[\bar{V}_0(\bar{x}_0)] = \frac{\beta x_0^{0T} \bar{S}_0 x_0^0 + \alpha x_0^{1T} \bar{R}_0 x_0^1}{\alpha + \beta}. \tag{B.3}$$

We claim that $\lim_{N\to\infty} \mathcal{J}_N$ exists. Consider the sequence

$$\mathcal{S}_{k+1} = g_1(\mathcal{S}_k, \mathcal{R}_k), \;\; \mathcal{R}_{k+1} = g_2(\mathcal{S}_k, \mathcal{R}_k), \;\; \mathcal{R}_0 = \mathcal{S}_0 = \epsilon I. \tag{B.4}$$

We observe that $g_1(X,Y)$ and $g_2(X,Y)$ are monotonically increasing functions in (X,Y). Because $\mathcal{S}_1 \geq \mathcal{S}_0$ and $\mathcal{R}_1 \geq \mathcal{R}_0$, we see that $\{\mathcal{S}_k\}$ and $\{\mathcal{R}_k\}$ are monotonically increasing in the semidefinite sense. Now consider the sequence

$$\bar{\mathcal{S}}_{k+1} = h_1(\bar{\mathcal{S}}_k, \bar{\mathcal{R}}_k), \bar{\mathcal{R}}_{k+1} = h_2(\bar{\mathcal{S}}_k, \bar{\mathcal{R}}_k), \bar{\mathcal{R}}_0 = \bar{\mathcal{S}}_0 = \epsilon I, \tag{B.5}$$

where we define

$$\begin{aligned}
h_1(X,Y) &\triangleq W + \alpha L_{(m)}^T U L_{(m)} + (1-\alpha)A^T X A + \alpha F^T Y F, \\
h_2(X,Y) &\triangleq W + (1-\beta) L_{(m)}^T U L_{(m)} + \beta A^T X A + (1-\beta) F^T Y F.
\end{aligned}$$

Again, we observe that $h_1(X,Y)$ and $h_2(X,Y)$ are monotonically increasing in (X,Y). Because $\bar{\mathcal{S}}_1 \geq \bar{\mathcal{S}}_0$ and $\bar{\mathcal{R}}_1 \geq \bar{\mathcal{R}}_0$, it can be seen that $\{\bar{\mathcal{S}}_k\}$ and $\{\bar{\mathcal{R}}_k\}$ are monotonically increasing in the

semidefinite sense. Moreover, due to Lemma 4 in [42], $\{\bar{\mathcal{S}}_k\}$ and $\{\bar{\mathcal{R}}_k\}$ converge. We observe that if $X \leq \bar{X}$ and $Y \leq \bar{Y}$, then $g_1(X,Y) \leq h_1(\bar{X},\bar{Y})$ and $g_2(X,Y) \leq h_2(\bar{X},\bar{Y})$. Since $\mathcal{R}_0 = \bar{\mathcal{R}}_0$ and $\mathcal{S}_0 = \bar{\mathcal{S}}_0$, it can be seen that $\mathcal{S}_k \leq \bar{\mathcal{S}}_k$ and $\mathcal{R}_k \leq \bar{\mathcal{R}}_k$ for all k.

As a result, $\{\mathcal{S}_k\}$ and $\{\mathcal{R}_k\}$ are bounded above by a monotonically increasing, convergent sequence, which in turn means that $\{\mathcal{S}_k\}$ and $\{\mathcal{R}_k\}$ are bounded. From the monotone convergence theorem $\{\mathcal{S}_k\}$ and $\{\mathcal{R}_k\}$ converge to some $\mathcal{S}^*$ and $\mathcal{R}^*$. It is immediately seen that

$$\lim_{N\to\infty} \mathcal{J}_N = \frac{\beta x_0^{0T} \mathcal{S}^* x_0^0 + \alpha x_0^{1T} \mathcal{R}^* x_0^1}{\alpha + \beta}. \tag{B.6}$$

Note that $\mathcal{J}_N = \sum_{k=0}^{N} \epsilon \mathbb{E}[\bar{x}_k^T \bar{x}_k]$. Since $\mathcal{J}_N$ converges to a finite constant, $\lim_{k\to\infty} \mathbb{E}[\bar{x}_k^T \bar{x}_k] = 0$. Since $0 \leq \mathbb{E}[(\bar{x}_k^i)^2] \leq \mathbb{E}[\bar{x}_k^T \bar{x}_k]$, we immediately obtain

$$\lim_{k\to\infty} \mathbb{E}[(\bar{x}_k^i)^2] = 0. \tag{B.7}$$

By symmetry this also implies that $\lim_{k\to\infty} \mathbb{E}[(\bar{x}_k^{j'})^2] = 0$. By Cauchy Schwartz, we see

$$0 \leq (\mathbb{E}[\bar{x}_k^i \bar{x}_k^{j'}])^2 \leq \mathbb{E}[(\bar{x}_k^i)^2]\mathbb{E}[(\bar{x}_k^{j'})^2]. \tag{B.8}$$

Since $\mathbb{E}[(\bar{x}_k^i)^2]\mathbb{E}[(\bar{x}_k^{j'})^2]$ converges to 0, we know $(\mathbb{E}[\bar{x}_k^i \bar{x}_k^{j'}])^2$ converges to 0 and thus

$$\lim_{k\to\infty} \mathbb{E}[\bar{x}_k^i \bar{x}_k^{j'}] = 0. \tag{B.9}$$

Next, we observe that

$$0 \leq \frac{\min(\alpha,\beta)}{\alpha+\beta} \mathbb{E}[(\bar{x}_k^i)^2 | \eta_{k-1} = l] \leq \mathbb{E}[(\bar{x}_k^i)^2], \tag{B.10}$$

where $l \in \{0,1\}$. As $\alpha, \beta > 0$ by assumption, we know $\lim_{k\to\infty} \mathbb{E}[(\bar{x}_k^i)^2 | \eta_{k-1} = l] = 0$. Using the Cauchy Schwartz inequality in a similar manner as before, we see

$$\lim_{k\to\infty} \mathbb{E}[\bar{x}_k^i \bar{x}_k^{j'} | \eta_{k-1} = 1] = 0, \quad \lim_{k\to\infty} \mathbb{E}[\bar{x}_k^i \bar{x}_k^{j'} | \eta_{k-1} = 0] = 0. \tag{B.11}$$

□

We are now ready to prove the desired result. To this end, we first observe that

$$\begin{pmatrix} \mathbb{E}[\bar{x}'_{k+1}\bar{x}^T_{k+1}|\eta_k = 0] \\ \mathbb{E}[\bar{x}'_{k+1}\bar{x}^T_{k+1}|\eta_k = 1] \end{pmatrix} = \mathcal{L}_0 \begin{pmatrix} \mathbb{E}[\bar{x}'_k\bar{x}^T_k|\eta_{k-1} = 0] \\ \mathbb{E}[\bar{x}'_k\bar{x}^T_k|\eta_{k-1} = 1] \end{pmatrix}. \tag{B.12}$$

As a result,

$$\begin{pmatrix} \mathbb{E}[\bar{x}'_k\bar{x}^T_k|\eta_{k-1} = 0] \\ \mathbb{E}[\bar{x}'_k\bar{x}^T_k|\eta_{k-1} = 1] \end{pmatrix} = \mathcal{L}_0^k \begin{pmatrix} x_0^{0'} x_0^{0\ T} \\ x_0^{1'} x_0^{1\ T} \end{pmatrix}. \tag{B.13}$$

Leveraging (B.11), we see that $\lim_{k\to\infty} \mathbb{E}[\bar{x}'_k\bar{x}^T_k|\eta_{k-1} = l] = 0$ for $l \in \{0, 1\}$. Consequently, we have

$$\lim_{k\to\infty} \mathcal{L}_0^k \begin{pmatrix} x_0^{0'} x_0^{0\ T} \\ x_0^{1'} x_0^{1\ T} \end{pmatrix} = 0. \tag{B.14}$$

Note that $x_0^{0'}, x_0^0, x_0^{1'}$, and x_0^1 can be chosen so that $\mathcal{L}_0^k$ is applied to an arbitrary canonical basis vector in $\mathbb{R}^{2n\times n}$. Thus, for all $M \in \mathbb{R}^{2n\times n}$, $\lim_{k\to\infty} \mathcal{L}_0^k(M) = 0$. Thus, $\mathcal{L}_0$ is stable. □

Appendix C

Proof of Theorem 3.4

Proof. We begin with the following Lemma.

Lemma C.1. *Suppose p_d is chosen so the system with IID drops has finite cost $J_{(b)}$ [42][Theorem 3]. Then the matrix $(A + \bar{p}_d BL_{(b)})$ is Schur stable. Moreover, the operator $\mathcal{L}_1(X) \triangleq \bar{p}_d(A + BL_{(b)})X(A+BL_{(b)})^T + p_d AXA^T$ is stable. Specifically, $\forall M \in \mathbb{R}^{n\times n}$, we have $\lim_{k\to\infty} \mathcal{L}_1^k(M) = 0$*

Proof. Consider the systems $\bar{x}_{k+1} = (A + \eta_k BL_{(b)})\bar{x}_k$, and $\bar{x}'_{k+1} = (A + \eta_k BL_{(b)})\bar{x}'_k$. where η_k is an IID drop process with drop probability p_d and $\bar{x}_0 = x_{0,*}$, $\bar{x}'_0 = x'_{0,*}$. Observe that

$$\mathbb{E}[\bar{x}_k] = (A + \bar{p}_d BL_{(b)})^k x_{0,*}. \tag{C.1}$$

Noting that the IID drop case is a special instance of Markovian drops, we know from Lemma B.1 that $\lim_{k\to\infty} \mathbb{E}[(\bar{x}_k^i)^2] = 0$. Using the fact that $\mathbb{E}[(\bar{x}_k^i)^2] \geq (\mathbb{E}[\bar{x}_k^i])^2 \geq 0$, we have $\lim_{k\to\infty} \mathbb{E}[\bar{x}_k] = 0$. As a result, for all $x_{0,*} \in \mathbb{R}^n$

$$\lim_{k\to\infty} (A + \bar{p}_d BL_{(b)})^k x_{0,*} = 0. \tag{C.2}$$

Thus, $(A + \bar{p}_d BL_{(b)})$ is Schur stable. Next, we note that

$$\mathbb{E}[\bar{x}'_{k+1}\bar{x}_{k+1}^T] = \mathcal{L}_1(\mathbb{E}[\bar{x}'_k\bar{x}_k^T]). \tag{C.3}$$

As a result,

$$\mathbb{E}[\bar{x}'_k\bar{x}_k^T] = \mathcal{L}_1^k(x'_{0,*}x_{0,*}^T). \tag{C.4}$$

Leveraging (B.9), we note $\lim_{k\to\infty}\mathbb{E}[\bar{x}_k'\bar{x}_k^T]=0$ and this implies

$$\lim_{k\to\infty}\mathcal{L}_1^k(x_{0,*}'x_{0,*}^T)=0. \tag{C.5}$$

Note that $x_{0,*}'$ and $x_{0,*}$ can be chosen so that $\mathcal{L}_1^k$ is applied to an arbitrary canonical basis vector in $\mathbb{R}^{n\times n}$. Thus, for all $M\in\mathbb{R}^{n\times n}$, $\lim_{k\to\infty}\mathcal{L}_1^k(M)=0$. Thus, $\mathcal{L}_1$ is stable. □

We now proceed to the main proof. We obtain an equivalent realization to (3.48) by using autocovariance functions $\Gamma(d)\triangleq\mathbb{E}[\Delta u_k\Delta u_{k+d}]$.

Step 1: Calculate $\bar{J}$ in terms of $\Gamma(d)$:

Let us first compute

$$\mathbb{E}[x_t^TWx_t+u_{t,c}^TUu_{t,c}]=\text{tr}(W\text{Cov}(x_t))+\text{tr}(U\text{Cov}(u_{t,c})),$$

for fixed $t\geq 0$. It can be seen that

$$\begin{aligned}
x_t&=l_{1,\{\eta_k\}}(w_{-\infty:t-1},v_{-\infty:t-1})+\gamma_t(\Delta u_{-\infty:t-1}),\\
u_{t,c}&=l_{2,\{\eta_k\}}(w_{-\infty:t-1},v_{-\infty:t})+\eta_tL_{(b)}\gamma_t(\Delta u_{-\infty:t-1})+\eta_t\Delta u_t,\\
\gamma_t(\Delta u_{-\infty:t-1})&=\sum_{i=1-t}^{\infty}\left[\prod_{j=1-i}^{t-1}(A+\eta_jBL_{(b)})\right]\eta_{-i}B\Delta u_{-i},
\end{aligned} \tag{C.6}$$

where l_1 and l_2 are linear functions of the process and sensor noise for fixed realizations of the drop process η_k. Since $\{w_k\}$ and $\{v_k\}$ are independent of Δu_k, we observe that

$$\bar{J}=J_{(b)}(p_d)+\frac{1}{N}\lim_{N\to\infty}\sum_{t=0}^{N-1}\left(\text{tr}(W\text{Cov}(\gamma_t))+\text{tr}(U\text{Cov}(\eta_t[L_{(b)}\gamma_t+\Delta u_t]))\right). \tag{C.7}$$

Step 1a: Calculate $\text{Cov}(\gamma_t)$:

Define $Z_t\triangleq\sum_{i=1-t}^{\infty}\gamma_{i,t}\gamma_{i,t}^T$ where

$$\gamma_{i,t}\triangleq\left[\prod_{j=1-i}^{t-1}(A+\eta_jBL_{(b)})\right]\eta_{-i}B\Delta u_{-i}.$$

We see that $\mathbb{E}[Z_{t+1}]$ is equal to

$$\mathbb{E}[(A+\eta_tBL_{(b)})Z_t(A+\eta_tBL_{(b)})^T+\eta_t^2B\Delta u_t\Delta u_t^TB^T].$$

Since η_t is independent of Z_t, we have

$$\mathbb{E}[Z_{t+1}] = \mathcal{L}_1(\mathbb{E}[Z_t]) + \bar{p}_d B\Gamma(0)B^T.$$

Since $\mathcal{L}_1$ is stable and the system has been running since $k = -\infty$, $\mathbb{E}[Z_t]$ is the unique solution of the following fixed point equation.

$$E[Z_t] = \mathcal{L}_1(\mathbb{E}[Z_t]) + \bar{p}_d B\Gamma(0)B^T = L_1(B\Gamma(0)B^T).$$

In addition, let

$$Y_t^d \triangleq \sum_{i=1-t}^{\infty} \xi_{i,t}^d \gamma_{i,t}^T, \;\; \xi_{i,t}^d \triangleq \left[\prod_{j=1-i-d}^{t-1} (A + \eta_j BL_{(b)}) \right] \eta_{-i-d} B\Delta u_{-i-d}.$$

By similar reasoning, we find that $E[Y_t^d]$ equals

$$L_1\left(\bar{p}_d(A + BL_{(b)})(A + \bar{p}_d BL_{(b)})^{d-1} B\Gamma(d)B^T\right).$$

We argue

$$\text{Cov}(\gamma_t) = \mathbb{E}\left[Z_t + \sum_{d=1}^{\infty} Y_t^d + (Y_t^d)^T \right] = 2\sum_{d=1}^{\infty} \text{sym}[Y_*^d] + L_1(B\Gamma(0)B^T), \tag{C.8}$$

where

$$Y_*^d = L_1\left(\bar{p}_d(A + BL_{(b)})(A + \bar{p}_d BL_{(b)})^{d-1} B\Gamma(d)B^T\right).$$

Step 1b: Calculate $\text{Cov}(\eta_t[L_{(b)}\gamma_t + \Delta u_t])$:

We argue that

$$\mathbb{E}[\eta_t^2 L_{(b)}\gamma_t \Delta u_t^T] = \bar{p}_d^2 L_{(b)} \sum_{d=0}^{\infty} (A + \bar{p}_d BL_{(b)})^d B\Gamma(d+1).$$

Therefore, we obtain

$$\begin{aligned}\text{Cov}(\eta_t[L_{(b)}\gamma_t + \Delta u_1]) = \bar{p}_d \left(\Gamma(0) + L_{(b)}\text{Cov}(\gamma)L_{(b)}^T\right) \\ + 2\text{sym}\left(\bar{p}_d^2 L_{(b)} \sum_{d=0}^{\infty} (A + \bar{p}_d BL_{(b)})^d B\Gamma(d+1) \right),\end{aligned} \tag{C.9}$$

where $\text{Cov}(\gamma) \triangleq \text{Cov}(\gamma_t)$ is given in (C.8). Note, $\text{Cov}(\gamma_t)$ is constant in t. Substituting (C.9) into (C.7), we have

$$\bar{J} = J_{(b)}(p_d) + \text{tr}(\bar{p}_d U \Gamma(0)) + \text{tr}((W + \bar{p}_d L_{(b)}^T U L_{(b)})\text{Cov}(\gamma)) + \text{tr}\left(2U\text{sym}\left(\bar{p}_d^2 L_{(b)} \sum_{d=0}^{\infty} (A + \bar{p}_d B L_{(b)})^d B \Gamma(d+1)\right)\right).$$

Step 2: Calculate $\mathbb{E}[y_k^T y_k' | \mathcal{H}_0]$ *in terms of* $\Gamma(d)$:

Recall from the proof of Theorem 3.3 and (3.47)

$$\mathbb{E}[y_k^T y_k' | \mathcal{H}_0] = \text{tr}\left(C\mathbb{E}[x_k' x_k^T]C^T\right).$$

We observe that $x_k' = \gamma_k$. Thus, from (C.6), we assert

$$\lim_{k\to\infty} \mathbb{E}[y_k^T y_k' | \mathcal{H}_0] = \text{tr}\left(C\text{Cov}(\gamma)C^T\right). \tag{C.10}$$

Step 3: Convert to Frequency Domain:

Optimizing over the autocovariance functions is intractable as there are infinitely many optimization variables. In this case, as in Theorem 2.6, we will leverage Bochner's theorem in [25, p.64] (see also [26]). For ease of presentation, the theorem is repeated here. This theorem provides a frequency domain representation of an autocovariance function of a stationary process:

Theorem C.1 (Bochner's theorem)**.** $\Gamma(d)$ *is an autocovariance function of a stationary Gaussian process* $\{\Delta u_k\}$ *if and only if there exists a unique positive Hermitian measure* ν *of size* $p \times p$ *satisfying*

$$\Gamma(d) = \int_{-0.5}^{0.5} \exp(2\pi j d\omega) d\nu(\omega).$$

Note that a positive Hermitian measure ν takes a Borel set in $[-0.5,\ 0.5]$ and outputs a positive semidefinite Hermitian matrix in $\mathbb{C}^{p\times p}$. We choose to optimize over $\tilde{\Gamma}(d)$, which has bijective relationship with $\Gamma(d)$. By assumption $\tilde{\Gamma}(d)$ is an autocovariance function of a stationary Gaussian process. As a result, we can use Bochner's theorem to rewrite $\tilde{\Gamma}(d)$ in terms of a Riemann sum. Specifically,

$$\tilde{\Gamma}(d) = \lim_{\sigma\to 0} 2\,\mathfrak{Re}\left[\sum_{i=1}^{q} \exp(2\pi j d\omega_i)\tilde{\nu}(I_i)\right], \tag{C.11}$$

where $I_i \cap I_j = \emptyset, \quad \cup_{i=1}^{q} I_i = [0,\ 0.5], \quad \omega_i \in I_i$ and σ is the maximum length of I_i. Here, we also leverage the fact that $\tilde{\Gamma}(d)$ is real. Moreover, from (C.8), we see that

$$\begin{aligned}
\text{Cov}(\gamma) &= \lim_{\sigma \to 0} \sum_{i=1}^{q} \left(2\,\mathfrak{Re}\left[2\text{sym}\Big(L_1 \Big[\bar{p}_d \exp(2\pi j \omega_i) \Omega_1 B \tilde{\nu}(I_i) B^T \Big] \Big) + L_1 \Big[B \tilde{\nu}(I_i) B^T \Big] \right] \right) \\
&= \lim_{\sigma \to 0} \sum_{i=1}^{q} \left(2\,\mathfrak{Re}\left[2\text{sym}\Big(L_1 \Big[\bar{p}_d \exp(2\pi j \omega_i) \Omega_2 B \tilde{\nu}(I_i) B^T \Big] \Big) + L_1 \Big[B \tilde{\nu}(I_i) B^T \Big] \right] \right), \\
&= \lim_{\sigma \to 0} \sum_{i=1}^{q} F_2(\omega_i, \tilde{\nu}(I_i), p_d).
\end{aligned}$$

where

$$\begin{aligned}
\Omega_1 &= \bar{\rho}(A + BL_{(b)}) \sum_{d=1}^{\infty} (\bar{\rho} \exp(2\pi j \omega_i)(A + \bar{p}_d B L_{(b)}))^{d-1}, \\
\Omega_2 &= \bar{\rho}(A + BL_{(b)})(I - \bar{\rho} \exp(2\pi j \omega_i)(A + \bar{p}_d B L_{(b)}))^{-1}.
\end{aligned}$$

The inverse is well defined since we showed $(A + \bar{p}_d B L_{(b)})$ is Schur stable. By similar reasoning it can be shown that

$$\bar{J} = J_{(b)}(p_d) + \lim_{\sigma \to 0} \sum_{i=1}^{q} F_1(\omega_i, \tilde{\nu}(I_i), p_d). \tag{C.12}$$

Replacing $\rho(A_\omega) \leq \bar{\rho}$ with Assumption 1 in problem (3.48), we arrive at the following equivalent formulation:

$$\begin{aligned}
\underset{\tilde{\nu}(I_i), p_d}{\text{maximize}} \quad & \lim_{\sigma \to 0} \sum_{i=1}^{q} \text{tr}(C F_2(\omega_i, \tilde{\nu}(I_i), p_d) C^T) \\
\text{subject to} \quad & J_{(b)}(p_d) + \lim_{\sigma \to 0} \sum_{i=1}^{q} F_1(\omega_i, \tilde{\nu}(I_i), p_d) \leq \delta, \\
& 0 \leq p_d \leq 1.
\end{aligned} \tag{C.13}$$

Step 4: Demonstrate Equivalence:

The rest of the result follows from Steps 2 and 3 in the proof of Theorem 2.6 when $p_d < 1$. In particular, we can leverage the linearity of F_2 and F_1 in H for fixed $p_d < 1$ and ω to show that the optimal value of (3.49) is an upper bound on the optimal value for problem (C.13). Then, we show that for Borel set $S_b \subset [-0.5,\ 0.5]$, the measure

$$\tilde{\nu}(S_b) = \mathbb{I}_{\omega_* \in S_b} H_* + \mathbb{I}_{-\omega_* \in S_b} \text{conj}(H_*), \tag{C.14}$$

where $\mathbb{I}$ is the indicator function and conj refers to the complex conjugate, achieves this upper bound. The resulting autocovariance function is

$$\Gamma(d) = 2\bar{\rho}^{|d|}\,\mathfrak{Re}(\exp(2\pi j d\omega_*)H_*), \tag{C.15}$$

and can be generated by the HMM (3.50) if there exists an optimal H_*, which has rank 1. Theorem 7 of [27] demonstrates the existence of such a solution, while the associated proof shows how such a solution can be constructed from an optimal H_* with rank greater than 1. When $p_d = 1$, F_2 and F_1 are identically 0, establishing the equivalence of (3.49) and (C.13) in this scenario. Note also in this case, (if $J_{(b)}(p_d) \leq \delta$) any stationary Gaussian process in the feasible region is optimal since the resulting additive input is immediately dropped. □

Appendix D

Proof of Theorem 4.7

Proof. For simplicity let the sth row of D^a be referred to as D^s. We first proof sufficiency. Suppose $\exists \lambda \in \Lambda^1 \cap \Lambda^2$ and $\alpha_1 \in \mathbb{C}^{\sum_i r_i(\lambda,1)}, \alpha_2 \in \mathbb{C}^{\sum_i r_i(\lambda,2)}$ such that

$$\mathcal{V}_s^{\lambda,1}\alpha_1 = \mathcal{V}_s^{\lambda,2}\alpha_2 \neq 0. \tag{D.1}$$

Let $\bar{V}^{\lambda,j}$ be given by

$$\begin{bmatrix} v_{1,1}^{\lambda,j} & v_{2,1}^{\lambda,j} & \cdots & v_{r_1,1}^{\lambda,j} & v_{1,2}^{\lambda,j} & v_{2,2}^{\lambda,j} & \cdots & v_{r_2,2}^{\lambda,j} & \cdots & v_{1,l_{\lambda,j}}^{\lambda,j} & v_{1,l_{\lambda,j}}^{\lambda,j} & \cdots & v_{r_{l_{\lambda,j}},l_{\lambda,j}}^{i,j} \end{bmatrix},$$

and let $x_0^a(j) = \bar{V}^{\lambda,j}\alpha_j$. Suppose $D^s d_k^a(j) = C^s(j)A(j)^k x_0^a(j)$. It can be shown that for $k \geq 0$

$$D^s d_k^a(j) = \begin{cases} \frac{\lambda^k}{0!}\mathcal{V}_{s,1}^{\lambda,j}\alpha_j + \frac{1}{1!}\frac{d}{d\lambda}(\lambda^k)\mathcal{V}_{s,2}^{\lambda,j}\alpha_j + \cdots \frac{1}{k!}\frac{d^k}{d\lambda^k}(\lambda^k)\mathcal{V}_{s,k+1}^{\lambda,j}\alpha_j & k \leq r(\lambda) - 1 \\ \frac{\lambda^k}{0!}\mathcal{V}_{s,1}^{\lambda,j}\alpha_j + \frac{1}{1!}\frac{d}{d\lambda}(\lambda^k)\mathcal{V}_{s,2}^{\lambda,j}\alpha_j + \cdots + \frac{1}{(r(\lambda)-1)!}\frac{d^{r(\lambda)-1}}{d\lambda^{r(\lambda)-1}}(\lambda^k)\mathcal{V}_{s,r(\lambda)}^{\lambda,j}\alpha_j, & k \geq r(\lambda), \end{cases} \tag{D.2}$$

where $\mathcal{V}_{s,k}^{\lambda,j}$ is the kth row of $\mathcal{V}_s^{\lambda,j}$. From (D.1), $D^s d_k^a(1) = D^s d_k^a(2)$. Moreover, using (D.1), it can be inductively shown that the attack is nonzero for some time k. If the attack is purely real, the result holds. If $D^s d_k^a(j)$ is 0 or purely imaginary for all k, then α_1 and α_2 can be scaled by a factor of i and the result will hold with a purely real attack. Finally, if alternatively, $D^s d_k^a(j)$ contains both real and imaginary components, then an attack can be constructed by adding the conjugate so

that

$$D^s d_k^a(j) = C^s(j)A(j)^k x_0^a(j) + \overline{C^s(j)A(j)^k x_0^a(j)} = C^s(j)A(j)^k(x_0^a(j) + \overline{x_0^a(j)}).$$

Therefore,

$$D^s d_k^a(1) = C^s(1)A(1)^k(x_0^a(1) + \overline{x_0^a(1)}) = C^s(2)A(2)^k(x_0^a(2) + \overline{x_0^a(2)}) = D^s d_k^a(2),$$

which will be nonzero since $C^s(j)A(j)^k x_0^a(j)$ has real components for some $k \geq 0$. Thus, the result holds.

We now prove the necessary assumption. Without loss of generality, suppose the first z eigenvalues of Λ^1 and Λ^2 are the same so that $\lambda_k^1 = \lambda_k^2$ for $k \leq z$. Assume the rest of the eigenvalues are different. In particular let $\Lambda^1 = \{\lambda_1, \cdots, \lambda_{q_1}\}$ and $\Lambda^2 = \{\lambda_1, \cdots, \lambda_z, \lambda_{q_1+1}, \cdots, \lambda_{q_1+q_2-z}\}$. Let $r^*(\lambda, j) = \max_i r_i(\lambda, j)$, characterize the maximum block size of eigenvalue λ for $A(j)$ and let $\tau + 1 \geq 2n$.

Define $G(\lambda_i, j) \in \mathbb{C}^{\tau+1 \times r^*(\lambda_i, j)}$ as

$$\begin{bmatrix} 1 & 0 & \cdots & 0 \\ \lambda_i & 1 & \cdots & 0 \\ \lambda_i^2 & 2\lambda_i & \cdots & 0 \\ \vdots & \vdots & \cdots & \vdots \\ \lambda_i^\tau & \tau\lambda_i^{\tau-1} & \cdots & \frac{1}{(r^*(\lambda_i,j)-1)!}\frac{d^{r^*(\lambda_i,j)-1}}{dr^*(\lambda_i,j)-1}(\lambda_i^\tau) \end{bmatrix}, \tag{D.3}$$

where the kth column is obtained by taking entrywise, the corresponding $(k-1)$ derivative of the associated entry in the first column and dividing by $(k-1)!$. Let G_a, G_b, G_c be given by

$$G_a = \begin{bmatrix} G(\lambda_1, 1) & G(\lambda_1, 2) & \cdots & G(\lambda_z, 1) & G(\lambda_z, 2) \end{bmatrix},$$
$$G_b = \begin{bmatrix} G(\lambda_{z+1}, 1) & \cdots & G(\lambda_{q_1}, 1) \end{bmatrix},$$
$$G_c = \begin{bmatrix} G(\lambda_{q_1+1}, 2) & \cdots & G(\lambda_{q_1+q_2-z}, 2) \end{bmatrix}.$$

Finally, let $G^* = \begin{bmatrix} G_a & G_b & G_c \end{bmatrix}$. Note that $G^* \in \mathbb{C}^{\tau+1 \times \kappa}$ where $\kappa \leq 2n$ by construction.

Consider vectors $\eta^{i,j} \in \mathbb{C}^{\sum_k r_k(\lambda_i,j)}$ and define $\tilde{\mathcal{V}}_s^{\lambda_i,j}, \tilde{\mathcal{V}}_s^a, \tilde{\mathcal{V}}_s^b, \tilde{\mathcal{V}}_s^c, \tilde{\mathcal{V}}_s$ as

$$\begin{aligned}
\tilde{\mathcal{V}}_s^{\lambda_i,j} &= \begin{bmatrix} \mathcal{V}_{s,1}^{\lambda_i,j}\eta^{i,j} & \cdots & \mathcal{V}_{s,r^*(\lambda_i,j)}^{\lambda_i,j}\eta^{i,j} \end{bmatrix}^T, \\
\tilde{\mathcal{V}}_s^a &= \begin{bmatrix} (\tilde{\mathcal{V}}_s^{\lambda_1,1})^T & (\tilde{\mathcal{V}}_s^{\lambda_1,2})^T & \cdots & (\tilde{\mathcal{V}}_s^{\lambda_z,1})^T & (\tilde{\mathcal{V}}_s^{\lambda_z,2})^T \end{bmatrix}^T, \\
\tilde{\mathcal{V}}_s^b &= \begin{bmatrix} (\tilde{\mathcal{V}}_s^{\lambda_{z+1},1})^T & \cdots & (\tilde{\mathcal{V}}_s^{\lambda_{q_1},1})^T \end{bmatrix}^T, \\
\tilde{\mathcal{V}}_s^c &= \begin{bmatrix} (\tilde{\mathcal{V}}_s^{\lambda_{q_1+1},2})^T & \cdots & (\tilde{\mathcal{V}}_s^{\lambda_{q_1+q_2-z},2})^T \end{bmatrix}^T, \\
\tilde{\mathcal{V}}_s &= \begin{bmatrix} (\tilde{\mathcal{V}}_s^a)^T & (\tilde{\mathcal{V}}_s^b)^T & (\tilde{\mathcal{V}}_s^c)^T \end{bmatrix}.
\end{aligned}$$

From Theorem 4.6, the attack is not unambiguously identifiable up to time τ if and only if there exists real vectors x_0^1 and x_0^2 such that

$$C(1)A(1)^k x_0^1 = C(2)A(2)^k x_0^2, \quad 0 \leq k \leq \tau. \tag{D.4}$$

with $C(1)A(1)^k x_0^1 \neq 0$ for some time in $0 \leq k \leq \tau$. It can be shown that (D.4) holds only if there exists vectors $\eta^{i,j} \in \mathbb{C}^{\sum_k r_k(\lambda_i,j)}$ such that $G^*\tilde{\mathcal{V}}_s = 0$. We now analyze the null space of G^*. We observe by construction that G^* has $r_{min} = \sum_{i=1}^{z} \min_j r^*(\lambda_i, j)$ pairs of identical columns. Thus, null$(G^*) \geq r_{min}$. Let $\tilde{G}^* \in \mathbb{C}^{\tau+1 \times r_{max}}$ be obtained by deleting duplicate columns of G^* where $r_{max} \leq 2n \leq \tau + 1$ is given by

$$r_{max} = \sum_{i=1}^{z} \max_j r^*(\lambda_i, j) + \sum_{i=z+1}^{q_1} r^*(\lambda_i, 1) + \sum_{i=q_1+1}^{q_2+q_1-z} r^*(\lambda_i, 2).$$

Let $\tilde{G^*}_{trunc}$ be a square matrix obtained by removing the last $\tau + 1 - r_{max}$ rows of $\tilde{G}^*$.

We first show that the null space $\tilde{G^*}_{trunc}^T$ is empty. Suppose it was not. This would imply the existence of a complex nonzero polynomial $p^*(x)$ of degree $r_{max} - 1$ with the property

$$p^*(\lambda_k) = 0, \frac{dp^*}{dx}(\lambda_k) = 0, \cdots, \frac{d^{\max_j r^*(\lambda_k,j)-1}p^*}{dx^{\max_j r^*(\lambda_k,j)-1}}(\lambda_k) = 0,$$

for $1 \leq k \leq z$,

$$p^*(\lambda_k) = 0, \frac{dp^*}{dx}(\lambda_k) = 0, \cdots, \frac{d^{r^*(\lambda_i,1)-1}p^*}{dx^{r^*(\lambda_k,1)-1}}(\lambda_k) = 0,$$

for $z+1 \leq k \leq q_1$, and

$$p^*(\lambda_k) = 0, \frac{dp^*}{dx}(\lambda_k) = 0, \cdots, \frac{d^{r^*(\lambda_i,2)-1}p^*}{dx^{r^*(\lambda_k,2)-1}}(\lambda_k) = 0,$$

for $q_1 + 1 \leq k \leq q_1 + q_2 - z$.

But this contradicts the fundamental theorem of algebra since it would imply a polynomial of degree $r_{max} - 1$ has r_{max} zeros. Thus, the null space of $\tilde{G^*}^T_{trunc}$ is empty.

Therefore, by the rank nullity theorem $\tilde{G^*}_{trunc}$ is full rank and therefore $\tilde{G^*}$ is full rank. Consequently, rank$(G^*) \geq r_{max}$. However, null$(G^*) \geq r_{min}$ and the number of columns in G^* is $r_{max} + r_{min}$. Therefore, strict equality holds and rank$(G^*) = r_{max}$ and null$(G^*) = r_{min}$. As a result, one excites the null space of G^* only by exciting pairs of identical columns in G^*.

Thus (D.4) holds only if there exists $\eta^{i,j} \in \mathbb{C}^{\sum_k r_k(\lambda_i,j)}$ such that for $1 \leq i \leq z$

$$\begin{bmatrix} \mathcal{V}_{s,1}^{\lambda_i,1} \\ \mathcal{V}_{s,2}^{\lambda_i,1} \\ \vdots \\ \mathcal{V}_{s,\min_j r^*(\lambda_i,j)}^{\lambda_i,1} \end{bmatrix} \eta^{i.1} + \begin{bmatrix} \mathcal{V}_{s,1}^{\lambda_i,2} \\ \mathcal{V}_{s,2}^{\lambda_i,2} \\ \vdots \\ \mathcal{V}_{s,\min_j r^*(\lambda_i,j)}^{\lambda_i,2} \end{bmatrix} \eta^{i.2} = 0,$$

and

$$\begin{bmatrix} \mathcal{V}_{s,\min_j r^*(\lambda_i,j)+1}^{\lambda_i,\arg\max_j r^*(\lambda_i,j)} \\ \vdots \\ \mathcal{V}_{s,\max_j r^*(\lambda_i,j)}^{\lambda_i,\arg\max_j r^*(\lambda_i,j)} \end{bmatrix} \eta^{i.\arg\max_j r^*(\lambda_i,j)} = 0.$$

Moreover, for $z+1 \leq i \leq q_1$,

$$\begin{bmatrix} \mathcal{V}_{s,1}^{\lambda_i,1} \\ \mathcal{V}_{s,2}^{\lambda_i,1} \\ \vdots \\ \mathcal{V}_{s,r^*(\lambda_i,1)}^{\lambda_i,1} \end{bmatrix} \eta^{i.1} = 0, \tag{D.5}$$

and for $q_1 + 1 \leq i \leq z$

$$\begin{bmatrix} \mathcal{V}_{s,1}^{\lambda_i,2} \\ \mathcal{V}_{s,2}^{\lambda_i,1} \\ \vdots \\ \mathcal{V}_{s,r^*(\lambda_i,2)}^{\lambda_i,2} \end{bmatrix} \eta^{i.2} = 0, \tag{D.6}$$

For $1 \leq i \leq z$, this can be rewritten as

$$\mathcal{V}_s^{\lambda_i,1}\eta^{i,1} + \mathcal{V}_s^{\lambda_i,1}\eta^{i,2} = 0.$$

From (D.5) and (D.6), we can see that $C(1)A(1)^k x_0^1 \neq 0$ for some time in $0 \leq k \leq \tau$, while (D.4) holds only if there exists $1 \leq i \leq z$ such that

$$\mathcal{V}_s^{\lambda_i,1}\eta^{i,1} + \mathcal{V}_s^{\lambda_i,1}\eta^{i,2} = 0, \quad \mathcal{V}_s^{\lambda_i,1}\eta^{i,1} \neq 0.$$

If the condition does not hold, since $\tau + 1 \geq 2n$, one can detect an attack at time $k = 2n - 1$ (or given $2n$ measurements). The result immediately follows. □