Towards Safe and Equitable Intelligent Transportation Systems: Leveraging Stochastic Control Theory in Attack Detection

by

Matthew C. Porter

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Mechanical Engineering)
in The University of Michigan
2021

Doctoral Committee:

Assistant Professor Ram Vasudevan, Chair
Associate Professor Anil Aswani, UC Berkeley
Associate Professor Kira Barton
Professor Jerome Lynch

Matthew C. Porter
matthepo@umich.edu
ORCID iD: 0000-0002-1598-3447

DEDICATION

To my grandfather, Robert Keith Dentel (1938-2021), who inspired me to learn continuously and taught me the value of both kindness and hard work.

ACKNOWLEDGEMENTS

Completing my PhD would not have been possible without the support of countless friends, family members, colleagues, and mentors. Despite my best attempt, I am sure to miss some of you when writing this section. However, please know that I appreciate your support all the same.

First, I would like to acknowledge my PhD advisor, Ram Vasudevan. Ram, I will admit to thinking you were a graduate student instructor when you first walked in to the class design of digital control systems. However, I could not have asked for a better mentor. It is obvious that you strive for excellence in all that you do, and you inspire others to do the same. You have given me the confidence to learn topics I once thought were out of reach and have helped me to realize my goals.

Second, I would like to acknowledge my collaborators from the University of California Berkeley: Anil Aswani and Pedro Hespanhol. It is because of you that I first started researching dynamic watermarking. Over the past couple years, I have truly appreciated the wealth of ideas and teamwork you both have brought to the table. It has been a pleasure to work with both of you.

Third, I would like to acknowledge the Master's and undergraduate students who have worked with me over the years: Fan Bu, Arnav Joshi, Sidhartha Dey, and Qirui Wu. Through many a late night, we have pulled off some particularly impressive demos. Without you all I would probably still be sitting in lab running experiments.

Fourth, I would like to acknowledge my friends from Michigan Tech: Brett Billington, Joshua Manela, Daniel Sullivan, Daniel Laforest, and Michael Spenle. Thank you all for the various shenanigans, bonfires, and road trips. Being able to get away from school every so often has kept me sane, and of course was tons of fun.

Fifth, I would like to acknowledge the members of the Ford Center for Autonomous Vehicles and the Robotics and Optimization for the Analysis of Human Motion Lab. In particular, I would like to acknowledge my fellow members of the Monday morning procrastination crew Alexa Carlson and Cyrus Anderson. I have yet to find a stronger cup of coffee or a group of people so capable of developing such strange ideas so early in the morning.

Sixth, I would like to acknowledge my family. Tuition payments, constantly changing housing arrangements, and unreliable vehicles sure do cause a lot of struggles for a perpetual student. However, your love and support has helped me through many a struggle and for that I am deeply thankful.

Seventh, I would acknowledge my wife Stephanie. Steph, thank you for putting up with my occasional hyper-focus on school work and my poor estimation of graduation timelines. You have been my steadfast support through it all and I cant wait to see what life brings us next.

Lastly, I would like to acknowledge the Ford Motor Company for supporting this work via the Ford-UM Alliance under award N022977.

TABLE OF CONTENTS

LIST OF FIGURES

FIGURE

LIST OF TABLES

TABLE

LIST OF ACRONYMS

3GPP 3rd generation partnership project

C-V2X cellular vehicle to everything

CAV connected and/or autonomous vehicle

CPS cyber-physical system

CUSUM cumulative sum

DOS denial of service

DSRC dedicated short range communication

GNSS global navigation satellite system

IID independent identically distributed

ITS intelligent transportation system

ITS-G5 intelligent transportation systems in the 5GHz frequency spectrum

LiDAR light detection and ranging

LTI linear time-invariant

LTV linear time-varying

MEWMA multivariate exponentially weighted moving average

MIMO multiple input multiple output

RAIM receiver autonomous integrity monitoring

SISO single input single output

UMTRI University of Michigan Transportation Research Institute

V2V vehicle to vehicle

V2X vehicle to everything

WAVE wireless access in vehicular environments

ABSTRACT

Intelligent transportation systems (ITSs) promise to significantly reduce traffic congestion while simultaneously improving road user safety. To accomplish this, ITSs make real-time control decisions using data collected from surrounding vehicles, infrastructure, and other networks. However, this reliance on networked communications results in vulnerabilities to cyber-attacks.

Traditional cyber-security methods such as encryption can often be used to secure messages sent over networks, but are unable to evaluate the trustworthiness of a message's source. As a result, these methods do not protect against attackers that distribute false data. In ITSs such an attack could be conducted for personal gain such as reducing the attackers travel time or malicious intent such as causing collisions and congestion. To address this threat, previous attack detection algorithms check for anomalies in the incoming data using a dynamical model of the system. However, these algorithms are limited to linear time-invariant models and are insufficient to describe the nonlinear dynamics that are present in many ITSs. In addition, most existing algorithms are unable to detect attacks that use knowledge of the system to generate false data.

To address these challenges, this dissertation develops a family of attack detection algorithms called dynamic watermarking. Dynamic watermarking introduces a watermark in the form of slightly altered control decisions. The incoming data is then checked for anomalies and for the presence of the watermark. In doing so, dynamic watermarking provably detects a wide range of sophisticated attacks. The proposed family of attack detection algorithms can be applied to a wide variety of applications including ITSs. Furthermore, this work develops tools for applying dynamic watermarking in real-world settings. These tools allow the user to tune the detection algorithms sensitivity and to approximate application specific parameters that are used to detect anomalies.

The effectiveness of the proposed dynamic watermarking-based algorithms and tools are then illustrated for autonomous vehicle localization and cooperative vehicle platooning. In each application, the proposed algorithm is shown to enable safe and equitable operation of the ITS even in the presence of false data. Since the proposed algorithms are derived using arbitrary dynamical models, their potential applications in ITSs and other cyber-physical systems are vast.

Chapter 1

Introduction

Traffic congestion is a growing global problem that leads to both economic waste and negative environmental impacts. In 2019 alone, the average American spent 99 hours stuck in traffic contributing to an estimated $88 billion dollars in lost time nationally [1]. Moreover, Americans wasted approximately 3.3 billion gallons of fuel as a result of congestion in 2017 [2]. The negative environmental effects of this added fuel consumption is considerable. One study suggests that CO2 emissions from road vehicles can be reduced by nearly 20% by taking steps to reduce congestion [3]. Additionally, despite numerous advances in vehicle safety, yearly fatalities from traffic collisions in the US continue to exceed 30 thousand [4]. Intelligent transportation systems (ITSs) aim to mitigate these problems by leveraging data from a multitude of sources to enable real-time traffic control decisions.

Applications of ITSs come in various forms including collaborative highway maneuvers, sharing of road safety information, optimization of traffic signals, and automated driving. In the past decade, the US department of transportation has implemented pilot programs to better understand the challenges and benefits of ITSs. These include the Safety Pilot Model Deployment project lead by the University of Michigan Transportation Research Institute (UMTRI) which ran from 2011 to 2014 [5], and the Connected Vehicle Pilot Deployment Program which started in 2015 and has implemented various technologies in New York City, Tampa Florida, and along interstate 80 in Wyoming [6]–[8]. While the specifics in each application vary, one common requirement is networked communications to transfer relevant information. Unfortunately, this requirement leads to cyber-security concerns, which have yet to be fully considered.

Cyber-attacks on large scale infrastructure are not a new phenomenon. In fact, there have been several well documented occurrences targeting everything from nuclear facilities to power grids [9]–[12]. Moreover, these systems are often protected by traditional cyber-security tools but these methods are insufficient due to the addition of networked physical infrastructure [13], [14]. Cyber-attacks on ITSs are likely to present an even bigger challenge [15]–[17]. Namely,

connected vehicles will rely on messages from external sources to execute safety-critical actions and to improve efficiency. In some cases light-weight encryption methods can be implemented to authenticate the identity of a message's source, however there is no guarantee that the source is truthful. The content of these messages must therefore be verified in a quick and efficient manner to ensure safe and equitable operation. For current large scale infrastructure, this can be accomplished using attack detection schemes that look for anomalous information [18]–[21]. However, these existing methods are unable to handle the inherent complexities of ITSs.

This dissertation addresses the challenges of cyber-security by developing an attack detection scheme, called dynamic watermarking, that leverages stochastic control theory to robustly detect false information. The proposed method is shown to be applicable to a variety of ITS applications both in simulation and through real world experiments. The remainder of this chapter is outlined as follows. Section 1.1 describes how dynamic watermarking functions and the challenges in applying it to ITSs. Section 1.2 reviews relevant literature on attack detection and how these methods relate to the security vulnerabilities in ITSs. Section 1.3 discusses the current and expected contributions of this dissertation. Section 1.4 lists the planned chapters of this dissertation.

1.1 Overview

The field of attack detection is rooted in the related field of anomaly detection. In fact, several attack detection schemes are indistinguishable from those used for anomaly detection. Generally speaking, attack detection schemes consider the residuals of the sensor measurements with respect to their estimated values. For simplicity, we refer to this quantity as the *measurement residual*. By making an assumption on the distribution of the measurement residuals, detection schemes impose a hypothesis test to determine if an attack is occurring.

Similarly, dynamic watermarking uses a hypothesis test on the measurement residuals. However, it also adds a watermark to the controller's inputs as a means to further validate the returned measurements as illustrated in Figure 1.1. In the proposed methods, the watermark takes the form of a pseudo-random Gaussian signal that is generated by the detector. Since the watermark is added to the controller's inputs, the watermark perturbs the state of the plant which is subsequently measured by the sensors and returned. This allows the detector to differentiate between authentic measurements and fabricated measurements that are seemingly reasonable by checking the correlation between the measurements and the watermark.

Figure 1.1, illustrates the basic system structure that we initially use to develop dynamic watermarking [22]. For this system, the plant is assumed to be a multiple input multiple output (MIMO) and linear time-invariant (LTI). Furthermore the attacker is assumed to have complete access to the authentic measurements and full control over altering them. While this structure acts as a good

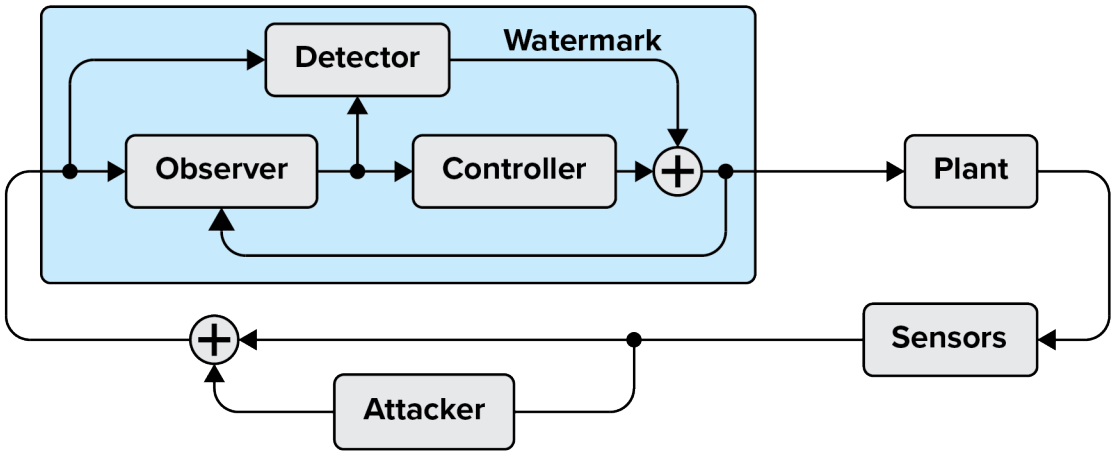
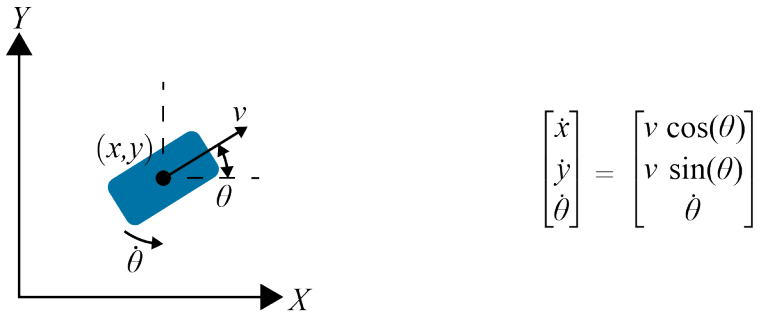


Figure 1.1: Block diagram of dynamic watermarking detection for a simple system with attacked sensor measurements.

starting point, it does not immediately extend to ITS. In particular, a LTI model is too restrictive for many ITS applications and the assumption of a controller is insufficient for collaborative actions such as platooning. These hurdles are discussed in more detail in the remainder of this section.

1.1.1 Plant Model Complexity

Vehicle dynamics can be incredibly complex, and as a result simplified models are often used. One such example is the Dubin's car model, illustrated in Figure 1.2, which travels in a 2 dimensional plane taking the forward velocity v and the angular velocity $\dot{\theta}$ as inputs.



$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} v\cos(\theta) \\ v\sin(\theta) \\ \dot{\theta} \end{bmatrix}$$

Figure 1.2: Dubin's car model with ground plane coordinates (x, y), heading θ and inputs of forward velocity v and angular velocity $\dot{\theta}$.

Though this simplistic model can be used in some applications, attack detection often require a more exact model of the underlying system dynamics. Nonetheless, linearizing these dynamics

about a given trajectory does not result in a LTI system unless the velocity is constant.

Ideally, we could apply dynamic watermarking to non-linear systems. However, this would greatly increase the complexity of the detection algorithm and deprive it of many of the tools associated with linear systems. In this work, we develop dynamic watermarking for LTV systems as it providers a convenient middle ground. Namely, it allows us to apply dynamic watermarking to increasingly complex ITS applications without losing the rich tool set associated with linear systems.

1.1.2 Networked Agents

While developing dynamic watermarking for increasingly complex plant dynamics allows for applications with a single vehicle, ITSs are particularly useful due to collaborative actions using V2X communications. In these applications, the plant may be a set of vehicles, where each acts as a subcontroller and broadcasts its measurements as illustrated in Figure 1.2. Systems such as these

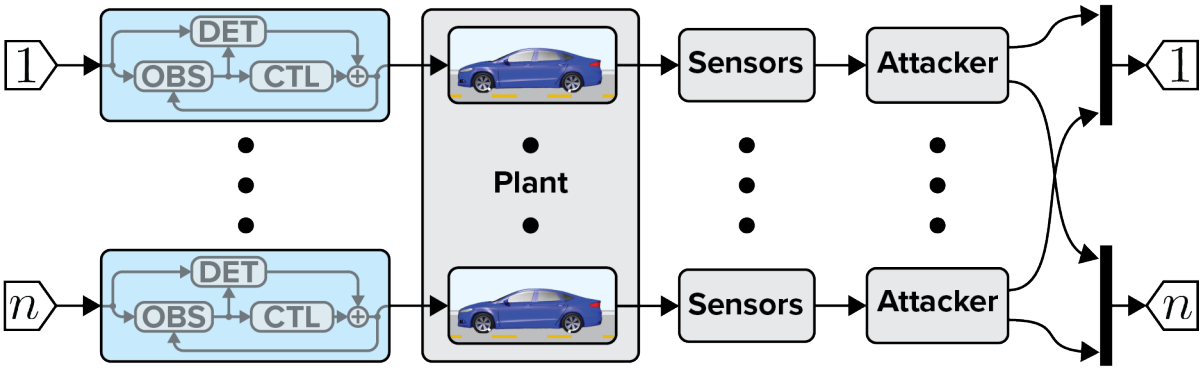


Figure 1.3: Block diagram of dynamic watermarking for a collaborative network of n CAVs. Each vehicle has its own observer, controller, detector, and watermark. The attacker is modeled as being in the loop since the attacker may be an untrustworthy vehicle. Measurements from each vehicle are shared using V2V communications. Watermarks are not shared with other vehicles allowing them to be used to validate measurements from other vehicles.

have an additional cyber-security threats in the form of non-truthful agents. To counter this threat, each vehicle generates and adds a watermark to its inputs but does not share the watermark with other vehicles. Attack detection is then carried out independently by each vehicle.

The result of the watermark on the communicated measurements is more complex than in the single vehicle case. Namely, depending on the application, a given vehicle's watermark may have no effect on another vehicles measurements. For example, in a platooning operation the watermark of a given vehicle may not affect the state of preceding vehicles. Moreover, each vehicle may only sense measurements corresponding to itself and adjacent vehicles. As a result, the watermark of a

given vehicle will never have an affect on the measurements of a vehicle several positions ahead in the platoon. Therefore, to ensure detection of attacks in these applications additional structure is needed. In this dissertation, we develop dynamic watermarking for networked systems that ensures detection of attacks by non-truthful agents.

1.2 Literature Review

This section discusses relevant literature in three categories. First, subsection 1.2.1 discusses potential vulnerabilities in ITS. Second, subsection 1.2.2 compares the various categories of attack models. Third, subsection 1.2.3 describes the state of the art in attack detection algorithms.

1.2.1 ITS Security Vulnerabilities

The focus of this dissertation is on detecting and mitigating attacks on ITS. To this end, various attack models are considered without discussion of how such an attack might be carried out. Nonetheless, this section provides a brief non-exhaustive review of ITS vulnerabilities to cyber-attacks.

On-Board Computers

Connected and/or autonomous vehicles (CAVs) use on-board computers to handle sensor measurements and actuator inputs. As a result, a hacker can take full control of the vehicle by compromising the security of these computers. Though modern cars have few avenues for accessing the on-board computer, they have been shown to be susceptible to hacking through wirelessly connecting to the infotainment system [23] or through a wireless receiver connected to the diagnostic port [24]. While these types of attacks are certainly troubling, they usually require some level of physical presence or a lack of sufficient isolation of safety-critical control systems from unrelated sub-systems.

GNSS Positioning

Broadcasting false global navigation satellite system (GNSS) signals is not exceedingly complex and can be accomplished with consumer off the shelf components [25]. Moreover, many probable occurrences of large scale GNSS spoofing have been documented in recent news articles [26]–[29]. However, generating signals that can deceive a GNSS receiver while evading detection has many challenges especially when various methods of spoofing detection are used. One common detection method is receiver autonomous integrity monitoring (RAIM) which uses redundant satellites to

check for inconsistencies. More sophisticated detection methods may use additional signal characteristics such as the signal amplitude, sensor arrays to measure the polarization and angle of arrival of each signal, or various other methods [30], [31]. Despite these advances in detection methods, researchers have demonstrated the feasibility of attacks that are capable of remaining stealthy in the presence of some or all of these methods [32]–[34]. Moreover, at the time of writing, manufacturers of commercial GNSS receivers have yet to implement more than the most rudimentary of spoofing detection methods [35].

Vehicle Localization

CAVs can approximate their localization by comparing feature points found by on-board sensors such as cameras and light detection and ranging (LiDAR) to features defined in high definition maps. The vulnerabilities of these approximations come in two forms. Namely, attacks that alter the features found by on-board sensors and those that alter the high definition map.

Altering Camera and LiDAR measurements: While cameras are susceptible to glare [36], further vulnerabilities lie in deceiving the object classifier that is run on the resulting images [37]. Furthermore, fabricated light detection and ranging (LiDAR) returns can be injected via lasers in an efficient enough manner to fool object detection algorithms [38]. In each of these cases, the features found by the sensors are altered and as a result the localization approximation can be compromised.

Altering High Definition Maps: High definition maps are comprised of features and their corresponding locations. Some of these features remain static such as the corner of a building, while others may change. As a result, high definition maps must be updated periodically. Moreover, maps from previously un-visited regions may need to be downloaded while driving. An attacker can use the process of downloading a new or updated map to distribute an altered map in which the location of feature points has been changed. In doing so, the localization approximation of CAVs can be altered. Specifics on the vulnerabilities in networked vehicular communications such as those used to download high definition maps is further discussed under the topic of V2X communications.

V2X Communications

Though several methods of facilitating vehicle to everything (V2X) communications have been proposed, the apparent leaders utilize IEEE 802.11p [39] or cellular vehicle to everything (C-V2X) which was originally standardized by release 14 of the 3rd generation partnership project (3GPP) [40]. In particular, the U.S. department of transportation's dedicated short range communication (DSRC) project and the European Telecommunications Standards Institute's intelligent transporta-

tion systems in the 5GHz frequency spectrum (ITS-G5) standard are based on 802.11p. Moreover, C-V2X is gaining momentum as a potential alternative or complementary technology to 802.11p based communications though its adoption by regulatory bodies is currently less widespread. The overall strengths and weaknesses of each technology, which are greatly debated, fall outside the scope of this work. However, their security standards and vulnerabilities are of particular interest.

In the United States, the security standards for 802.11p fall under the umbrella of the IEEE 1609.0 wireless access in vehicular environments (WAVE) standard [41]. Specifically, IEEE 1609.2 outlines security standards mostly comprised of network structure and encryption options [42]. With the exception of unforeseen security gaps, encryption should suffice to validate the identity of networked agents. However, it does not guarantee that the content of communications is truthful. In contrast, security practices for C-V2X are still lacking in formal standardization and acceptance. As a result, C-V2X has additional vulnerabilities that may allow attackers to carry out identity spoofing and denial of service attacks [43], [44].

1.2.2 Attack Models

Generally, cyber-attacks take one of four forms: *denial of service* (DOS) attacks where communications are disrupted, *hijacking attacks* in which the attacker attempts to take control of part or all of the system, *confidentiality attacks* where an attacker attempts to collect private information from intercepted communications, and *deception attacks* where communications are altered in an attempt to deceive [45]. DOS attacks can be detrimental, but if they stop all communication, they are trivial to detect and when only a portion of communication is stopped, their effects can be minimized using graceful degradation [46]. Furthermore, Hijacking and Confidentiality attacks can be avoided using standard security measures such as encryption. In contrast, deception attacks are less straightforward to detect and avoid. Moreover, by altering communicated measurements, an attacker can influence the actions of agents in the system. As a result, this dissertation focuses on the detection of deception attacks.

Several types of deception attacks have been described in literature. The simplest deception attacks add noise using arbitrary or random strategies [47]. For bias injection attacks, the attacker injects a constant bias into the system [48]. Routing attacks send measurement signals through a linear transform [49]. Other attacks attempt to decoupled the system such that the measurements are unaltered while certain states of the system are attacked [50]. Zero-dynamics attacks take advantage of un-observable states or remove the effects of their attacks in the resulting measurement signal [48]. Replay attacks involve an attacker replaying recorded measurements while possibly altering control as well [48].

The amount of knowledge of the system dynamics and detection scheme, along with the capa-

bility of the attacker to alter certain signals necessary to carry out these attacks varies greatly. While random, bias injection, routing, and replay attacks do not require any knowledge of the underlying system dynamics, decoupling and zero-dynamics attack require almost full knowledge. This knowledge can be difficult to obtain for non-insider attackers but it is not impossible [51], [52]. Nonetheless, this dissertation focuses on replay attacks due to the simplicity of implementation and its use in real-world attacks [9]. Furthermore, we consider attacks that only alter measurement signals since many of the systems we care about use local controllers while operating using externally received measurements.

1.2.3 Attack Detection Algorithms

In recent literature, the topic of anomaly detection has been extensively studied for the purpose of fault detection and machine health monitoring in manufacturing. While some consider the problem of anomaly detection from a purely data-driven perspective [53]–[56], others compare sensor measurements to their expected value using dynamic models or other contextual information [57]–[60]. Almost all attack detection algorithms follow the latter approach and use the measurement residual, defined as the difference between the measurement and the expected measurement. However, an adversary may generate an attack that does not result in anomalous measurements as discussed in Subsection 1.2.2. As a result, anomaly detection algorithms re-purposed for detecting attacks are inadequate for ensuring safe and equitable operation. The remainder of this subsection discusses the advances in the field of attack detection algorithms to address more sophisticated attacks.

Generally, attack detection algorithms can be separated into two categories: those that only observe the system, called *passive methods*, and those that alter the system while observing, called *active methods*. While the passive methods do not degrade control performance, active methods have been shown, in some cases, to be able to detect more complex attacks with minimal performance degradation [61]–[63]. These categories can be further subdivided into *stateful* and *stateless* detectors. While stateless detectors only consider the current residual, stateful detectors rely on previous measurement residuals as well.

Passive Methods

The χ^2 detector uses the inner product of the normalized measurement residual which follows a χ^2 distribution. Due to its simplicity, the χ^2 detector has been studied in several works. Under the assumption that an attacker cannot increase the probability of an alarm under a χ^2 detector, the ability of an attacker to affect the system can be approximated [64], [65]. Sufficient and necessary conditions for several types of attacks to avoid detection by a χ^2 detector have also been derived [66]. Furthermore, extensions of the χ^2 detector to non-Gaussian noise have been considered [67].

While the χ^2 is widely used, it is a stateless detector and has difficulty detecting small yet persistent changes without also being overly sensitive to the inherent noise of stochastic systems.

Two stateful passive detectors are the cumulative sum (CUSUM) detector which looks at the decaying sum of the χ^2 test metric and the multivariate exponentially weighted moving average (MEWMA) detector which uses the exponentially weighted average of the measurement residual. When comparing these stateful detectors to the χ^2 detector, it has been shown that the stateful detectors can often provide stronger guarantees on detection while the χ^2 detector boasts both simpler implementation and generally takes less time to detect attacks [68], [69].

Active Methods

As an alternative to passive methods, several active methods have also been proposed. Most active methods fall into one of two categories: *moving target defense*, which change the parameters of the system in an effort to keep attackers from having full knowledge of the system, and *watermarking-based methods* which encrypt measurement signals with a watermark that is added to the control input.

Moving Target Defense: The concept of moving target defenses has been a topic of continued interest for the field of cyber security and, among other things, can take the form of randomizing the order of code execution and physical memory storage locations [70]. In CPS, moving target defense takes the form of altering the dynamics of the system over time to increase the attacker's uncertainty of the current configuration. For systems that have redundant measurements, moving target defenses that select subsets of measurements that maintain observability have been well studied [19], [20], [71]–[73]. Similarly, changing in the control inputs to the system either by altering the control strategy or by altering the input matrix for linear systems is also well studied [71], [73]. Furthermore, there has been some consideration of altering the system dynamics [19], [20], [71]. In the first two cases, and possibly the third, the resulting system has some amount of performance degradation as a result of the alterations. While many works do not attempt to mitigate this degradation, others choose to favor an optimal strategy to minimize the drop in performance [73].

An alternative to changing the dynamics of the physical system is to append their dynamics with an auxiliary system [74]–[76]. The benefit of these methods is that the operation of the performance of the original system under normal operation is unchanged. The auxiliary system can have more complex dynamics such as linear time-varying (LTV) [74], periodically switched [75], or nonlinear [76]. Furthermore, in [77] the configuration of an LTV auxiliary system is optimized to improve detection and system estimation performance. Despite the consideration of more complex dynamics in the auxiliary system, these methods have only been applied to systems that have LTI dynamics.

Watermarking-Based Methods: The introduction of a watermark was first proposed as a way of making the χ^2 detector robust to replay attacks [78] and other more advanced attacks [79]. The watermark takes the form of independent identically distributed (IID) Gaussian noise which is added to the control input while the χ^2 detector itself remains unchanged. In Satchidanandan and Kumar [80], the method developed by Mo and Sinopoli [78] is applied to a single input single output (SISO) LTI system with an additional test that looks for correlation with the watermarking signal [80]. By considering the limit of the average value of each test and assuming open loop stability and an attack of non-zero asymptotic power, they prove, in infinite time, almost sure detection of the attack. Furthermore, they are able to provide equivalent results for multiple input multiple output (MIMO) LTI systems given that they are also fully observable by instead considering the outer product of the residual vector in place of the χ^2 detector. In practice, statistical tests are used in place of these limits by applying temporal windowing with a fixed window size for both SISO and MIMO cases. This work is further extended by relaxing the constraint of full observability to partially observable MIMO LTI systems [22], [81]. In [22] the need for the system to be open loop stable is also removed and detection of an attack that is more general than a replay attack is proven in the limit-based form of the test. Furthermore, persistent disturbances are also taken into account to avoid additional false alarms. In several works, an effort to optimize the design of the watermark being added to the control is studied [18], [82]. Other literature has considered allowing the watermark signal to be autocorrelated [83] or non-Gaussian distributed [84]. Additionally, extensions to systems with distributed control are studied in [85]–[87] and extensions to a subset of nonlinear systems is provided in [81], [88]. Other literature has considered watermarks in the form of intentional package drops in the control input signal [89], a combination of package drops and additive Gaussian noise[90], sending measurement signals through parameterized linear transformations [49], [91], [92], and B-splines added to feed forward inputs for output tracking [93].

1.3 Specific Contributions

This section describes published work that has been completed as part of this dissertation [22], [61], [85], [94]–[97]. For each publication we provide an overview of its specific contributions and the impact on making dynamic watermarking feasible for ITSs.

General LTI Systems

The theory of dynamic watermarking was developed for an LTI system with minimal assumptions in Hespanhol *et al.* [22]. While previous works have developed dynamic watermarking for various subsets of LTI systems, our method only requires the system to be controllable and observable.

Under these lesser assumptions, we guarantee detection of replay attacks using limit based tests and provide a statistical test that can be implemented in real-time. Furthermore, we provide a means to handle persistent disturbances such as wind and illustrate the effectiveness of our proposed algorithm on a simulated autonomous vehicle.

Networked LTI Systems

We extended dynamic watermarking to networks of distributed controllers in Hespanhol *et al.* [85]. While others have made similar extensions [80], [88], our method differs in that it allows for partial state observations. Furthermore, sufficient conditions on the control policies are developed to ensure that each agent can detect an attack. Using Heymann's lemma, we derived two methods for designing control policies to meet these conditions. This work enables dynamic watermarking to be implemented on ITSs following the structure shown in Figure 1.3 so long as the underlying system is LTI. The effectiveness of our method is illustrated using a simulated platoon of autonomous vehicles with a constant reference velocity.

Simulated and Real-World Evaluation

Despite the growing amount of literature in detecting deception attacks, few works have attempted to provide real-world implementations leaving the true effectiveness of dynamic watermarking and other attack detection algorithms to speculation. Moreover, to the best of our knowledge no previous metrics for evaluating detection algorithms can be applied to dynamic watermarking. As a result, tuning the parameters present in dynamic watermarking was left to trial and error and comparisons between dynamic watermarking and other algorithms were non-existent. We addressed this lack of real-world testing and tools in Porter *et al.* [61].

Specifically, we developed a metric that measures the capability of an attacker to perturb the state of the system while remaining undetected. We prove that, while the direct computation may be intractable, an over-approximation of this metric can be computed using sum-of-squares programming. The proposed metric and corresponding approximation technique can be applied to a wide range of detection algorithms. Using our proposed metric, we provided a comparison of dynamic watermarking (as described in Hespanhol *et al.* [22]), with three classical detection algorithms: 1) the χ^2 detector, 2) the cumulative sum (CUSUM) detector, and 3) the multivariate exponentially weighted moving average (MEWMA) detector. Due to the approximations we were unable to make a strong conclusion from our comparison. However, the capability of the attacker under dynamic watermarking is shown to be similar to that of other detectors and we concluded that approximation was tight enough to provide a meaningful bound for parameter selection.

In addition to the theoretical comparisons, we implemented each detection algorithm on the

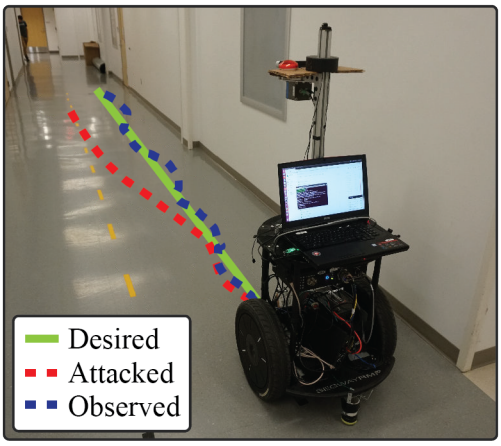


Figure 1.4: Real-world platform for attack detection algorithm evaluation. A Segway robot attempts to follow a simple predefined trajectory at a constant speed (desired). An attacker replaces the truthful measurements with the attacked measurements which are used to generate the observations (observed) resulting in the segway drifting off course (attacked).

Segway robot illustrated in Figure 1.4. The Segway was programmed to perform a path following task with location measurements provided by Google Cartographer [98] using a planar LiDAR and wheel odometry data. Using this platform, each algorithm was evaluated on its ability to detect two different attack models. For the first attack model the attacker adds random noise to the measurement signal, while in the second attack model the attacker replaces the true measurements with fabricated data that is meant to deceive the Segway. From the experiments, we concluded that dynamic watermarking provides more consistent detections than other popular detection algorithms and is able to detect more sophisticated attack models. Video of these experiments and findings can be found at Porter *et al.* [99].

Dynamic Watermarking for LTV Systems

We opened up dynamic watermarking to more complex ITSs with the introduction of time-varying dynamic watermarking in Porter *et al.* [96]. This new algorithm incorporates a carefully designed normalization factor that accommodates the temporal changes in the system. Since the normalization factor is a function of the system dynamics and random noise whose distribution may not be known, a method for approximating the normalization factor for real-world systems was derived. Similar to our prior work, guarantees for detection of replay attacks was provided for the limit-based tests and the effectiveness of the corresponding statistical tests are illustrated using a simulated autonomous vehicle.

Detecting Deception Attacks on Autonomous Vehicles

To further demonstrate the effectiveness of time-varying dynamic watermarking, we implemented our method both on a high fidelity vehicle model in CarSim, and on a $1/10^{\text{th}}$ scale autonomous rover as illustrated in Figure 1.5. In each case, we showed that time-varying dynamic watermarking is able reliably detect replay attacks that would otherwise lead to devastating results. This work also addressed two issues that are not considered in our original time-varying dynamic watermarking: 1) systems that require several time steps for the effect of a given control input to appear in the measurement and 2) the issue of auto-correlation in the measurement residual.

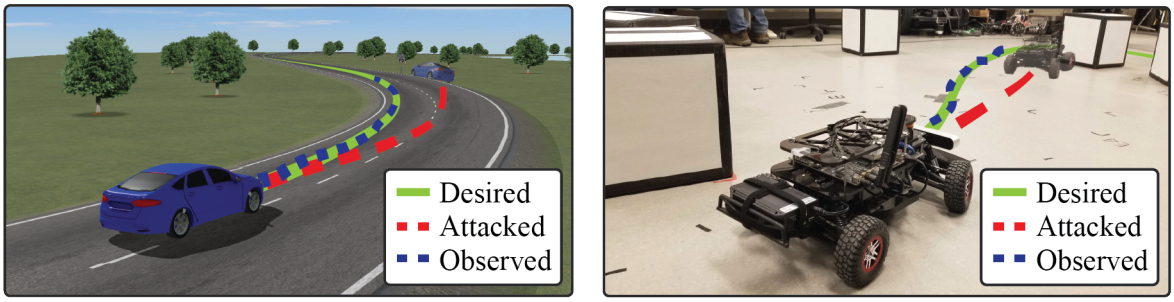


Figure 1.5: A high-fidelity vehicle model in CarSim(Left) and $1/10^{\text{th}}$ scale autonomous rover (Right) are used to evaluate time-varying dynamic watermarking.

For many single agent systems the effect of a control input appears in the measurements in a single time step. However, this may not be true for systems that have inherent delays. To handle such cases, we apply a corresponding delay to the watermark when checking for correlation between the watermark and measurement signal. For this generalization, we prove that the guarantees of detection from Porter *et al.* [96] still hold.

In attack detection literature, it is assumed that a Kalman filter with perfect knowledge of the system dynamics, process noise, and measurement noise is used to observe the system state. As a result, the measurement residual is independent identically distributed (IID) for steady state operation. Of course, the assumptions of perfect knowledge are rarely accurate for real-world systems. Moreover, in each of our publications we have assumed that a Luenberger type observer is used. As a result, the measurement residuals are likely not independent. For an LTI system this results in a constant effect on the detection algorithm which is often negligible. However, for a LTV system the effect can vary with time leading to greater issues. In this work we provided a second normalizing factor to remove the effect of auto-correlation, and derived a method for approximating it using real-world data.

Networked LTV Systems

We have designed an extension for time-varying dynamic watermarking and demonstrated its effectiveness on a simulated platoon of autonomous vehicles in Porter *et al.* [100]. Furthermore, we developed a mitigation strategy for the platoon that enables graceful degradation in the event that an attack is detected.

This work also accommodates drift along the trajectory which has previously not been considered. Namely, for a predefined trajectory, time is mapped to the nominal state of the vehicle/platoon for that time. However, if a vehicle/platoon travels slightly faster or slower at a given point in time it may drift farther ahead or behind in the trajectory. In past works, we have used the controller to correct this drift but in real-world scenarios the controller can operate based its current location along the trajectory instead of the nominal location as a function of time. This work derives a method for approximating the normalizing factors proposed in Porter *et al.* [96] and Porter *et al.* [95] based on location to enable drift along the trajectory.

1.4 Outline of Chapters

The remainder of this dissertation is outlined as follows.

Chapter 2: Preliminaries Mathematical notation used throughout the dissertation and a brief review of topics in statistical analysis.

Chapter 3: Dynamic Watermarking An in depth discussion of the dynamic watermarking theory for both LTI and LTV systems; methods handling persistent disturbances and systems where inputs take more than a single step to be seen in the measurements.

Chapter 4: Tools for Selecting User-Defined Parameter Tools for enabling the tuning of dynamic watermarking specific parameters and comparing dynamic watermarking to other detection algorithms.

Chapter 5: Single Autonomous Vehicle Applications Demonstrations of LTV dynamic watermarking on a single autonomous vehicle with attacked position measurements both in simulation using CarSim and on a real-world system.

Chapter 6: Autonomous Platoon Applications Demonstration of LTV distributed dynamic watermarking on a platoon of autonomous vehicles with attacked V2V communications.

Chapter 7: Conclusions and Future Directions Concluding remarks regarding the results of this dissertation and possible future research directions.

Chapter 2

Preliminaries

This chapter provides the notation used throughout this dissertation in Section 2.1, and some preliminary results and statistical background in Section 2.2. Note that some readers may prefer to skip Section 2.2 then return to view the preliminary results when used in proofs throughout other chapters.

2.1 Notation

This section briefly introduces the notation used in this dissertation.

2.1.1 Vectors and Matrices

The 2-norm of a vector x is denoted $\|x\|$. Similarly, the 2-norm of a matrix X is denoted $\|X\|$. The trace of a matrix X is denoted $\mathrm{tr}(X)$. Zero matrices of dimension $i \times j$ are denoted $0_{i\times j}$, and in the case that $i = j$, the notation is simplified to 0_i. Identity matrices of dimension i are denoted I_i. Block diagonal matrices using blocks $X_1,\ X_2,\ \ldots$ are denoted $\mathrm{blkdiag}(X_1,\ X_2, \ldots)$. The Kronecker product operator is denoted $\otimes$ [101, Definition A.4.1]. The minimum singular value of a matrix $X \in \mathbb{R}^{n\times m}$ is denoted $s_1(X)$.

2.1.2 Probability

The Wishart distribution with scale matrix Σ and i degrees of freedom is denoted $\mathcal{W}(\Sigma, i)$ [101, Section 7.2]. The multivariate gamma function corresponding to dimension i is denoted $\Gamma_{(i)}$ [101, Defintion 7.2.1]. The multivariate Gaussian distribution with mean μ and covariance Σ is denoted $\mathcal{N}(\mu, \Sigma)$. The chi-squared distribution with i degrees of freedom is denoted $\chi^2(i)$. The matrix Gaussian distribution with mean $\mathcal{M}$, and parameters Σ and Ω is denoted $\mathcal{N}(\mathcal{M}, \Sigma, \Omega)$. The expectation of a random variable a is denoted $\mathbb{E}[a]$. The probability of an event E is denoted $\mathbb{P}(E)$. Given a

sequence of random variables $\{a_i\}_{i=1}^{\infty}$, convergence in probability is denoted p-$\lim_{i\to\infty} a_i$ and almost sure convergence is denoted as-$\lim_{i\to\infty} a_i$ [102, Definition 7.2.1].

2.1.3 Set Operations

The cardinality of a set H is denoted card(H). The closed ball of radius ϵ is denoted $\mathcal{B}_\epsilon$. The Minkowsi sum is denoted $\oplus$.

2.2 Statistical Background

First, we provide inequalities for functions of random variables using the following three theorems.

Theorem 1. *[96, Theorem A.1] Let $(a_i)_{i=1}^s$ be a finite set of random variables then*

$$\mathbb{P}\left(\sum_{i=1}^{s} a_i > \epsilon\right) \leq \sum_{i=1}^{s} \mathbb{P}\left(a_i > \frac{\epsilon}{s}\right). \tag{2.1}$$

Proof. **(Theorem 1)** Assume $a_i < \frac{\epsilon}{s}\ \forall i$. This would imply that

$$\sum_{i=1}^{s} a_i < \sum_{i=1}^{s} \frac{\epsilon}{s} = \epsilon, \tag{2.2}$$

$$\left\{\sum_{i=1}^{s} a_i > \epsilon\right\} \subseteq \bigcup_{i=1}^{s} \left\{a_i > \frac{\epsilon}{s}\right\}. \tag{2.3}$$

Furthermore,

$$\mathbb{P}\left(\sum_{i=1}^{s} a_i > \epsilon\right) \leq \mathbb{P}\left(\bigcup_{i=1}^{s} \left\{a_i > \frac{\epsilon}{s}\right\}\right) \leq \sum_{i=1}^{s} \mathbb{P}\left(a_i > \frac{\epsilon}{s}\right). \tag{2.4}$$

where the first inequality comes from the inclusion of the events and the final inequality comes from Boole's Inequality. ■

Theorem 2. *[96, Theorem A.2] Let $(a_i)_{i=1}^s$ be a finite set of random variables then*

$$\mathbb{P}\left(\prod_{i=1}^{s} |a_i| > \epsilon\right) \leq \sum_{i=1}^{s} \mathbb{P}\left(|a_i| > \epsilon^{\frac{1}{s}}\right). \tag{2.5}$$

Proof. **(Theorem 2)** Assume $|a_i| < \epsilon^{\frac{1}{s}}\ \forall i$. This would imply that

$$\prod_{i=1}^{s} |a_i| < \prod_{i=1}^{s} \epsilon^{\frac{1}{s}} = \epsilon. \tag{2.6}$$

The remainder of the proof follows closely to Theorem 1. ■

Theorem 3. *[96, Theorem A.3] Let a and b be random variables then for $\epsilon, \gamma > 0$ we have*

$$\mathbb{P}\left(|ab| < \epsilon\right) \geq \mathbb{P}\left(|a| < \gamma\right) + \mathbb{P}\left(|b| < \tfrac{\epsilon}{\gamma}\right) - 1. \tag{2.7}$$

Proof. **(Theorem 3)** Note that

$$\mathbb{P}(|ab| < \epsilon) \geq \mathbb{P}\left(\{|a| < \gamma\} \cap \left\{|b| < \tfrac{\epsilon}{\gamma}\right\}\right) \tag{2.8}$$

since $|a| < \gamma$ and $|b| < \epsilon/\gamma$ implies $|ab| < \epsilon$. By expanding the RHS of (2.8) using inclusion exclusion and bounding the union term by 1, we get

$$\mathbb{P}(|ab| < \epsilon) \geq \mathbb{P}(|a| < \gamma) + \mathbb{P}\left(|b| < \tfrac{\epsilon}{\gamma}\right) - 1. \tag{2.9}$$

■

It is often helpful to split a probabilistic limit into components of the underlying random variable. While this is not possible for all cases, we provide sufficient conditions here.

Theorem 4. *[96, Theorem A.4] Given sequences of random variables a_i and b_i, and constants a and b, suppose that* $\underset{i\to\infty}{\text{p-lim}}\ a_i + b_i = a + b$ *and* $\underset{i\to\infty}{\text{p-lim}}\ a_i = a$ *then* $\underset{i\to\infty}{\text{p-lim}}\ b_i = b$.

Proof. **(Theorem 4)** Assume $\underset{i\to\infty}{\text{p-lim}}\ a_i + b_i = a + b$ and $\underset{i\to\infty}{\text{p-lim}}\ a_i = a$ hold. Given an $\epsilon > 0$, we have that

$$\mathbb{P}\left(\|b_i - b\| > \epsilon\right) \leq \mathbb{P}\left(\|a_i - a + b_i - b\| > \tfrac{\epsilon}{2}\right) + \mathbb{P}\left(\|a_i - a\| > \tfrac{\epsilon}{2}\right)$$

where the inequality comes from triangle inequality and Theorem 1. Since both terms in this upper bound converge to zero, their sum must as well. ■

Similarly we can combine probabilistic limits as follows.

Corollary 5. *[96, Corollary A.5] Consider sequences of random variables a_i and b_i and constants a and b. If* $\underset{i\to\infty}{\text{p-lim}}\ b_i = b$ *and* $\underset{i\to\infty}{\text{p-lim}}\ a_i = a$ *then* $\underset{i\to\infty}{\text{p-lim}}\ a_i + b_i = a + b$.

Proof. **(Corollary 5)** Let $a_i' = -a_i$, $a' = -a$, $b_i' = a_i + b_i$ and $b' = a + b$. Using Theorem 4 on the new random variables gives us

$$\underset{i\to\infty}{\text{p-lim}}(a_i + b_i) = \underset{i\to\infty}{\text{p-lim}} b_i' = b' = a + b. \tag{2.10}$$

■

Since many of the limits in this paper deal with the average outer product of random vectors, it is important to know how and when these limits converge. The following theorem provides sufficient conditions for convergence.

Theorem 6. *[96, Theorem A.6] Consider the sequences of vectors $(f_i)_{i=1}^{\infty}$ and $(g_i)_{i=1}^{\infty}$ where $f_i \sim \mathcal{N}(0_{s\times 1}, \Sigma_{f,i})$ and $g_i \sim \mathcal{N}(0_{t\times 1}, \Sigma_{g,i})$. Let η and ϵ be scalar values such that $0 < \eta < \infty$ and $\epsilon > 1$. If*

$$\left\|\mathbb{E}\left[f_j f_i^\intercal\right]\right\|, \ \left\|\mathbb{E}\left[g_j g_i^\intercal\right]\right\|, \ \left\|\mathbb{E}\left[f_j g_i^\intercal\right]\right\| < \frac{\eta}{\epsilon^{|i-j|}}, \tag{2.11}$$

$\forall\ i, j \in \mathbb{N}$, then

$$\underset{i\to\infty}{\text{p-lim}} \quad \frac{1}{i}\sum_{j=1}^{i} f_j g_j^\intercal - \mathbb{E}\left[f_j g_j^\intercal\right] = 0_{s\times t}. \tag{2.12}$$

Proof. **(Theorem 6)** For (2.12) to hold, each of the element must also converge to 0 with probability 1. Therefore we will consider an arbitrary element and show it converges using an inequality derived from Chebyshev's inequality. Selecting the element in an arbitrary row m and column n such that $0 \le m \le s$ and $0 \le n \le t$, let

$$h_m^\intercal = \begin{bmatrix} 0_{1\times(m-1)} & 1 & 0_{1\times(s-m)} \end{bmatrix}, \tag{2.13}$$

$$h_n^\intercal = \begin{bmatrix} 0_{1\times(n-1)} & 1 & 0_{1\times(t-n)} \end{bmatrix}, \tag{2.14}$$

then the sum for this single element can be written as

$$\rho_i = \frac{1}{i}\sum_{j=1}^{i} h_m^\intercal f_i g_i^\intercal h_n - h_m^\intercal \mathbb{E}\left[f_j g_j^\intercal\right] h_n. \tag{2.15}$$

In order to use Chebyshev's inequality we must first bound the second moment of ρ_i. We start by

expanding ρ_i^2 using (2.15) and canceling like terms to get

$$\left|\mathbb{E}\left[\rho_i^2\right]\right| = \left|\frac{1}{i^2}\sum_{j=1}^{i}\sum_{k=1}^{i}\mathbb{E}\left[h_m^\intercal f_j g_j^\intercal h_n h_m^\intercal f_k g_k^\intercal h_n\right] - h_m^\intercal \mathbb{E}\left[f_j g_j^\intercal\right] h_n h_m^\intercal \mathbb{E}[f_k g_k^\intercal]\, h_n\right|. \tag{2.16}$$

Expanding the expectation in the first term using [103, Equation 2.3.8] and once again canceling like terms results in

$$\left|\mathbb{E}\left[\rho_i^2\right]\right| = \left|\frac{1}{i^2}\sum_{j=1}^{i}\sum_{k=1}^{i} h_m^\intercal \mathbb{E}\left[f_j g_k^\intercal\right] h_n h_m^\intercal \mathbb{E}\left[f_k g_j^\intercal\right] h_n + h_m^\intercal \mathbb{E}\left[f_j f_k^\intercal\right] h_m h_n^\intercal \mathbb{E}\left[g_j g_k^\intercal\right] h_n\right|. \tag{2.17}$$

Distributing the norm across the addition and multiplication using triangle inequality and the sub-multiplicative property of the 2 norm we then get the upper bound

$$\left|\mathbb{E}\left[\rho_i^2\right]\right| \leq \frac{1}{i^2}\sum_{j=1}^{i}\sum_{k=1}^{i}\|h_m\|^2\|h_n\|^2\left\|\mathbb{E}\left[f_j g_k^\intercal\right]\right\|\ \left\|\mathbb{E}\left[f_k g_j^\intercal\right]\right\| + \|h_m\|^2\|h_n\|^2\left\|\mathbb{E}\left[f_j f_k^\intercal\right]\right\|\ \left\|\mathbb{E}\left[g_j g_k^\intercal\right]\right\|. \tag{2.18}$$

Applying the bounds in (2.11) and the fact that $\|h_m\| = \|h_n\| = 1$ we can further upper bound resulting in

$$\left|\mathbb{E}\left[\rho_i^2\right]\right| \leq \frac{1}{i^2}\sum_{j=1}^{i}\sum_{k=1}^{i}\frac{2\eta^2}{\epsilon^{2|j-k|}} \tag{2.19}$$

Furthermore,

$$\left|\mathbb{E}\left[\rho_i^2\right]\right| \leq \frac{1}{i^2}\sum_{j=1}^{i}\sum_{k=1}^{\infty}\frac{4\eta^2}{\epsilon^{2k}} = \frac{4\eta^2}{i\left(1-\frac{1}{\epsilon^2}\right)}. \tag{2.20}$$

where the inequality comes from the summation in (2.20) containing all of the summands in (2.19) and the fact that all summands are non-negative. Finally, using this bound and applying Chebyshev's Inequality [104, Equation 5.32] we have that, for an arbitrary choice of $\beta > 0$,

$$\mathbb{P}\left(|\rho_i| > \beta\right) \leq \frac{\mathbb{E}\left[\rho_i^2\right]}{\beta^2} = \frac{4\eta^2}{i\beta^2\left(1-\frac{1}{\epsilon^2}\right)}. \tag{2.21}$$

Therefore, ρ_i converges to 0 with probability 1. Since the matrix element was chosen arbitrarily, (2.12) must hold. ■

Using Theorem 6, we can make similar claims for bounded linear transforms of Gaussian sequences.

Corollary 7. *[96, Corollary A.7] Consider a pair of sequences of vectors $(f_i)_{i=1}^{\infty}$ and $(g_i)_{i=1}^{\infty}$ where $f_i \sim \mathcal{N}(0_{s\times 1}, \Sigma_{f,i})$ and $g_i \sim \mathcal{N}(0_{t\times 1}, \Sigma_{g,i})$. Furthermore, consider the sequences of time varying matrices $(T_i)_{i=1}^{\infty}$ and $(U_i)_{i=1}^{\infty}$, where $T_i \in \mathbb{R}^{s'\times s}$ and $U_i \in \mathbb{R}^{t'\times t}$. Assume that*

$$\|T_i\| \leq \eta_T \;\; \textit{and} \;\; \|U_i\| \leq \eta_U. \tag{2.22}$$

Let $\eta, \epsilon \in \mathbb{R}$ such that $0 < \eta < \infty$ and $\epsilon > 1$. If

$$\left\|\mathbb{E}\left[f_j f_i^{\mathsf{T}}\right]\right\|, \;\; \left\|\mathbb{E}\left[g_j g_i^{\mathsf{T}}\right]\right\|, \;\; \left\|\mathbb{E}\left[f_j g_i^{\mathsf{T}}\right]\right\| < \frac{\eta}{\epsilon^{|i-j|}}, \tag{2.23}$$

$\forall\, i, j \in \mathbb{N}$, then

$$\underset{i\to\infty}{\text{p-lim}} \quad \frac{1}{i}\sum_{j=1}^{i} T_j f_j g_j^{\mathsf{T}} U_j^{\mathsf{T}} - T_j \mathbb{E}\left[f_j g_j^{\mathsf{T}}\right] U_j^{\mathsf{T}} = 0_{s'\times t'}. \tag{2.24}$$

Proof. **(Corollary 7)** We prove this result by showing that the bounded linear transform generates new sequences that satisfy the conditions described in Theorem 6. Let

$$f_i' = T_i f_i \;\forall i \;\text{ and } g_i' = U_i g_i \;\forall i \tag{2.25}$$

then $f_i' \sim \mathcal{N}(0_{s'\times 1}, T_i \Sigma_{f,i} T_i^{\mathsf{T}})$ and $g_i' \sim \mathcal{N}(0_{t'\times 1}, U_i \Sigma_{g,i} U_i^{\mathsf{T}})$. Furthermore, we have that

$$\left\|\mathbb{E}\left[f_j' f_i'^{\mathsf{T}}\right]\right\| \leq \|T_j\|\,\|T_i\| \;\left\|\mathbb{E}\left[f_j f_i^{\mathsf{T}}\right]\right\| < \frac{\eta_T^2 \eta}{\epsilon^{|i-j|}} \tag{2.26}$$

where the first inequality comes from the submultiplicative property of the spectral norm and the second from applying (2.23) and (2.22). Similarly,

$$\left\|\mathbb{E}\left[g_j' g_i'^{\mathsf{T}}\right]\right\| < \frac{\eta_U^2 \eta}{\epsilon^{|i-j|}} \quad \text{and} \quad \left\|\mathbb{E}\left[f_j' g_i'^{\mathsf{T}}\right]\right\| < \frac{\eta_U \eta_T \eta}{\epsilon^{|i-j|}}. \tag{2.27}$$

Let $\eta' = \max\{\eta_U^2\eta, \eta_T^2\eta, \eta_U\eta_T\eta\}$ and $\epsilon' = \epsilon$ then

$$\left\|\mathbb{E}\left[f_j' f_i'^{\mathsf{T}}\right]\right\|, \;\; \left\|\mathbb{E}\left[g_j' g_i'^{\mathsf{T}}\right]\right\|, \;\; \left\|\mathbb{E}\left[f_j' g_i'^{\mathsf{T}}\right]\right\| < \frac{\eta'}{\epsilon'^{|i-j|}} \tag{2.28}$$

which satisfies the conditions for using Theorem 6 which implies that

$$\underset{i\to\infty}{\text{p-lim}} \quad \frac{1}{i}\sum_{j=1}^{i} f'_j g'^{\intercal}_j - \mathbb{E}\left[f'_j g'^{\intercal}_j\right] = 0_{s'\times t'}, \tag{2.29}$$

which completes the proof since

$$f'_j g'^{\intercal}_j - \mathbb{E}\left[f'_j g'^{\intercal}_j\right] = T_j f_j g_j^{\intercal} U_j^{\intercal} - T_j \mathbb{E}\left[f_j g_j^{\intercal}\right] U_j^{\intercal}. \tag{2.30}$$

■

To use Theorem 6 and Corollary 7, we provide sufficient conditions for a Gaussian sequence to satisfy conditions (2.11) and (2.23).

Theorem 8. *[96, Theorem A.8] Consider the Gaussian process*

$$a_{i+1} = M_i a_i + b_i \tag{2.31}$$

where $a_0 = 0_{s\times 1}$ and b_i are independent gaussian distributed random vaiables such that $b_i \sim \mathcal{N}(0_{s\times 1}, \Sigma_{b,i})$. If $\exists \epsilon_1, \epsilon_2$ such that $||M_i|| < \epsilon_1 < 1$ and $||\Sigma_{b,i}|| < \epsilon_2 < \infty \; \forall \; i$ then

$$\left\|\mathbb{E}\left[a_j a_i^{\intercal}\right]\right\| < \frac{\eta}{\epsilon^{|i-j|}}, \tag{2.32}$$

where $\eta = \frac{\epsilon_2}{1-\epsilon_1^2}$ and $\epsilon = \frac{1}{\epsilon_1}$.

Proof. **(Theorem 8)** Consider the LHS of (2.32) when $i = j$. We can expand $a_j a_j^T$ using (2.31) iterativley to get

$$\left\|\mathbb{E}\left[a_j a_j^{\intercal}\right]\right\| = \left\|\sum_{i=1}^{j} M_{j-1}\dots M_{j-i+1}\Sigma_{b,j-i}M_{j-i+1}^{\intercal}\dots M_{j-1}^{\intercal}\right\|. \tag{2.33}$$

We upper bound this norm as follows

$$\begin{aligned}\left\|\mathbb{E}\left[a_j a_j^{\intercal}\right]\right\| &\le \sum_{i=1}^{j} ||M_{j-1}||\dots||M_{j-i+1}||\;||\Sigma_{b,j-i}||\;||M_{j-i+1}^{\intercal}||\dots||M_{j-1}^{\intercal}|| < \\ &< \sum_{i=1}^{i} \epsilon_2\epsilon_1^{2(j-1)} \le \frac{\epsilon_2}{1-\epsilon_1^2},\end{aligned} \tag{2.34}$$

where the first inequality comes from applying triangle inequality and the sub-multiplicative property of the spectral norm and the second inequality comes from applying the bounds on $||M_i||$ and

$\|\Sigma_{b,i}\|$ and then bounding the resulting geometric series.

We now focus on (2.32) for when $i \neq j$. Consider the following which has been expanded using (2.31)

$$\left\|\mathbb{E}\left[a_{j+i}a_j^\intercal\right]\right\| = \left\|\mathbb{E}\left[a_j a_{j+i}^\intercal\right]\right\| = \left\|\mathbb{E}\left[a_j(M_{j+i-1}\dots M_j a_j + \sum_{k=1}^{i} M_{j+i-1}\dots M_{j+i-k+1}b_{j+i-k})^\intercal\right]\right\|. \quad (2.35)$$

Since $\mathbb{E}[a_j b_{j+i-k}] = 0 \; \forall \; k \leq i$, this simplifies to

$$\left\|\mathbb{E}\left[a_{j+i}a_j^\intercal\right]\right\| = \left\|\mathbb{E}\left[a_j a_{j+i}^\intercal\right]\right\| = \left\|\mathbb{E}\left[a_j a_j^\intercal\right] M_j^\intercal \dots M_{j+i-1}^\intercal\right\| < \frac{\eta}{\epsilon^i}, \quad (2.36)$$

where the inequality comes from (2.34) and $\|M_i\| < \epsilon_1$. ■

Next, we relate limits of the outer product to those of the inner product in the following Lemma.

Lemma 9. *[96, Lemma A.11] Consider a sequence of random vectors $(b_n)_{n=0}^{\infty}$ such that $b_n \in \mathbb{R}^s$.*

$$\operatorname*{p-lim}_{i\to\infty} \frac{1}{i}\sum_{n=0}^{i-1} b_n b_n^\intercal = 0_s \quad (2.37)$$

if and only if

$$\operatorname*{p-lim}_{i\to\infty} \frac{1}{i}\sum_{n=0}^{i-1} b_n^\intercal b_n = 0. \quad (2.38)$$

Proof. **(Lemma 9)** Assume that (2.38) holds. Note that

$$\left\|\frac{1}{i}\sum_{n=0}^{i-1} b_n b_n^\intercal\right\| \leq \frac{1}{i}\sum_{n=0}^{i-1}\|b_n b_n^\intercal\| = \frac{1}{i}\sum_{n=0}^{i-1} b_n^\intercal b_n. \quad (2.39)$$

where the inequality comes from the triangle inequality and the equality comes from the matrix $b_n b_n^\intercal$ being singular. This implies that

$$\mathbb{P}\left(\left|\frac{1}{i}\sum_{n=0}^{i-1} b_n^\intercal b_n\right| > \epsilon\right) \geq \mathbb{P}\left(\left\|\frac{1}{i}\sum_{n=0}^{i-1} b_n b_n^\intercal\right\| > \epsilon\right). \quad (2.40)$$

Since the LHS of (2.40) converges to zero as $i \to \infty$ as a result of our assumption, the RHS must do so as well which directly implies (2.37) holds.

Now assume that (2.37) holds. Then since

$$\frac{1}{i}\sum_{n=0}^{i-1} b_n^\intercal b_n = \operatorname{tr}\left(\frac{1}{i}\sum_{n=0}^{i-1} b_n b_n^\intercal\right), \tag{2.41}$$

and for the matrix to converge it must also converge element-wise, we have that (2.38) also holds. ■

Next, we show that if conditions such as (2.37) do not hold, linear transforms of the limit also do not converge to zero given the conditions in the following lemma hold.

Lemma 10. *[96, Lemma A.11] Consider a family of matrices $R_n \in \mathbb{R}^{t\times s}$ with full column rank. Assume there exists $\eta \in \mathbb{R}$ such that $0 < \eta \leq \lambda_n$, where λ_n is the smallest eigenvalue of $R_n^T R_n$. Furthermore, consider a sequence of random vectors $f_n \sim \mathcal{N}(0_{s\times 1}, \Sigma_f)$ such that $\Sigma_{f,n}$ is positive semi-definite. If*

$$\sum_{i=1}^{\infty} \left\| \mathbb{E}\left[f_n f_{n+i}^\intercal\right] \right\| < \infty \; \forall n \tag{2.42}$$

$$\operatorname*{p\text{-}lim}_{i\to\infty} \frac{1}{i}\sum_{n=0}^{i-1} f_n f_n^\intercal \neq 0_s, \tag{2.43}$$

then

$$\operatorname*{p\text{-}lim}_{i\to\infty} \frac{1}{i}\sum_{n=0}^{i-1} R_n f_n f_n^\intercal R_n^\intercal \neq 0_t. \tag{2.44}$$

Proof. **(Lemma 10)** Assume that (2.42)-(2.43) holds, but

$$\operatorname*{p\text{-}lim}_{i\to\infty} \frac{1}{i}\sum_{n=0}^{i-1} R_n f_n f_n^\intercal R_n^\intercal = 0_t. \tag{2.45}$$

Applying Lemma 9 we have that

$$\operatorname*{p\text{-}lim}_{i\to\infty} \frac{1}{i}\sum_{n=0}^{i-1} f_n^\intercal R_n^\intercal R_n f_n = 0. \tag{2.46}$$

This implies that

$$\operatorname*{p\text{-}lim}_{i\to\infty} \frac{\eta}{i}\sum_{n=0}^{i-1} f_n^\intercal f_n = 0 \tag{2.47}$$

since $\eta f_n^\intercal f_n \leq \lambda_n f_n^\intercal f_n \leq f_n^\intercal R_n^\intercal R_n f_n$. Since the limit is not affected by the constant η, and using

Lemma 9, this contradicts (2.43). Therefore, (2.44) must hold. ■

Chapter 3

Dynamic Watermarking

This chapter derives dynamic watermarking for both LTI and LTV system models in Section 3.1 and Section 3.2 respectively. Both methods are then extended to distributed control systems in Section 3.3. In Section 3.4, we derive methods for obtaining the system dependent parameter used by dynamic watermarking. Then we conclude the chapter with a discussion in Section 3.5.

Though specifics vary, the derivations in this chapter also share several common variables. For convenience these variables are listed in their simplest form in Table 3.1. For more complex scenarios additional subscripts are used to further specify the particular variable being referenced. The dimensions of the common variables are as follows.

Variable	Description
A	State transition matrix
B	Input matrix
C	Measurement matrix
K	Control gain matrix
L	Observer gain Matrix
x_n	System state vector at step n
$\hat{x}_n$	Observed state vector at step n
δ_n	Observer error at step n
y_n	Measurement vector at step n
w_n	Process noise vector at step n
z_n	Measurement noise vector at step n
e_n	Watermark vector at step n
v_n	Attack vector at step n

Table 3.1: Commonly used variables

$$
\begin{aligned}
x_n, \hat{x}_n, w_n, \delta_n &\in \mathbb{R}^p \\
e_n &\in \mathbb{R}^q \\
y_n, z_n, v_n &\in \mathbb{R}^o
\end{aligned} \tag{3.1}
$$

For simplicity we assume that $x_0, \hat{x}_0 = 0_{p\times 1}$. Furthermore, the process noise, watermark, and measurement noise are assumed to be mutually independent Gaussian random variables such that

$$
\begin{aligned}
w_n &\sim \mathcal{N}(0_{p\times 1}, \Sigma_w), \\
e_n &\sim \mathcal{N}(0_{q\times 1}, \Sigma_e), \\
z_n &\sim \mathcal{N}(0_{o\times 1}, \Sigma_z).
\end{aligned} \tag{3.2}
$$

3.1 LTI Dynamic Watermarking

This section describes LTI dynamic watermarking as described in Hespanhol *et al.* [22] with some modifications resulting from Porter *et al.* [95]. We start by defining the LTI model and necessary assumptions in Subsection 3.1.1. Then, in Subsection 3.1.2 we define the limit-based tests for detecting attacks and their corresponding guarantee of detection. A formal proof for the guarantee of detection and relevant intermediate results are presented in Subsection 3.1.3. Finally, in Subsection 3.1.4, we discuss using the limit-based tests to formulate an implementable real-time statistical test.

3.1.1 LTI Model

Consider an LTI system with state x_n, measurement y_n, process noise w_n, measurement noise z_n, watermark e_n, additive attack v_n, and stabilizing feedback that uses the observed state $\hat{x}$

$$
\begin{aligned}
x_{n+1} &= Ax_n + BK\hat{x}_n + Be_n + w_n, \\
\hat{x}_{n+1} &= (A + BK + LC)\hat{x}_n + Be_n - Ly_n, \\
y_n &= Cx_n + z_n + v_n.
\end{aligned} \tag{$\mathcal{D}1$}
$$

While the process and measurement noise are unknown to the controller, the watermark signal is generated by the controller and is known. The following assumption is made on the controller, observer, and watermark design.

Assumption 11. *Assume the matrices $A + BK$ and $A + LC$ are Shur stable and that Σ_e is full rank.*

Note that to satisfy the assumption on $A + BK$, one could for instance assume that the controllabilty matrix constructed from A and B was full rank. Under that assumption one could design K

using eigenvalue assignment and selecting eigenvalues of magnitude less than one [105, Section 4.4.1]. Moreover, the assumption on $A + LC$ could be satisfied similarly if the observability matrix constructed from A and C was full rank. Since the watermark is user-defined, the remaining assumption can be satisfied by proper selection of Σ_e.

The *measurement residual* for this system takes the form

$$r_n = C\hat{x}_n - y_n. \tag{3.3}$$

When an attack is not present, the distribution of the measurement residuals converge to a zero mean Gaussian distribution with covariance Σ_r where

$$\Sigma_r = \lim_{n\to\infty} \mathbb{E}[(C\hat{x}_n - y_n)(C\hat{x}_n - y_n)^\intercal]. \tag{3.4}$$

Under Assumption 11, this limit is guaranteed to exist and can be found analytically by first considering the observer error $\delta_n = \hat{x}_n - x_n$ which satisfies the dynamics

$$\delta_{n+1} = (A + LC)\delta_n - w_n - Lz_n - Lv_n. \tag{3.5}$$

When no attack is present the observer error follows a zero mean Gaussian distribution with covariance Σ_δ which is the solution to the discrete lyapunov equation

$$\Sigma_\delta = (A + LC)\Sigma_\delta(A + LC)^\intercal + \Sigma_w + L\Sigma_z L^\intercal. \tag{3.6}$$

Then the covariance of the measurement residual can be written as $\Sigma_r = C\Sigma_\delta C^\intercal + \Sigma_z$. To simplify later notation we also define the matrix normalizing factor V as

$$V = \Sigma_r^{-1/2}. \tag{3.7}$$

Remark 12. *Both the covariance of the measurement residual and the matrix normalizing factor can be calculated analytically when all noise parameters are known. However, this is often not the case for real world systems. Instead, the covariance of the measurement residual and the matrix normalizing factor can be estimated as described in Section 3.4.*

Next, consider a generalization of a replay attack satisfying

$$v_n = \alpha(Cx_n + z_n) + C\xi_n + \zeta_n \tag{3.8}$$

$$\xi_{n+1} = (A + BK)\xi_n + \omega_n \tag{3.9}$$

where $\alpha \in \mathbb{R}$ is called the *attack scaling factor*, the false state $\xi_n \in \mathbb{R}^p$ has process noise $\omega_n \in$

$\mathbb{R}^p$ and measurement noise $\zeta_n \in \mathbb{R}^o$ that take the form $\omega_n \sim \mathcal{N}(0_{p\times 1}, \Sigma_\omega)$ and $\zeta_n \sim \mathcal{N}(0_{o\times 1}, \Sigma_\zeta)$, and are mutually independent with each other and with w_n and z_n. Though this attack structure does not include all forms of deception attacks, it does allow an attacker to carry out a variety of documented attacks. For example, selecting $\Sigma_\omega = 0_p$ and an attack scaling factor α of 0 results in independent identically distributed noise being added to the measurement. Moreover, when Σ_ω and Σ_ζ are selected such that the covariance of the measurement residual is unaltered and the attack scaling parameter is -1, this model can approximate a replay attack. While attackers may have the ability to start and stop attacks at will, attacks that are only present for finite time are not guaranteed to be detected. Therefore, when considering asymptotic guarantees of detection, the assumption of persistence is made. To formally describe these persistent attacks, consider the following definition.

Definition 13. *The asymptotic attack power is defined as*

$$\underset{i\to\infty}{\text{as-lim}} \ \tfrac{1}{i} \textstyle\sum_{n=0}^{i-1} v_n^\intercal v_n. \tag{3.10}$$

Under this definition, an attack with non-zero asymptotic power is deemed to be persistent.

3.1.2 Limit-Based Tests

The asymptotic claims of LTI dynamic watermarking consider the following conditions

$$\underset{N\to\infty}{\text{as-lim}} \frac{1}{N} \sum_{n=0}^{N-1} r_n r_n^\intercal = V \tag{LTI.C1}$$

$$\underset{N\to\infty}{\text{as-lim}} \frac{1}{N} \sum_{n=0}^{N-1} r_n e_{n-\rho-1} = 0 \tag{LTI.C2}$$

Here, the (LTI.C1) checks for changes in the covariance of the residual while (LTI.C2) ensures that the attack is uncorrelated with the true measurement and cannot avoid changing this covariance. The delay of the watermark by ρ in (LTI.C2) is chosen to satisfy

$$C(A + BK)^\rho B \neq 0_{o\times q}. \tag{3.11}$$

which ensures that the effect of the watermark is present in the measurement signal. The existence of such a ρ is guaranteed for systems that are both controllable and observable by the following lemma.

Lemma 14. *[22, Corollary 1] Consider an LTI system satisfying* ($\mathcal{D}$1)*. If* (A, B) *is controllable and* (A, C) *is observable then there exists a* $\rho \in \mathbb{N}$ *where* $\rho \leq p - 1$ *satisfying* (3.11)*.*

Proof. Since (A, B) is controllable, we have that $(A + BK, B)$ is controllable meaning the controllability matrix

$$\mathfrak{C} = \begin{bmatrix} B & (A+BK)B & \cdots & (A+BK)^{p-1}B \end{bmatrix} \tag{3.12}$$

has a rank of p. By Sylvester's rank inequality we have that

$$\operatorname{rank}(C\mathfrak{C}) \geq \operatorname{rank}(C) + \operatorname{rank}(\mathfrak{C}) - p = \operatorname{rank}(C). \tag{3.13}$$

Furthermore, (A, C) is observable, and so the observability matrix

$$\mathfrak{O} = \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{p-1} \end{bmatrix} = \operatorname{blkdiag}(C, \ldots, C) \begin{bmatrix} I \\ A \\ \vdots \\ A^{p-1} \end{bmatrix} \tag{3.14}$$

has a rank of p. Again applying Sylvester's rank inequality implies

$$p\operatorname{rank}(C) \geq \operatorname{rank}(\mathfrak{O}) = p, \tag{3.15}$$

or equivalently that $\operatorname{rank}(C) \geq 1$. Combining this with (3.13) gives us $\operatorname{rank}(C\mathfrak{C}) \geq 1$, and so $C\mathfrak{C} \neq 0_{o \times pq}$. This implies the existence of $\rho \leq p - 1$ satisfying (3.11) since $C\mathfrak{C}$ is a block matrix consisting of the blocks $C(A + BK)^m B$ for $m = 0, \ldots, p - 1$. ∎

The asymptotic claims of LTI dynamic watermarking then take the form of the following theorem.

Theorem 15. *[22, Theorem 1] Consider an attacked LTI system satisfying* ($\mathcal{D}$1)*. Let V be as defined in* (3.7) *and ρ be the smallest value for which* (3.11) *holds. If $v_n = 0_{o \times 1}$, for all $n \in \mathbb{N}$, then* (LTI.C1) *and* (LTI.C2) *hold. Furthermore if the attack follows the dynamics in* (3.46) *and has non-zero asymptotic attack power as defined in Definition 13, then* (LTI.C1) *and* (LTI.C2) *cannot both hold.*

To prove Theorem 15, a few intermediate results must first be provided.

3.1.3 Intermediate Results

First consider the combined dynamics that are used to improve notation in the intermediate results.

$$\bar{x}_{n+1} = \bar{A}\bar{x}_n + \bar{B}e_n + \bar{D}w_n + \bar{L}(z_n + v_n) \tag{3.16}$$

where $\bar{x}^\intercal = \begin{bmatrix} x_n^\intercal & \hat{x}_n^\intercal \end{bmatrix}$, $\bar{B}^\intercal = \begin{bmatrix} B^\intercal & B^\intercal \end{bmatrix}$, $\bar{C} = \begin{bmatrix} C & 0_{o\times p} \end{bmatrix}$, $D^\intercal = \begin{bmatrix} I_p & 0_p \end{bmatrix}$, $\bar{L}^\intercal = \begin{bmatrix} 0_{p\times o} & -L^\intercal \end{bmatrix}$, and

$$\bar{A} = \begin{bmatrix} A & BK \\ -LC & A + BK + LC \end{bmatrix} \tag{3.17}$$

Next we provide two lemmas that will aid in the proof of Theorem 15.

Lemma 16. *[22, Lemma 1] We have that*

$$\bar{A}^m\bar{B} = \begin{bmatrix} (A + BK)^m B \\ (A + BK)^m B \end{bmatrix} \tag{3.18}$$

for all $o \geq 0$.

Proof. (Lemma 16) The result holds for $o = 0$ since $\bar{A}^0 = I_{2p}$ and $(A + BK)^0 = I_p$. It remains to show the inductive step. Assume that the result holds for m then

$$\bar{A}^{m+1}\bar{B} = \bar{A}\begin{bmatrix} (A + BK)^m B \\ (A + BK)^m B \end{bmatrix} = \begin{bmatrix} (A + BK)^{m+1} B \\ (A + BK)^{m+1} B \end{bmatrix}. \tag{3.19}$$

Hence the result follows by induction. ∎

Lemma 17. *[22, Proposition 1] Let*

$$A(\alpha) = \bar{A} + \alpha\bar{H} \tag{3.20}$$

with

$$\bar{H} = \begin{bmatrix} 0_p & 0_p \\ -LC & 0_p \end{bmatrix} \tag{3.21}$$

and let ρ *be the smallest value for which* (3.11) *holds. Then* $\bar{A}(\alpha)^m\bar{B} = \bar{A}^m\bar{B}$ *for* $0 \leq m \leq \rho$

Proof. (Lemma 17) If $\rho = 0$, then the result holds trivially. So assume $\rho \geq 1$. We have rhat $\bar{A}(\alpha)^0\bar{B} = \bar{A}\bar{B}$ since $\bar{A}(\alpha)^0 = \bar{A}^0 = I_{2p}$. Now suppose that $\bar{A}(\alpha)^m\bar{B} = \bar{A}^m\bar{B}$ for some $0 \leq m \leq \rho$.

Using Lemma 16 implies that

$$\bar{A}(\alpha)^{m+1}\bar{B} = \bar{A}^{m+1}\bar{B} + \alpha\bar{H}\begin{bmatrix}(A+BK)^m B\\ (A+BK)^m B\end{bmatrix} = \bar{A}^{m+1}\bar{B} + \alpha\begin{bmatrix}0_{p\times q}\\ -LC(A+BK)^m B\end{bmatrix} = \bar{A}^{m+1}\bar{B} \tag{3.22}$$

where the final equality comes from the fact that $LC(A+BK)^m B = 0$ since $m < \rho$. Hence the result follows by induction. ■

Now we proceed to the proof of Theorem 15.

Proof. (Theorem 15) Note that using the attack dynamics (3.8)-(3.9) and (3.20)-(3.21) we can rewrite the combined dynamics in (3.16) as

$$\bar{x}_{n+1} = \bar{A}(\alpha)\bar{x}_b + \bar{B}e_n + \bar{D}w_n + \bar{L}((1+\alpha)z_n + C\xi_n + \zeta_n). \tag{3.23}$$

Next note that a basic calculation gives

$$\begin{aligned}\bar{x}_n = \bar{A}(\alpha)^m \bar{x}_{n-m} + \sum_{m'=0}^{m-1} \bar{A}(\alpha)^{m-m'-1}\Big(&\bar{B}e_{n+m'-m} + \bar{D}w_{n+m'-m} + \\ &+ \bar{L}\left((1+\alpha)z_{n-m'-m} + C\xi_{n+m'-m} + \zeta_{n+m'-m}\right)\Big).\end{aligned} \tag{3.24}$$

If we define $\bar{\bar{C}} = \begin{bmatrix}-C & C\end{bmatrix}$, then $r_n = \bar{\bar{C}}\bar{x}_n - \alpha\bar{C}\bar{x}_n - (1-\alpha)z_n - C\xi_n - \zeta_n$, and so

$$\frac{1}{N}\sum_{n=0}^{N-1}\mathbb{E}\left[(r_n e_{n-\rho-1}^\intercal\right] = (\bar{\bar{C}} - \alpha\bar{C})\bar{A}(\alpha)^{\rho-1}\bar{B}\Sigma_e. \tag{3.25}$$

By Lemma 14 we know that $\rho \le p-1$ so Lemma 17 can be applied to (3.25) to get

$$\frac{1}{N}\sum_{n=0}^{N-1}\mathbb{E}\left[(r_n e_{n-\rho-1}^\intercal]\right] = (\bar{\bar{C}} - \alpha\bar{C})\bar{A}^\rho\bar{B}\Sigma_e = -\alpha\bar{C}\bar{A}^\rho\bar{B}\Sigma_e \tag{3.26}$$

where the second equality holds by Lemma 16 and by the definition of $\bar{\bar{C}}$. Because (LTI.C1) holds, the quantity in (3.26) should equal 0. But since Σ_e is full rank by Assumption 11, Sylvester's rank inequality implies $\bar{C}\bar{A}^\rho\bar{B}\Sigma_e \neq 0_{o\times q}$ since

$$\bar{C}\bar{A}^\rho\bar{B} = \bar{C}\begin{bmatrix}(A+BK)^\rho B\\ (A+BK^\rho B\end{bmatrix} = C(A+BK)^\rho B \neq 0, \tag{3.27}$$

where the first equality holds by Lemma 16 and the second by the definition of $\bar{C}$. Thus we must

have $\alpha = 0$.

Next consider the expression

$$\frac{1}{N}\sum_{n=0}^{N-1} r_n r_n^\intercal = \frac{1}{N}\sum_{n=0}^{N-1} (C\hat{x}_n - (1+\alpha)(Cx_n + z_n) - C\xi_n - \zeta_n)\,(C\hat{x}_n - (1+\alpha)(Cx_n + z_n) - C\xi_n - \zeta_n)^\intercal\,. \tag{3.28}$$

We showed that $\alpha = 0$, and so the expectation of the above expression is

$$C\Sigma_\delta C^\intercal + \Sigma_z + \Sigma_\zeta + \frac{1}{N}\sum_{n=0}^{N-1} \mathbb{E}[C\xi_n\xi_n^\intercal C^\intercal] + \frac{1}{N}\sum_{n=0}^{N-1} \Big(C(A+BK)^{N-1}x_0\Big)\Big(C(A+BK)^{N-1}\xi_0\Big)^\intercal + \\ + \frac{1}{N}\sum_{n=0}^{N-1} \Big(C(A+BK)^{N-1}\xi_0\Big)\Big(C(A+BK)^{N-1}x_0\Big)^\intercal\,. \tag{3.29}$$

Since $(A + BK)$ is Schur stable, the associated property of exponential stability implies

$$\operatorname*{as-lim}_{N\to\infty} \frac{1}{N}\sum_{n=0}^{N-1} C(A+BK)^{N-1}x_0(C(A+BK)^{N-1}\xi_0)^\intercal = 0_r \tag{3.30}$$

by combining Cauchy-Schwartz inequality with the exponential stability. However, from the (LTI.$C2$), the expectation must equal $C\Sigma_\delta C^\intercal + \Sigma_z$ in the limit. Since all terms in the above expectation (3.29) are positive semidefinite or have zero limit this implies that

$$\Sigma_\zeta + \operatorname*{as-lim}_{N\to\infty} \frac{1}{N}\sum_{n=0}^{N-1} \mathbb{E}\left[C\xi_n\xi_n^\intercal C^\intercal\right] = 0_r. \tag{3.31}$$

Finally, consider the expression

$$\frac{1}{N}\sum_{n=0}^{N-1} v_n v_n^\intercal = \frac{1}{N}\sum_{n=0}^{N-1} ((\alpha(Cx_n + z_n) + C\xi_n + \zeta_n)\,((\alpha(Cx_n + z_n) + C\xi_n + \zeta_n)^\intercal\,. \tag{3.32}$$

Since $\alpha = 0$, the expectation of the above expression is

$$\Sigma_\zeta + \frac{1}{N}\sum_{n=0}^{N-1} \mathbb{E}\left[C\xi_n\xi_n^\intercal C^\intercal\right] + \frac{1}{N}\sum_{n=0}^{N-1} \Big(C(A+BK)^{N-1}x_0\Big)\Big(C(A+BK)^{N-1}\xi_0\Big)^\intercal + \\ + \frac{1}{N}\sum_{n=0}^{N-1} \Big(C(A+BK)^{N-1}\xi_0\Big)\Big(C(A+BK)^{N-1}x_0\Big)^\intercal\,. \tag{3.33}$$

Combining (3.30)-(3.33) implies

$$\underset{N\to\infty}{\text{as-lim}}\ \frac{1}{N}\sum_{n=0}^{N-1} v_n v_n^\intercal = 0_r. \tag{3.34}$$

However, $v_n^\intercal v_n$ equals the trace of $v_n v_n^\intercal$. Thus we have

$$\underset{N\to\infty}{\text{as-lim}}\ \frac{1}{N}\sum_{n=0}^{N-1} v_n^\intercal v_n = 0. \tag{3.35}$$

■

3.1.4 Statistical Tests

While Theorem 15 provides guarantees for limit-based tests in infinite time, (LTI.$C1$)-(LTI.$C2$) cannot be used for real-time testing. However, to make these tests implementable in real-time, a statistical test is derived using a sliding window of fixed size ℓ

$$P_n = \begin{bmatrix}\Psi_{n-\ell+1} & \cdots & \Psi_n\end{bmatrix} \qquad \text{where} \qquad \Psi_n = \begin{bmatrix} V(C\hat{x}_n - y_n) \\ \Sigma_e^{-1/2} e_{n-\rho-1}\end{bmatrix}. \tag{3.36}$$

At each step, the combined partial sums in (LTI.$C1$)-(LTI.$C2$) then take the form

$$S_n = P_n P_n^\intercal = \sum_{i=n-\ell+1}^{n} \begin{bmatrix} V(C\hat{x}_i - y_i) \\ \Sigma_e^{-1/2} e_{i-\rho-1}\end{bmatrix} \begin{bmatrix} (C\hat{x}_i - y_i)^\intercal V^\intercal & e_{i-\rho-1}^\intercal \Sigma_e^{-1/2}\end{bmatrix}. \tag{3.37}$$

Under the assumption of no attack, S_n converges asymptotically to the Wishart distribution with scale matrix I_{q+o} and ℓ degrees of freedom as $\ell \to \infty$. Furthermore, for a generalized replay attack of non-zero asymptotic power, Theorem 15 gives us that the scale matrix for S_n is no longer I_{o+q}, since either (LTI.$C1$) or (LTI.$C2$) is not satisfied. Given the sampled matrix S_n, the statistical test then uses the negative log likelihood of the scale matrix

$$\mathcal{L}(S_n) = (q + o + 1 - \ell)\log(|S_n|) + \operatorname{tr}(S_n) + \log\left(2^{(q+o)\ell/2}\Gamma_{(q+o)}\left(\frac{\ell}{2}\right)\right). \tag{3.38}$$

where $\Gamma_{(q+o)}$ is the multivariate gamma function as described in Subsection 2.1.2. Note that the final term in (3.38) is a constant that is only dependent on the dimension of the system and is often omitted to simplify notation in other literature. Negative log likelihood values that exceed a user-defined threshold signal an attack.

Remark 18. *The above derivation of the statistical test is equivalent to that of Hespanhol* et al. *[22]. Note that the distribution of S_n as defined in* (3.36)-(3.37) *converges to a Wishart distribution though it is not necessarily Wishart distributed for a finite value of ℓ. This is due to the measurement residuals being auto-correlated. For an LTI system this results in some consistent approximation error which in many cases may be negligible. However, in Porter* et al. *[95] the definition of S_n in* (3.37) *is modified such that*

$$S_n = P_n G^{-1} P_n^\intercal \qquad \textit{where} \qquad G = \frac{\mathbb{E}[P_n^\intercal P_n]}{q+o}. \tag{3.39}$$

As a result, S_n is in fact Wishart distributed when no attack is present. Estimation of the matrix G is discussed in Section 3.4.

3.2 LTV Dynamic Watermarking

This section describes LTV dynamic watermarking as described in Porter *et al.* [96] with some modifications resulting from Porter *et al.* [95]. We start by defining the LTV model and necessary assumptions in Subsection 3.2.1. Then, in Subsection 3.2.2 we define the limit-based tests for detecting attacks and their corresponding guarantee of detection. A formal proof for the guarantee of detection and relevant intermediate results are presented in Subsection 3.2.3. Note that in this case proofs of the intermediate results have been moved Subsection 3.2.5 to improve readability. Finally, in Subsection 3.2.4, we discuss using the limit-based tests to formulate an implementable real-time statistical test.

3.2.1 LTV Model

Consider an LTV system with state x_n, measurement y_n, process noise w_n, measurement noise z_n, watermark e_n, additive attack v_n, and stabilizing feedback that uses the observed state $\hat{x}$

$$\begin{aligned} x_{n+1} &= A_n x_n + B_n K_n \hat{x}_n + B_n e_n + w_n, \\ \hat{x}_{n+1} &= (A_n + B_n K_n + L_n C_n)\hat{x}_n + B_n e_n - L_n y_n, \\ y_n &= C_n x_n + z_n + v_n. \end{aligned} \tag{$\mathcal{D}2$}$$

While the process and measurement noise are unknown to the controller, the watermark signal is generated by the controller and is known. For simplicity, define

$$\bar{A}_n = (A_n + B_n K_n) \qquad \text{and} \qquad \underline{A}_n = (A_n + L_n C_n). \tag{3.40}$$

Furthermore, let

$$\bar{A}_{(n,m)} = \begin{cases} \bar{A}_n \times \cdots \times \bar{A}_m & n \geq m, \\ I_p & m > n, \end{cases} \quad \text{and} \quad \underline{A}_{(n,m)} = \begin{cases} \underline{A}_n \times \cdots \times \underline{A}_m & n \geq m, \\ I_p & m > n. \end{cases} \tag{3.41}$$

We make the following assumption.

Assumption 19. *The covariances Σ_e, $\Sigma_{w,n}$, and $\Sigma_{z,n}$, of the random variables used in ($\mathcal{D}2$), are full rank. Furthermore, there exists positive constants $\eta_w, \eta_z, \eta_{\bar{A}}, \eta_B, \eta_C \in \mathbb{R}$ such that $\|\Sigma_{w,n}\| < \eta_w$, $\|\Sigma_{z,n}\| < \eta_z$, $\|\bar{A}_n\| < \eta_{\bar{A}} < 1$, $\|B_n\| < \eta_B$, and $\|C_n\| < \eta_C$, for all $n \in \mathbb{N}$.*

The assumption of bounded full rank covariances for the process and measurement noise are satisfied for most systems by modeling error and sensor noise. Furthermore, the input and output matrices are often constrained to be finite by sensor and actuator limits. Note to satisfy the assumption on $\bar{A}_n$, one could for instance assume that the controllabilty matrix constructed from A_n and B_n for all $n \geq 0$ was full rank. Under that assumption one could design K_n using eigenvalue assignment and selecting real distinct eigenvalues that are less than 1 [105, Section 4.4.1]. Since the watermark is user-defined, the remaining assumption can be satisfied by proper selection of Σ_e.

Next, we make the following assumption about the observer.

Assumption 20. *There exists positive constants $\eta_{\underline{A}}$, η_L, η_δ, $\eta_V \in \mathbb{R}$ such that $\|\underline{A}_n\| < \eta_{\underline{A}} < 1$, $\|L_n\| < \eta_L$, $\|\Sigma_{\delta,n}\| < \eta_\delta$, and $\|V_n\| < \eta_V$, for all $n \in \mathbb{N}$.*

Note to satisfy the assumption on $\underline{A}_n$, one could for instance assume that the observability matrix constructed from A_n and C_n for all $n \geq 0$ was full rank. Under that assumption one could design L_n using eigenvalue assignment and selecting real distinct eigenvalues that are less than 1 [105, Section 4.8.1]. Previous assumptions imply the assumptions on L_n, $\Sigma_{\delta,n}$, and V_n are satisfied, but the bounds here simplify notation.

The *measurement residual* for this system takes the form

$$r_n = C_n \hat{x} - y_n \tag{3.42}$$

When no attack is present, the distribution of the measurement residual at a given step n is zero mean Gaussian distributed with time-varying covariance $\Sigma_{r,n}$. This covariance can be found analytically by first considering the observer error

$$\delta_{n+1} = (A_n + L_n C_n)\delta_n - w_n - L_n(z_n + v_n), \tag{3.43}$$

When no attack is present the observer error has time-varying covariance $\Sigma_{\delta,n}$ which can be found

as

$$\Sigma_{\delta,n} = \sum_{i=0}^{n} \underline{A}_{(n-1,n-i+1)}(\Sigma_{w,n-i} + L_{n-i}\Sigma_{z,n-i}L_{n-i}^{\intercal})\underline{A}_{(n-1,n-i+1)}^{\intercal}. \tag{3.44}$$

Then the covariance of the measurement residual can be written as $\Sigma_r = C_n\Sigma_{\delta,n}C_n^{\intercal} + \Sigma_{z,n}$. The *matrix normalization factor* is then defined as

$$V_n = (C_n\Sigma_{\delta,n}C_n^{\intercal} + \Sigma_{z,n})^{-1/2}, \tag{3.45}$$

which exists since $\Sigma_{z,n}$ is full rank. For the LTV system, the matrix normalization factor can be thought of as a time-varying normalization for the covariance of the measurement residual.

Remark 21. *Similar to the LTI case, both the measurement residual and the matrix normalizing factor often must be estimated for real world applications. As such we discuss how to estimate these parameters in Section 3.4.*

Next, we alter the attack defined in (3.8)-(3.9) to create a time-varying equivalent. Consider an attack v_n that satisfies

$$v_n = \alpha(C_n x_n + z_n) + C_n\xi_n + \zeta_n \tag{3.46}$$

$$\xi_{n+1} = \bar{A}_n\xi_n + \omega_n, \tag{3.47}$$

where $\alpha \in \mathbb{R}$ is called the *attack scaling factor*, the *false state* $\xi_n \in \mathbb{R}^p$ has process noise $\omega_n \in \mathbb{R}^p$ and measurement noise $\zeta_n \in \mathbb{R}^o$ that take the form $\omega_n \sim \mathcal{N}(0_{p\times 1}, \Sigma_{\omega,n})$ and $\zeta_n \sim \mathcal{N}(0_{o\times 1}, \Sigma_{\zeta,n})$ and are mutually independent with each other and with w_n and z_n. Similar to the LTI case, when $\Sigma_{\omega,n}$ and $\Sigma_{\zeta,n}$ are selected properly and the attack scaling parameter is -1, this model can approximate a replay attack. While an attacker could choose to allow the noise to have unbounded covariance, the resulting attack would be trivial to detect. Therefore, we make the following assumption about the attack model.

Assumption 22. *When there is an attack, v_n follows the dynamics* (3.46)-(3.47) *with the attack scaling factor remaining constant. Furthermore, there exists positive constants $\eta_\omega, \eta_\eta \in \mathbb{R}$ such that $\|\Sigma_{\omega,n}\| < \eta_\omega$, $\|\Sigma_{\zeta,n}\| < \eta_\zeta$, for all $n \in \mathbb{N}$.*

To make asymptotic guarantees of detection, we also assume the persistence of attacks using the following definition.

Definition 23. *The asymptotic attack power is defined as*

$$\underset{i\to\infty}{\text{p-lim}} \ \tfrac{1}{i} \textstyle\sum_{n=0}^{i-1} v_n^{\intercal} v_n. \tag{3.48}$$

3.2.2 Limit-Based Tests

The asymptotic claims of LTV dynamic watermarking consider the following conditions.

$$\underset{N\to\infty}{\text{p-lim}}\frac{1}{N}\sum_{n=0}^{N-1} V_n r_n r_n^{\mathsf{T}} V_n^{\mathsf{T}} = I \tag{LTV.C1}$$

$$\underset{N\to\infty}{\text{p-lim}}\frac{1}{N}\sum_{n=0}^{N-1} V_n r_n e_{n-\rho-1} = 0 \tag{LTV.C2}$$

Similar to their LTI counterparts (LTV.C1) checks for changes in the covariance of the residual while (LTV.C2) ensures that the attack is uncorrelated with the true measurement and cannot avoid changing this covariance. The delay of the watermark by ρ in (LTV.C2) was initially set to zero in Porter *et al.* [96] with the following assumption

Assumption 24. *[96, Assumption III.2]*

$$\lim_{i\to\infty} \tfrac{1}{i}\textstyle\sum_{n=0}^{i-1} C_n B_{n-1} \neq 0_{o\times q}. \tag{3.49}$$

Here, (3.49) guarantees an asymptotic correlation between the measurement signal y_n and the watermark signal e_{n-1}, which has been delayed by a single time step. This ensures that the watermark has a persistent measurable effect on the measurement signal, which can then be used for validation purposes. This is similar to assuming ρ is equal to 0 for the LTI case.

Remark 25. *Assumption 24 was later relaxed to the following assumption in Porter* et al. *[95].*

Assumption 26. *[95, Assumption IV.2] There exists $\rho \in \mathbb{N}$ such that*

$$\lim_{i\to\infty} \frac{1}{i}\sum_{n=0}^{i-1} C_n \bar{A}_{(n-1,n-\rho+1)} B_{n-\rho} \neq 0_{o\times q}. \tag{3.50}$$

This assumption guarantees an asymptotic correlation between the measurement signal y_n and the delayed watermark $e_{n-\rho-1}$ for an arbitrary non-negative delay *rho*.

The asymptotic claims of LTV dynamic watermarking then take the form of the following theorem.

Theorem 27. *[96, Theorem III.6] Consider an attacked LTV system satisfying the dynamics in* ($\mathcal{D}$2) *and Assumption 24. Let V_n be defined as in* (3.45) *and $\rho = 0$. If $v_n = 0_{o\times 1}$, for all $n \in \mathbb{N}$, then* (LTV.C1) *and* (LTV.C2) *hold. Furthermore, if the attack follows the dynamics in* (3.46)-(3.47) *and has non-zero asymptotic attack power as defined in definition 23, then* (LTV.C1) *and* (LTV.C2) *cannot both be satisfied.*

Remark 28. *Theorem 27 was later relaxed in Porter* et al. *[95] by replacing Assumption 24 with Assumption 26 resulting in the following theorem.*

Theorem 29. *[95, Theorem IV.3] Consider an attacked LTV system satisfying the dynamics in* ($\mathcal{D}$2)*. Let V_n be defined as in* (3.45) *and ρ be the smallest value for which* (3.50) *holds. If $v_n = 0_{o\times 1}$, for all $n \in \mathbb{N}$, then* (LTV.C1) *and* (LTV.C2) *hold. Furthermore, if the attack follows the dynamics in* (3.46)-(3.47) *and has non-zero asymptotic attack power as defined in definition 23, then* (LTV.C1) *and* (LTV.C2) *cannot both be satisfied.*

To prove Theorem 27 and Theorem 29, several intermediate results must first be provided.

3.2.3 Intermediate Results

The proofs of these results have been omitted to improve readability. However, they are available in Subsection 3.2.5. We aim to show that α is equal to 0. Doing so allows us to use the following lemma to simplify the application of the results in Section 2.2.

Lemma 30. *[96, Theorem A.9] Consider an attacked LTV system satisfying the dynamics in* (6.15)-(3.53) *and the attack model in* (3.46)-(3.47)*. Assume the attack scaling factor α is equal to 0. Then $\exists\ \eta > 0$ and $\epsilon > 1$ such that*

$$\left\| \mathbb{E}\left[\begin{bmatrix} x_n \\ \bar{\delta}_n \\ \hat{\delta}_n \\ \xi_n \end{bmatrix} \begin{bmatrix} x_{n+i} \\ \bar{\delta}_{n+i} \\ \hat{\delta}_{n+i} \\ \xi_{n+i} \end{bmatrix}^\top \right] \right\| < \frac{\eta}{\epsilon^i}. \tag{3.51}$$

First, we consider the asymptotic limit (LTV.C2) and show that it implies that the attack scaling factor α is equal to 0. This allows us to assume that α is equal to 0 for the remainder of the intermediate results.

Lemma 31. *[96, Lemma III.7] Consider an attacked LTV system satisfying* ($\mathcal{D}$2) *and Assumption 24 and the attack model satisfying* (3.46)-(3.47)*. Let V_n be as defined in* (3.45) *and $\rho = 0$.* (LTV.C2) *holds if and only if the attack scaling factor α is equal to 0.*

Next we provide an equivalent result after replacing Assumption 24 with the less restrictive Assumption 26.

Theorem 32. *[95, Theorem IV.4] Consider an attacked LTV system satisfying the dynamics in* ($\mathcal{D}$2) *and an attack model satisfying* (3.46)-(3.47)*. Let V_n be as defined in* (3.45)*, and ρ being the smallest value for which* (3.50) *holds.* (LTV.C2) *holds if and only if the attack scaling factor α is equal to 0.*

In a sense, (LTV.$C2$) checks that the attack v_n is uncorrelated with the true measurement, which is true only when the attack scaling factor α is zero.

Assuming α is equal to 0, we show that (LTV.$C1$) is equivalent to another condition that is only dependent on the attack v_n and its contribution to the observer error $\hat{\delta}_n$. Let

$$\bar{\delta}_{n+1} = \underline{A}_n \bar{\delta}_n - w_n - L_n z_n \tag{3.52}$$

$$\hat{\delta}_{n+1} = \underline{A}_n \hat{\delta}_n - L_n v_n \tag{3.53}$$

where $\bar{\delta}_0 = \hat{\delta}_0 = 0_{p\times 1}$. Note that $\delta_n = \bar{\delta}_n + \hat{\delta}_n$ and that when $v_n = 0_{o\times 1},\ \forall n$ we have that $\hat{\delta}_n = 0_{p\times 1},\ \forall n$. Here $\bar{\delta}_n$ can be thought of as the portion of the observer error that results from the original noise of the system, while $\hat{\delta}_n$ is the contribution of the attack to the observer error. defined as Note that $\hat{\delta}_n$ is not computable given the available knowledge of the system, but the provided condition is an amenable surrogate to (LTV.$C1$).

Next, we show that, when α being equal to 0, the full system state satisfies the conditions of Theorem 8.

Lemma 33. *[96, Lemma III.8] Consider an attacked LTV system satisfying* ($\mathcal{D}2$) *and an attack model satisfying* (3.46)-(3.47)*. Let V_n be as defined in* (3.45)*. Assume the attack scaling factor α is equal to 0.* (LTV.$C1$) *holds if and only if*

$$\underset{i\to\infty}{\text{p-lim}}\ \tfrac{1}{i}\sum_{n=0}^{i-1} V_n(C_n\hat{\delta}_n - v_n)(C_n\hat{\delta}_n - v_n)^\intercal V_n^\intercal = 0_r. \tag{3.54}$$

Here (3.54) can be thought of as that contribution of the attack to the value of the LHS of (LTV.$C1$).

For an attack scaling factor α of 0, the attack v_n is only dependent on the random vectors ξ_n and ζ_n. Similar to Lemma 33, these vectors are not computable by the controller, but can be used to connect (LTV.$C1$) to the asymptotic attack power.

Lemma 34. *[96, Lemma III.9] Consider an attacked LTV system satisfying* ($\mathcal{D}2$) *and an attack model satisfying* (3.46)-(3.47)*. Assume that the attack scaling factor α is equal to 0. The asymptotic attack power as defined in* (3.48) *is 0 if and only if*

$$\underset{i\to\infty}{\text{p-lim}}\tfrac{1}{i}\sum_{n=0}^{i-1} \zeta_n\zeta_n^\intercal = 0_r, \tag{3.55}$$

and

$$\underset{i\to\infty}{\text{p-lim}}\tfrac{1}{i}\sum_{n=0}^{i-1} C_n\xi_n\xi_n^\intercal C_n^\intercal = 0_r. \tag{3.56}$$

Each of the prior equations can be thought of as the contribution of each random vector to the

asymptotic attack power.

Next, we start to complete the connection between (LTV.$C1$) and zero asymptotic attack power by proving (3.54) implies (3.55). Furthermore, we prove a related result that makes it simpler to prove that (3.54) implies (3.56).

Lemma 35. *[96, Lemma III.10] Consider an attacked LTV system satisfying* ($\mathcal{D}2$) *and an attack model satisfying* (3.46)-(3.47)*. Let V_n be as defined in* (3.45)*. Assume the attack scaling factor α is equal to 0. If* (3.54) *holds, then* (3.55) *holds as well and*

$$\operatorname*{p-lim}_{i\to\infty} \tfrac{1}{i} \textstyle\sum_{n=0}^{i-1} (C_n\hat{\delta}_n - C_n\xi_n)(C_n\hat{\delta}_n - C_n\xi_n)^\intercal = 0_r. \tag{3.57}$$

Since ζ_n adds additional noise to the measurement signal, the link between (3.54) and (3.55) is clear. In particular, (3.57) is constructed by removing ζ_n's effect from (3.54).

Next we claim that (3.54) implies (3.56) to complete the relation between (LTV.$C1$) and the asymptotic attack power.

Lemma 36. *[96, Lemma III.11] Consider an attacked LTV system satisfying* ($\mathcal{D}2$) *and an attack model satisfying* (3.46)-(3.47)*. Let V_n be as defined in* (3.45)*. Assume the attack scaling factor α is equal to 0. If* (3.54) *holds then* (3.56) *holds as well.*

The proof of Lemma 36 makes use of Lemma 35 and instead shows that (3.57) implies (3.56). Despite the removal of ζ_n in (3.57), the correlation between $\hat{\delta}_n$ and ξ_n introduces a potential complication. To address this challenge, we prove the contrapositive statement. Assuming that (3.56) does not hold, we make the following assertion.

Lemma 37. *[96, Lemma III.12] Consider an attacked LTV system satisfying* ($\mathcal{D}2$) *and an attack model satisfying* (3.46)-(3.47)*. Let V_n be as defined in* (3.45)*. Assume the attack scaling factor α is equal to 0. If* (3.56) *does not hold then there exists $m \in \mathbb{N}$ for which*

$$\operatorname*{p-lim}_{i\to\infty} \frac{1}{i} \sum_{n=0}^{i-1} \left(C_n \sum_{j=1}^{m_n} \bar{A}_{(n-1,n-j+1)}\omega_{n-j} \right) \left(C_n \sum_{j=1}^{m_n} \bar{A}_{(n-1,n-j+1)}\omega_{n-j} \right)^\intercal \neq 0_r. \tag{3.58}$$

where $m_n = \min\{n, m\}$. Furthermore, there exists an $m' \in \mathbb{N}$ such that $m' \leq m$ and

$$\operatorname*{p-lim}_{i\to\infty} \frac{1}{i} \sum_{n=0}^{i-1} C_n \bar{A}_{(n-1,n-j+1)}\omega_{n-j}\omega_{n-j}^\intercal \bar{A}_{(n-1,n-j+1)}^\intercal C_n^\intercal \neq 0_r \tag{3.59}$$

for $j = m'$ but not for $j < m'$.

Here (3.56) is expanded into a summation over a triangular array. Splitting ξ_n in (3.57), allows us to modify the cross terms and complete the proof.

Having proven several intermediate results, we are now able to formally prove Theorem 27 and Theorem 29.

Proof. **(Theorem 27)** When no attack is present, (LTV.$C2$) holds using Lemma 31 since the attack scaling factor α is equal to 0. Furthermore, (LTV.$C1$) holds since the observer error $\delta = \overline{\delta}$.

Now assume that an attack of non-zero asymptotic power is present and consider the following cases.

Case 1 ($\alpha \neq 0$): Using Lemma 31, (LTV.$C2$) does not hold.

Case 2 ($\alpha = 0$): Note, (LTV.$C1$) implies zero asymptotic attack power as follows.

$$\text{(LTV.}C1\text{)} \underset{\text{Lemma 33}}{\Longleftrightarrow} (3.54) \begin{array}{l} \underset{\text{Lemma 35}}{\Longrightarrow} (3.55) \\ \underset{\text{Lemma 36}}{\Longrightarrow} (3.56) \end{array} \underset{\text{Lemma 34}}{\Longleftrightarrow} \begin{pmatrix} \text{zero asymptotic} \\ \text{attack power} \end{pmatrix}$$

Under our assumption of non-zero asymptotic power, the contrapositive implies that (LTV.$C1$) does not hold. ■

Proof. **(Theorem 29)** When no attack is present, (LTV.$C2$) holds using Theorem 32 since the attack scaling factor α is equal to 0. Furthermore, (LTV.$C1$) holds since the observer error $\delta = \overline{\delta}$.

Now assume that an attack of non-zero asymptotic power is present and consider the following cases.

Case 1 ($\alpha \neq 0$): Using Theorem 32, (LTV.$C2$) does not hold.

Case 2 ($\alpha = 0$): Note, (LTV.$C1$) implies zero asymptotic attack power using the same argument as the proof for Theorem 27 above. Under our assumption of non-zero asymptotic power, the contrapositive implies that (LTV.$C1$) does not hold. ■

3.2.4 Statistical Tests

While Theorem 27 and Theorem 29 provides guarantees for limit-based tests in infinite time, (LTV.$C1$)-(LTV.$C2$) cannot be used for real-time testing. However, to make these tests implementable in real-time, a statistical test is derived using a sliding window of fixed size ℓ

$$P_n = \begin{bmatrix} \Psi_{n-\ell+1} & \cdots & \Psi_n \end{bmatrix} \qquad \text{where} \qquad \Psi_n = \begin{bmatrix} V_n(C_n\hat{x}_n - y_n) \\ \Sigma_e^{-1/2} e_{n-\rho-1} \end{bmatrix}. \tag{3.60}$$

At each step, the combined partial sums in (LTV.$C1$)-(LTV.$C2$) then take the form

$$S_n = P_n P_n^\intercal = \sum_{i=n-\ell+1}^{n} \begin{bmatrix} V_n(C_n\hat{x}_i - y_i) \\ \Sigma_e^{-1/2} e_{i-\rho-1} \end{bmatrix} \begin{bmatrix} (C_n\hat{x}_i - y_i)^\intercal V_n^\intercal & e_{i-\rho-1}^\intercal \Sigma_e^{-1/2} \end{bmatrix}. \tag{3.61}$$

Under the assumption of no attack, S_n converges asymptotically to the Wishart distribution with scale matrix I_{q+o} and ℓ degrees of freedom as $\ell \to \infty$. Furthermore, for a generalized replay attack of non-zero asymptotic power, Theorem 27 and Theorem 29 give us that the scale matrix for S_n is no longer I_{o+q}, since either (LTV.$C1$) or (LTV.$C2$) is not satisfied. Given the sampled matrix S_n, the statistical test then uses the negative log likelihood of the scale matrix

$$\mathcal{L}(S_n) = (q + o + 1 - \ell)\log(|S_n|) + \operatorname{tr}(S_n) + \log\left(2^{(q+o)\ell/2}\Gamma_{(q+o)}\left(\frac{\ell}{2}\right)\right). \tag{3.62}$$

Negative log likelihood values that exceed a user-defined threshold signal an attack.

Remark 38. *The above derivation of the statistical test is equivalent to that of Porter* et al. *[96]. Note that the distribution of S_n as defined in* (3.36)-(3.37) *converges to a Wishart distribution though it is not necessarily Wishart distributed for a finite value of ℓ. This is due to the measurement residuals being auto-correlated. For an LTV system this results in some consistent approximation error which in many cases may be negligible. However, in Porter* et al. *[95] the definition of S_n in* (3.61) *is modified such that*

$$S_n = P_n G_n^{-1} P_n^\intercal \qquad \textit{where} \qquad G_n = \frac{\mathbb{E}[P_n^\intercal P_n]}{q + o}. \tag{3.63}$$

As a result, S_n is in fact Wishart distributed when no attack is present. Estimation of the matrix G is discussed in Section 3.4.

3.2.5 Proofs

Proof. **(Lemma 30)** We prove this result using Theorem 8. First note that using (6.15)-(3.53), (3.46)-(3.47), and assuming $\alpha = 0$ we can write

$$a_{n+1} = M_n a_n + b_n \tag{3.64}$$

where $a_n = \begin{bmatrix} x_n^\intercal & \bar{\delta}_n^\intercal & \hat{\delta}_n^\intercal & \xi_n^\intercal \end{bmatrix}^\intercal$,

$$M_n = \begin{bmatrix} \bar{A}_n & B_n K_n & B_n K_n & 0_p \\ 0_p & \underline{A}_n & 0_p & 0_p \\ 0_p & 0_p & \underline{A}_n & -L_n C_n \\ 0_p & 0_p & 0_p & \bar{A}_n \end{bmatrix}, \tag{3.65}$$

and $b_n = T_n \begin{bmatrix} e_n^\intercal & w_n^\intercal & z_n^\intercal & \zeta_n^\intercal & \omega_n^\intercal \end{bmatrix}^\intercal$ with

$$T_n = \begin{bmatrix} B_n & I_p & 0_{p\times r} & 0_{p\times r} & 0_p \\ 0_{p\times q} & -I_p & -L_n & 0_{p\times r} & 0_p \\ 0_{p\times q} & 0_p & 0_{p\times r} & -L_n & 0_p \\ 0_{p\times q} & 0_p & 0_{p\times r} & 0_{p\times r} & I_p \end{bmatrix}. \tag{3.66}$$

Let $\epsilon_1 = \max\{\eta_{A1}, \eta_{A2}\}$ then $\|M_n\| < \epsilon_1 < 1$ since the eigenvalues of upper block diagonal matrices are the set of eigenvalues of the block elements on the diagonal and $\|\bar{A}_n\| < \eta_{A1} < 1$ and $\|\underline{A}_n\| < \eta_{A2} < 1$. Furthermore, $b_n \sim \mathcal{N}(0, \Sigma_{b,n})$ where

$$\Sigma_{b,n} = T_n \operatorname{blkdiag}(\Sigma_e, \Sigma_{w,n}, \Sigma_{z,n}, \Sigma_{\zeta,n}, \Sigma_{\omega,n}) T_n^\intercal. \tag{3.67}$$

Since $B_n, L_n, \Sigma_e, \Sigma_{w,n}, \Sigma_{z,n}, \Sigma_{\zeta,n}$, and $\Sigma_{\omega,n}$ are all bounded we have that $\|\Sigma_{b,n}\| < \epsilon_2$ for some $0 \leq \epsilon_2 < \infty$. Using Theorem 8 completes the proof. ■

Proof. **(Theorem 32)** Assume that α is equal to 0. Then (LTV.C2) holds by the same reasoning as for the proof of the original theorem [96, Theorem III.7].

Now assume that (LTV.C2) holds. Rearranging (LTV.C2) using ($\mathcal{D}$2), (3.43), and (3.46) results in

$$\begin{aligned} \operatorname*{p\text{-}lim}_{i\to\infty} \frac{1}{i} \sum_{n=0}^{i-1} V_n (C_n \hat{x}_n - y_n) e_{n-\rho}^\intercal = \\ \operatorname*{p\text{-}lim}_{i\to\infty} \frac{1}{i} \sum_{n=0}^{i-1} V_n (C_n \delta_n - (1+\alpha) z_n - \alpha C_n x_n - C_n \xi_n - \zeta_n) e_{n-\rho}^\intercal. \end{aligned} \tag{3.68}$$

Note that

$$\operatorname*{p\text{-}lim}_{i\to\infty} \frac{1}{i} \sum_{n=0}^{i-1} V_n (-(1+\alpha) z_n - C_n \xi_n - \zeta_n) e_{n-\rho}^\intercal = 0_{o\times q} \tag{3.69}$$

by Corollary 7 since z_n, ζ_n, ξ_n and $e_{n-\rho}$ are mutually independent and satisfy the necessary auto-correlation bound. Then by Theorem 4 we can cancel these terms resulting in

$$\operatorname*{p-lim}_{i\to\infty} \frac{1}{i}\sum_{n=0}^{i-1} V_n(C_n\hat{x}_n - y_n)e_{n-\rho}^{\intercal} = \operatorname*{p-lim}_{i\to\infty} \frac{1}{i}\sum_{n=0}^{i-1} V_n(C_n\delta_n - \alpha C_n x_n)e_{n-\rho}^{T}. \tag{3.70}$$

Expanding x_n, δ_n in (3.70) by $\kappa+1$ steps using ($\mathcal{D}2$) and (3.43) then collecting all terms that do not depend on $e_{n-\rho-1}$ and denoting them a_n results in

$$\begin{aligned}\operatorname*{p-lim}_{i\to\infty} \frac{1}{i}\sum_{n=0}^{i-1} V_n(C_n\hat{x}_n - y_n)e_{n-\rho}^{\intercal} &= \\ = \operatorname*{p-lim}_{i\to\infty} \frac{1}{i}\sum_{n=0}^{i-1} V_n\Big(a_n - \alpha\sum_{j=0}^{\kappa-1} M_{j,n}C_{n-j}\bar{A}_{(n-j-1,n-\rho+1)}B_{n-\rho}e_{n-\rho}\Big)e_{n-\rho}^{\intercal}.\end{aligned} \tag{3.71}$$

where $M_{j,n} \in \mathbb{R}^{o\times o}$ is a bounded linear transform due to the dynamics being bounded and κ being finite. Moreover, $M_{0,n} = I_r$ and due to our choice of κ terms for $j > 0$ can be cancelled by Theorem 4 since they converge to $0_{q,o}$ by Corollary 7 resulting in

$$\begin{aligned}\operatorname*{p-lim}_{i\to\infty} \frac{1}{i}\sum_{n=0}^{i-1} V_n(C_n\hat{x}_n - y_n)e_{n-\rho}^{\intercal} &= \\ = \operatorname*{p-lim}_{i\to\infty} \frac{1}{i}\sum_{n=0}^{i-1} V_n\Big(a_n - \alpha C_n\bar{A}_{(n-1,n-\rho+1)}B_{n-\rho}e_{n-\rho}\Big)e_{n-\rho}^{\intercal}.\end{aligned} \tag{3.72}$$

Then by Corollary 7 we have that

$$\operatorname*{p-lim}_{i\to\infty} \frac{1}{i}\sum_{n=0}^{i-1} -\alpha V_n C_n\bar{A}_{(n-1,n-\rho+1)}B_{n-\rho}(e_{n-\rho}e_{n-\rho}^{\intercal} - \Sigma_e) = 0_{q\times o}. \tag{3.73}$$

Therefore by Theorem 4 we have

$$\begin{aligned}\operatorname*{p-lim}_{i\to\infty} \frac{1}{i}\sum_{n=0}^{i-1} V_n(C_n\hat{x}_n - y_n)e_{n-\rho}^{\intercal} &= \\ = \operatorname*{p-lim}_{i\to\infty} \frac{1}{i}\sum_{n=0}^{i-1} V_n a_n e_{n-\rho}^{\intercal} - \alpha V_n C_n\bar{A}_{(n-1,n-\rho+1)}B_{n-\rho}\Sigma_e.\end{aligned} \tag{3.74}$$

Note, that all elements of

$$V_n a_n e_{n-\rho}^{\intercal} \tag{3.75}$$

are distributed symmetrically about 0 for all $n \in \mathbb{N}$ since a_n is a zero mean Gaussian random vector. Consider an element of (3.74) for which the corresponding element in

$$\frac{1}{i}\sum_{n=0}^{i-1} V_n C_n \bar{A}_{(n-1,n-\rho+1)} B_{n-\rho} \Sigma_e \tag{3.76}$$

does not converge. For each i, the probability that the matrix element in (3.74) is farther away from 0 than the corresponding element in (3.76) is at least 0.5. Therefore the element cannot converge in probability to 0 completing the proof. ■

Proof. **(Lemma 31)** Assume that α is equal to 0. Rearranging the LHS of (LTV.$C2$) using ($\mathcal{D}2$), (3.43), and (3.46) results in

$$\operatorname*{p-lim}_{i\to\infty} \frac{1}{i}\sum_{n=0}^{i-1} V_n(C_n\hat{x}_n - y_n)e_{n-1}^\intercal = \operatorname*{p-lim}_{i\to\infty} \frac{1}{i}\sum_{n=0}^{i-1} V_n(C_n\delta_n - z_n - C_n\xi_n - \zeta_n)e_{n-1}^\intercal. \tag{3.77}$$

Corollary 5 says that to show that the RHS of (3.77) converges in probability to $0_{o\times q}$, it is sufficient to show that each term in the sum converges in probability to $0_{o\times q}$. Note that

$$\operatorname*{p-lim}_{i\to\infty} \frac{1}{i}\sum_{n=0}^{i-1} V_n(C_n\delta_n - C_n\xi_n)e_{n-1}^\intercal = 0_{o\times q} \tag{3.78}$$

by Corollary 7 since e_{n-1} is independent identically distributed with bounded covariance, and $V_n(C_n\delta_n - C_n\xi_n)$ is a bounded linear transform of a random vector that satisfies the necessary auto correlation bound as a result of Theorem 30. Similarly,

$$\operatorname*{p-lim}_{i\to\infty} \frac{1}{i}\sum_{n=0}^{i-1} V_n(-z_n - \zeta_n)e_{n-1}^\intercal = 0_{o\times q} \tag{3.79}$$

by Corollary 7 since z_n, ζ_n, and e_{n-1} are mutually independent identically distributed with bounded covariances. Therefore $\alpha = 0$ implies (LTV.$C2$) holds.

Now assume that (LTV.$C2$) holds. Rearranging (LTV.$C2$) using ($\mathcal{D}2$), (3.43), and (3.46) results in

$$\begin{aligned}\operatorname*{p-lim}_{i\to\infty} \frac{1}{i}\sum_{n=0}^{i-1} V_n(C_n\hat{x}_n - y_n)e_{n-1}^\intercal &= \\ = \operatorname*{p-lim}_{i\to\infty} \frac{1}{i}\sum_{n=0}^{i-1} V_n(C_n\delta_n - (1+\alpha)z_n - \alpha C_n x_n - C_n\xi_n - \zeta_n)e_{n-1}^\intercal.\end{aligned} \tag{3.80}$$

Now since (3.79) holds by the same argument as before, we can use Theorem 4 to cancel these terms resulting in

$$\operatorname*{p-lim}_{i\to\infty} \frac{1}{i}\sum_{n=0}^{i-1} V_n(C_n\hat{x}_n - y_n)e_{n-1}^{\intercal} = \operatorname*{p-lim}_{i\to\infty} \frac{1}{i}\sum_{n=0}^{i-1} V_n(C_n\delta_n - \alpha C_n x_n)e_{n-1}^{T}. \tag{3.81}$$

Expanding x_n in (3.81) by one step using $(\mathcal{D}2)$ then results in

$$\begin{aligned}&\operatorname*{p-lim}_{i\to\infty} \frac{1}{i}\sum_{n=0}^{i-1} V_n(C_n\hat{x}_n - y_n)e_{n-1}^{\intercal} = \\ &= \operatorname*{p-lim}_{i\to\infty} \frac{1}{i}\sum_{n=0}^{i-1} V_n(C_n\delta_n - \alpha C_n(A_{n-1}x_{n-1} + B_{n-1}K_{n-1}\hat{x}_{n-1} + B_{n-1}e_{n-1} + w_{n-1}))e_{n-1}^{\intercal}.\end{aligned} \tag{3.82}$$

Using Corollary 7 we have that

$$\operatorname*{p-lim}_{i\to\infty} \frac{1}{i}\sum_{n=0}^{i-1} -\alpha V_n C_n B_{n-1}(e_{n-1}e_{n-1}^{\intercal} - \Sigma_e) = 0_{q\times o}. \tag{3.83}$$

Therefore by Theorem 4 we have

$$\begin{aligned}&\operatorname*{p-lim}_{i\to\infty} \frac{1}{i}\sum_{n=0}^{i-1} V_n(C_n\hat{x}_n - y_n)e_{n-1}^{\intercal} = \\ &= \operatorname*{p-lim}_{i\to\infty} \frac{1}{i}\sum_{n=0}^{i-1} V_n(C_n\delta_n - \alpha C_n(A_{n-1}x_{n-1} + B_{n-1}K_{n-1}\hat{x}_{n-1} + w_{n-1}))e_{n-1}^{\intercal} + \alpha V_n C_n B_{n-1}\Sigma_e.\end{aligned} \tag{3.84}$$

Note that all elements of

$$V_n(C_n\delta_n - \alpha C_n(A_{n-1}x_{n-1} + B_{n-1}K_{n-1}\hat{x}_{n-1} + w_{n-1}))e_{n-1}^{\intercal} \tag{3.85}$$

are distributed symmetrically about 0 for all $n \in \mathbb{N}$. Consider an element of (3.84) for which the corresponding element in

$$\frac{1}{i}\sum_{n=0}^{i-1} V_n C_n B_{n-1}\Sigma_e \tag{3.86}$$

does not converge. For each i, the probability that the matrix element in (3.84) is farther away from 0 than the corresponding element in (3.86) is at least 0.5. Therefore the element cannot converge in probability to 0 completing the proof. ■

Proof. **(Lemma 33)** Expanding (LTV.$C1$) using ($\mathcal{D}2$) and (3.43)-(3.53) gives us

$$\begin{aligned}\underset{i\to\infty}{\text{p-lim}}\ \frac{1}{i}\sum_{n=0}^{i-1} V_n(C_n\hat{x}_n - y_n)(C_n\hat{x}_n - y_n)^\intercal V_n^\intercal = \\ = \underset{i\to\infty}{\text{p-lim}}\ \frac{1}{i}\sum_{n=0}^{i-1} V_n(C_n\bar{\delta}_n - z_n)(C_n\bar{\delta}_n - z_n)^\intercal V_n^\intercal + V_n(C_n\bar{\delta}_n - z_n)(C_n\hat{\delta}_n - v_n)^\intercal V_n^\intercal + \\ + V_n(C_n\hat{\delta}_n - v_n)(C_n\bar{\delta}_n - z_n)^\intercal V_n^\intercal + V_n(C_n\hat{\delta}_n - v_n)(C_n\hat{\delta}_n - v_n)^\intercal V_n^\intercal. \end{aligned} \tag{3.87}$$

By Corollary 7 and Theorem 30,

$$\underset{i\to\infty}{\text{p-lim}}\ \frac{1}{i}\sum_{n=0}^{i-1} V_n(C_n\bar{\delta}_n - z_n)(C_n\bar{\delta}_n - z_n)^\intercal V_n^\intercal = I_r, \tag{3.88}$$

$$\underset{i\to\infty}{\text{p-lim}}\ \frac{1}{i}\sum_{n=0}^{i-1} V_n(C_n\bar{\delta}_n - z_n)(C_n\hat{\delta} - v_n)^\intercal V_n^\intercal = 0_o \tag{3.89}$$

since, by the definition of V_n in (3.45), the expectation for each summand in (3.88) is I_r, and $V_n(C_n\bar{\delta}_n - z_n)$ is uncorrelated with $V_n(C_n\hat{\delta} - v_n)$. First, assume that (LTV.$C1$) holds. By Theorem 4, it follows from (3.87)-(3.89) that (3.54) must hold. Next, assume that (3.54) holds. By Corollary 5, it follows from (3.87)-(3.89) that (LTV.$C1$) holds. ∎

Proof. **(Lemma 34)** Assume that $\alpha = 0$. Using Lemma 9, the asymptotic attack power is 0 if and only if

$$\underset{i\to\infty}{\text{p-lim}}\frac{1}{i}\sum_{n=0}^{i-1} v_n v_n^\intercal = 0_o. \tag{3.90}$$

Expanding the LHS of (3.90) using (3.46)-(3.47) we get an equivalent condition.

$$\underset{i\to\infty}{\text{p-lim}}\frac{1}{i}\sum_{n=0}^{i-1} C_n\xi_n\xi_n^\intercal C_n^\intercal + C_n\xi_n\zeta_n^\intercal + (C_n\xi_n\zeta_n^\intercal)^\intercal + \zeta_n\zeta_n^\intercal = 0_r \tag{3.91}$$

Since ξ_n and ζ_n are uncorrelated, from Theorem 30 and Corollary 7 we have

$$\underset{i\to\infty}{\text{p-lim}}\frac{1}{i}\sum_{n=0}^{i-1} C_n\xi_n\zeta_n^\intercal = 0_r. \tag{3.92}$$

First, assume that (3.55) and (3.56) hold. By Corollary 5 we have that (3.91) must hold since, when separated, the limit for each term converges to 0_r. Next, assume that (3.91) holds. By Theorem 4

we can rewrite (3.91) as

$$\underset{i\to\infty}{\text{p-lim}}\frac{1}{i}\sum_{n=0}^{i-1}\zeta_n\zeta_n^\intercal + C_n\xi_n\xi_n^\intercal C_n^\intercal = 0_r, \tag{3.93}$$

since (3.92) holds. Note, both terms are positive-semidefinite matrices. Therefore, for an arbitrary $\epsilon > 0$ we have that

$$\mathbb{P}\left(\left\|\frac{1}{i}\sum_{n=0}^{i-1}\zeta_n\zeta_n^\intercal\right\| > \epsilon\right) \leq \mathbb{P}\left(\left\|\frac{1}{i}\sum_{n=0}^{i-1}\zeta_n\zeta_n^\intercal + C_n\xi_n\xi_n^\intercal C_n^\intercal\right\| > \epsilon\right) \tag{3.94}$$

Furthermore, (3.93) implies

$$\lim_{i\to\infty}\mathbb{P}\left(\left\|\frac{1}{i}\sum_{n=0}^{i-1}\zeta_n\zeta_n^\intercal + C_n\xi_n\xi_n^\intercal C_n^\intercal\right\| > \epsilon\right) = 0_r,\ \forall\epsilon > 0 \tag{3.95}$$

Then, by (3.94) and (3.95)

$$\lim_{i\to\infty}\mathbb{P}\left(\left\|\frac{1}{i}\sum_{n=0}^{i-1}\zeta_n\zeta_n^\intercal\right\| > \epsilon\right) = 0_r,\ \forall\epsilon > 0. \tag{3.96}$$

Therefore, (3.55) must hold. Applying Theorem 4 to (3.93) using (3.55) implies (3.56) must also hold. ■

Proof. **(Lemma 35)** Assume that (3.54) holds. Expanding the LHS of (3.54) using (3.46) we get

$$\begin{aligned}\underset{i\to\infty}{\text{p-lim}}\frac{1}{i}\sum_{n=0}^{i-1} & V_n(C_n\hat{\delta}_n - C_n\xi_n)(C_n\hat{\delta}_n - C_n\xi_n)^\intercal V_n^\intercal + V_n(C_n\hat{\delta}_n - C_n\xi_n)\zeta_n^\intercal V_n^\intercal + \\ & + (V_n(C_n\hat{\delta}_n - C_n\xi_n)\zeta_n^\intercal V_n^\intercal)^\intercal + V_n\zeta_n\zeta_n^\intercal V_n^\intercal = 0_r.\end{aligned} \tag{3.97}$$

Using Corollary 7 and Theorem 30 we have

$$\underset{i\to\infty}{\text{p-lim}}\frac{1}{i}\sum_{n=0}^{i-1} V_n(C_n\hat{\delta}_n - C_n\xi_n)\zeta_n^\intercal V_n^\intercal = 0_r. \tag{3.98}$$

Therefore, by applying Theorem 4 to (3.97) we have

$$\underset{i\to\infty}{\text{p-lim}}\frac{1}{i}\sum_{n=0}^{i-1} V_n(C_n\hat{\delta}_n - C_n\xi_n)(C_n\hat{\delta}_n - C_n\xi_n)^\intercal V_n^\intercal + V_n\zeta_n\zeta_n^\intercal V_n^\intercal = 0_r. \tag{3.99}$$

Note, both terms are positive-semidefinite matrices. Using the same method used on (3.93), we then have

$$\underset{i\to\infty}{\text{p-lim}}\frac{1}{i}\sum_{n=0}^{i-1} V_n\zeta_n\zeta_n^\intercal V_n^\intercal = 0_r, \tag{3.100}$$

$$\underset{i\to\infty}{\text{p-lim}}\frac{1}{i}\sum_{n=0}^{i-1} V_n(C_n\hat{\delta}_n - C_n\xi_n)(C_n\hat{\delta}_n - C_n\xi_n)^\intercal V_n^\intercal = 0_r. \tag{3.101}$$

We complete the proof using Lemma 10 but we must first lower bound the eigenvalues of $V_n^\intercal V_n$. Let λ_n denote the smallest eignenvalue of $V_n^\intercal V_n$, then λ_n is lower bounded since

$$\lambda_n = \frac{1}{\|(V_n^\intercal V_n)^{-1}\|} = \frac{1}{\|C_n\Sigma_{\delta,n}C_n^\intercal + \Sigma_{z,n}\|} \geq \frac{1}{\eta_C^2\eta_\delta + \eta_z} > 0. \tag{3.102}$$

If we assume that (3.55) does not hold then applying Lemma 10 contradicts (3.100). Therefore (3.55) must hold. Similarly, assuming that (3.57) does not hold would result in a contradiction with (3.101). Therefore (3.57) must also hold. ■

Proof. **(Lemma 37)** First, we prove the existence of m. Assume that (3.56) does not hold. Expanding the LHS of (3.56) using (3.47) results in

$$\underset{i\to\infty}{\text{p-lim}}\frac{1}{i}\sum_{n=0}^{i-1}\left(\sum_{j=1}^{n} C_n\bar{A}_{(n-1,n-j+1)}\omega_{n-j}\right)\left(\sum_{j=1}^{n} C_n\bar{A}_{(n-1,n-j+1)}\omega_{n-j}\right)^\intercal \neq 0_r. \tag{3.103}$$

Then, using Lemma 9 we have that

$$\underset{i\to\infty}{\text{p-lim}}\frac{1}{i}\sum_{n=0}^{i-1}\left\|\sum_{j=1}^{n} C_n\bar{A}_{(n-1,n-j+1)}\omega_{n-j}\right\|^2 \neq 0. \tag{3.104}$$

Since (3.56) does not hold there exists $\epsilon, \tau > 0$ such that

$$\mathbb{P}\left(\frac{1}{i}\sum_{n=0}^{i-1}\left\|\sum_{j=1}^{n} C_n\bar{A}_{(n-1,n-j+1)}\omega_{n-j}\right\|^2 > \epsilon\right) > \tau \tag{3.105}$$

for infinitely many i. We prove that there exists an m such that for each i that (3.105) holds we have

$$\mathbb{P}\left(\frac{1}{i}\sum_{n=0}^{i-1}\left\|\sum_{j=1}^{m_n} C_n\bar{A}_{(n-1,n-j+1)}\omega_{n-j}\right\|^2 > \frac{\epsilon}{6}\right) > \frac{\tau}{4} \tag{3.106}$$

which is equivalent to (3.58) as a result of Lemma 9. To make statements on the truncated sum, we start by finding the relationship between the probability in the LHS of (3.105) and the probability in the LHS of (3.106). For each i such that (3.105) holds, we apply triangle inequality to get

$$\tau < \mathbb{P}\left(\frac{1}{i}\sum_{n=0}^{i-1}\left(\left\|\sum_{j=1}^{m_n} C_n\bar{A}_{(n-1,n-j+1)}\omega_{n-j}\right\| + \left\|\sum_{j=m_n+1}^{n} C_n\bar{A}_{(n-1,n-j+1)}\omega_{n-j}\right\|\right)^2 > \epsilon\right). \tag{3.107}$$

Further expanding and applying Theorem 1 result in

$$\begin{aligned}\tau < \mathbb{P}\left(\frac{1}{i}\sum_{n=0}^{i-1}\left\|\sum_{j=1}^{m_n} C_n\bar{A}_{(n-1,n-j+1)}\omega_{n-j}\right\|^2 > \frac{\epsilon}{3}\right) + \mathbb{P}\left(\frac{1}{i}\sum_{n=0}^{i-1}\left\|\sum_{j=m_n+1}^{n} C_n\bar{A}_{(n-1,n-j+1)}\omega_{n-j}\right\|^2 > \frac{\epsilon}{3}\right)\\ + \mathbb{P}\left(\frac{2}{i}\sum_{n=0}^{i-1}\left\|\sum_{j=1}^{m_n} C_n\bar{A}_{(n-1,n-j+1)}\omega_{n-j}\right\|\left\|\sum_{j=m_n+1}^{n} C_n\bar{A}_{(n-1,n-j+1)}\omega_{n-j}\right\| > \frac{\epsilon}{3}\right).\end{aligned} \tag{3.108}$$

Focusing on the center term in the RHS of (3.108), we can write

$$\begin{aligned}&\mathbb{P}\left(\frac{2}{i}\sum_{n=0}^{i-1}\left\|\sum_{j=1}^{m_n} C_n\bar{A}_{(n-1,n-j+1)}\omega_{n-j}\right\|\left\|\sum_{j=m_n+1}^{n} C_n\bar{A}_{(n-1,n-j+1)}\omega_{n-j}\right\| > \frac{\epsilon}{3}\right) \leq\\ &\leq \mathbb{P}\left(\sqrt{\frac{2}{i}\sum_{n=0}^{i-1}\left\|\sum_{j=1}^{m_n} C_n\bar{A}_{(n-1,n-j+1)}\omega_{n-j}\right\|^2}\sqrt{\frac{2}{i}\sum_{n=0}^{i-1}\left\|\sum_{j=m_n+1}^{n} C_n\bar{A}_{(n-1,n-j+1)}\omega_{n-j}\right\|^2} > \frac{\epsilon}{3}\right) \leq\\ &\leq \mathbb{P}\left(\frac{2}{i}\sum_{n=0}^{i-1}\left\|\sum_{j=1}^{m_n} C_n\bar{A}_{(n-1,n-j+1)}\omega_{n-j}\right\|^2 > \frac{\epsilon}{3}\right) + \mathbb{P}\left(\frac{2}{i}\sum_{n=0}^{i-1}\left\|\sum_{j=m_n+1}^{n} C_n\bar{A}_{(n-1,n-j+1)}\omega_{n-j}\right\|^2 > \frac{\epsilon}{3}\right),\end{aligned} \tag{3.109}$$

where the first inequality comes from applying the Cauchy Schwarz Inequality and the second inequality comes from applying Theorem 2. Then since

$$\mathbb{P}\left(\frac{1}{i}\sum_{n=0}^{i-1}\left\|\sum_{j=1}^{m_n} C_n\bar{A}_{(n-1,n-j+1)}\omega_{n-j}\right\|^2 > \frac{\epsilon}{3}\right) \leq \mathbb{P}\left(\frac{2}{i}\sum_{n=0}^{i-1}\left\|\sum_{j=1}^{m_n} C_n\bar{A}_{(n-1,n-j+1)}\omega_{n-j}\right\|^2 > \frac{\epsilon}{3}\right), \tag{3.110}$$

$$\mathbb{P}\left(\frac{1}{i}\sum_{n=0}^{i-1}\left\|\sum_{j=m_n+1}^{n} C_n\bar{A}_{(n-1,n-j+1)}\omega_{n-j}\right\|^2 > \frac{\epsilon}{3}\right) \leq \mathbb{P}\left(\frac{2}{i}\sum_{n=0}^{i-1}\left\|\sum_{j=m_n+1}^{n} C_n\bar{A}_{(n-1,n-j+1)}\omega_{n-j}\right\|^2 > \frac{\epsilon}{3}\right), \tag{3.111}$$

we can combine (3.108) with (3.109)-(3.111) to obtain

$$\tau < 2\mathbb{P}\left(\frac{1}{i}\sum_{n=0}^{i-1}\left\|\sum_{j=1}^{m_n} C_n\bar{A}_{(n-1,n-j+1)}\omega_{n-j}\right\|^2 > \frac{\epsilon}{6}\right) + 2\mathbb{P}\left(\frac{1}{i}\sum_{n=0}^{i-1}\left\|\sum_{j=m_n+1}^{n} C_n\bar{A}_{(n-1,n-j+1)}\omega_{n-j}\right\|^2 > \frac{\epsilon}{6}\right). \tag{3.112}$$

If we can upper bound the second term in the RHS or (3.112) by $\frac{\tau}{2}$ the first term must be lower bounded by $\frac{\tau}{2}$ completing the proof. To provide this bound we make use of Markov's Inequality. To this end, we first bound the expectation

$$\begin{aligned}
&\mathbb{E}\left(\frac{1}{i}\sum_{n=0}^{i-1}\left\|\sum_{j=m_n+1}^{n} C_n\bar{A}_{(n-1,n-j+1)}\omega_{n-j}\right\|^2\right) = \\
&= \mathbb{E}\Big(\frac{1}{i}\sum_{n=0}^{i-1}\sum_{j=m_n+1}^{n}\left(C_n\bar{A}_{(n-1,n-j+1)}\omega_{n-j}\right)^{\mathsf{T}}\left(C_n\bar{A}_{(n-1,n-j+1)}\omega_{n-j}\right)\Big) \leq \\
&\leq \mathbb{E}\Big(\frac{1}{i}\sum_{n=0}^{i-1}\sum_{j=m+1}^{\infty}\left(C_{n+j}\bar{A}_{(n+j-1,n+1)}\omega_n\right)^{\mathsf{T}}\left(C_{n+j}\bar{A}_{(n+j-1,n+1)}\omega_n\right)\Big) \leq \\
&\leq \frac{1}{i}\sum_{n=0}^{i-1}\sum_{j=m+1}^{\infty} p\eta_C^2\eta_{A1}^{2(j-1)}\eta_\omega^2 = \frac{p\eta_C^2\eta_{A1}^{2m}\eta_\omega^2}{1-\eta_{A1}^2},
\end{aligned} \tag{3.113}$$

where the the first equality comes from expanding the norm and ignoring uncorrelated terms, the first inequality comes from rearranging the summation and allowing the second summation to go to infinity, the second inequality comes from distributing the expectation and upper bounding each element, and the final equality comes from evaluating the summations. Since $\eta_{A1} < 1$, we can choose m sufficiently large such that

$$\frac{p\eta_C^2\eta_{A1}^{2m}\eta_\omega^2}{1-\eta_{A1}^2} < \frac{\tau\epsilon}{24}. \tag{3.114}$$

Using Markov's inequality [104, Equation 5.31] we have that

$$\mathbb{P}\left(\frac{1}{i}\sum_{n=0}^{i-1}\left\|\sum_{j=m_n+1}^{n} C_n\bar{A}_{(n-1,n-j+1)}\omega_{n-j}\right\|^2 > \frac{\epsilon}{6}\right) \leq \frac{6p\eta_C^2\eta_{A1}^{2m}\eta_\omega^2}{(1-\eta_{A1}^2)\epsilon} < \frac{\tau}{4} \tag{3.115}$$

which completes the proof for the existence of m.

Next, to prove the existence of m' we expand (3.58)

$$\operatorname*{p-lim}_{i\to\infty}\frac{1}{i}\sum_{n=0}^{i-1} C_n\sum_{j=1}^{m_n}\sum_{k=1}^{m_n}\bar{A}_{(n-1,n-j+1)}\omega_{n-j}\omega_{n-k}^{\mathsf{T}}\bar{A}_{(n-1,n-k+1)}^{\mathsf{T}}C_n^{\mathsf{T}} \neq 0. \tag{3.116}$$

Considering the summands where $j \neq k$ we have that

$$\operatorname*{p-lim}_{i\to\infty}\frac{1}{i}\sum_{n=0}^{i-1} C_n\bar{A}_{(n-1,n-j+1)}\omega_{n-j}\omega_{n-k}^{\mathsf{T}}\bar{A}_{(n-1,n-k+1)}^{\mathsf{T}}C_n^{\mathsf{T}} = 0 \tag{3.117}$$

by Theorem 6 since ω_n is independent and the dynamics are bounded and stable. If we further assume that there does not exist an m' for which (3.59) holds then by Theorem 1 we have that (3.58) does not hold which is a contradiction. Therefore, the set of integers less than or equal to m for which (3.59) holds, is a non-empty finite set. The smallest element of this set then satisfies the conditions for m'. ■

Proof. **(Lemma 36)** WLOG, in this proof, we allow summations to reference variables with negative index by assuming these values to be 0_r to ease notation. Assume that (3.54) holds but (3.56) does not. Since (3.56) does not hold, m' be chosen such that it satisfies the description in Lemma 37. From Lemma 35 we have that (3.54) implies (3.57). Expanding the LHS of (3.57) using (3.47) gives us

$$\begin{aligned}
&\operatorname*{p-lim}_{i\to\infty}\frac{1}{i}\sum_{n=0}^{i-1}(C_n\hat{\delta}_n - C_n\xi_n)(C_n\hat{\delta}_n - C_n\xi_n)^\intercal = \\
&= \operatorname*{p-lim}_{i\to\infty}\frac{1}{i}\sum_{n=0}^{i-1}\left(C_n\left(\hat{\delta}_n - \sum_{j=1}^{n}\bar{A}_{(n-1,n-j+1)}\omega_{n-j}\right)\left(\hat{\delta}_n - \sum_{k=1}^{n}\bar{A}_{(n-1,n-k+1)}\omega_{n-k}\right)^\intercal C_n^\intercal\right) = 0_r. \qquad (3.118)
\end{aligned}$$

By separating the index m' we can write

$$\begin{aligned}
&\operatorname*{p-lim}_{i\to\infty}\frac{1}{i}\sum_{n=0}^{i-1}(C_n\hat{\delta}_n - C_n\xi_n)(C_n\hat{\delta}_n - C_n\xi_n)^\intercal = \\
&= \operatorname*{p-lim}_{i\to\infty}\frac{1}{i}\sum_{n=0}^{i-1} C_n\left(\hat{\delta}_n - \sum_{\substack{1\le j\le n\\ j\neq m'}}\bar{A}_{(n-1,n-j+1)}\omega_{n-j}\right)\left(\hat{\delta}_n - \sum_{\substack{0\le k\le n\\ k\neq m'}}\bar{A}_{(n-1,n-k+1)}\omega_{n-k}\right)^\intercal C_n^\intercal + \\
&- C_n\left(\hat{\delta}_n - \sum_{\substack{1\le j\le n\\ j\neq m'}}\bar{A}_{(n-1,n-j+1)}\omega_{n-j}\right)\omega_{n-m'}^\intercal\bar{A}_{(n-1,n-m'+1)}^\intercal C_n^\intercal + \\
&- C_n\bar{A}_{(n-1,n-m'+1)}\omega_{n-m'}\left(\hat{\delta}_n - \sum_{\substack{0\le k\le n\\ k\neq m'}}\bar{A}_{(n-1,n-k+1)}\omega_{n-k}\right)^\intercal C_n^\intercal + \\
&+ C_n\bar{A}_{(n-1,n-m'+1)}\omega_{n-m'}\omega_{n-m'}^\intercal\bar{A}_{(n-1,n-m'+1)}^\intercal C_n^\intercal = 0_r. \qquad (3.119)
\end{aligned}$$

For now suppose that

$$\operatorname*{p-lim}_{i\to\infty}\frac{1}{i}\sum_{n=0}^{i-1}-C_n\left(\hat{\delta}_n-\sum_{\substack{1\le j\le n\\ j\neq m'}}\bar{A}_{(n-1,n-j+1)}\omega_{n-j}\right)\omega_{n-m'}^{\intercal}\bar{A}_{(n-1,n-m'+1)}^{\intercal}C_n^{\intercal}=0_r. \tag{3.120}$$

Then by Theorem 4 we have that

$$\begin{aligned}
&\operatorname*{p-lim}_{i\to\infty}\frac{1}{i}\sum_{n=0}^{i-1}(C_n\hat{\delta}_n-C_n\xi_n)(C_n\hat{\delta}_n-C_n\xi_n)^{\intercal}=\\
&=\operatorname*{p-lim}_{i\to\infty}\frac{1}{i}\sum_{n=0}^{i-1}C_n\left(\hat{\delta}_n-\sum_{\substack{1\le j\le n\\ j\neq m'}}\bar{A}_{(n-1,n-j+1)}\omega_{n-j}\right)\left(\hat{\delta}_n-\sum_{\substack{0\le k\le n\\ k\neq m'}}\bar{A}_{(n-1,n-k+1)}\omega_{n-k}\right)^{\intercal}C_n^{\intercal}+\\
&\quad+C_n\bar{A}_{(n-1,n-m'+1)}\omega_{n-m'}\omega_{n-m'}^{\intercal}\bar{A}_{(n-1,n-m'+1)}^{\intercal}C_n^{\intercal}=0_r.
\end{aligned} \tag{3.121}$$

Furthermore, by our choice of m' we have that

$$\operatorname*{p-lim}_{i\to\infty}\frac{1}{i}\sum_{n=0}^{i-1}C_n\bar{A}_{(n-1,n-m'+1)}\omega_{n-m'}\omega_{n-m'}^{\intercal}\bar{A}_{(n-1,n-m'+1)}^{\intercal}C_n^{\intercal}\neq 0_r, \tag{3.122}$$

and since the terms are all positive-semidefinite matrices

$$\begin{aligned}
&\mathbb{P}\left(\left\|\frac{1}{i}\sum_{n=0}^{i-1}C_n\bar{A}_{(n-1,n-m'+1)}\omega_{n-m'}\omega_{n-k}^{\intercal}\bar{A}_{(n-1,n-k+1)}^{\intercal}C_n^{\intercal}\right\|>\epsilon\right)\le\\
&\le\mathbb{P}\left(\left\|\frac{1}{i}\sum_{n=0}^{i-1}C_n\left(\hat{\delta}_n-\sum_{\substack{1\le j\le n\\ j\neq m'}}\bar{A}_{(n-1,n-j+1)}\omega_{n-j}\right)\left(\hat{\delta}_n-\sum_{\substack{0\le k\le n\\ k\neq m'}}\bar{A}_{(n-1,n-k+1)}\omega_{n-k}\right)^{\intercal}C_n^{\intercal}+\right.\right.\\
&\qquad\left.\left.+C_n\bar{A}_{(n-1,n-m'+1)}\omega_{n-m'}\omega_{n-m'}^{\intercal}\bar{A}_{(n-1,n-m'+1)}^{\intercal}C_n^{\intercal}\right\|>\epsilon\right).
\end{aligned} \tag{3.123}$$

This implies that (3.119) cannot hold which contradicts (3.54). Therefore (3.56) must hold since otherwise there exists an m' satisfying Lemma 10.

To complete the proof, we now show that (3.120) indeed holds. By Corollary 5 this is equivalent

to proving

$$\underset{i\to\infty}{\text{p-lim}}\frac{1}{i}\sum_{n=0}^{i-1} C_n \sum_{\substack{1\le j\le n\\ j\neq m'}} \bar{A}_{(n-1,n-j+1)}\omega_{n-j}\omega_{n-m'}^{\intercal}\bar{A}_{(n-1,n-m'+1)}^{\intercal}C_n^{\intercal} = 0_r, \tag{3.124}$$

$$\underset{i\to\infty}{\text{p-lim}}\frac{1}{i}\sum_{n=0}^{i-1} -C_n\hat{\delta}_n\omega_{n-m'}^{\intercal}\bar{A}_{(n-1,n-m'+1)}^{\intercal}C_n^{\intercal} = 0_r. \tag{3.125}$$

Note, (3.124) holds by Corollary 7 since all ω_n are mutually independent, $\|C_n\bar{A}_{(n-1,n-m'+1)}\| \le \|C_n\| < \eta_C$, and

$$\begin{aligned}
&\left\|\mathbb{E}\left[\left(\sum_{\substack{1\le j\le n\\ j\neq m'}} \bar{A}_{(n-1,n-j+1)}\omega_{n-j}\right)\left(\sum_{\substack{1\le k\le n+i\\ k\neq m'}} \bar{A}_{(n+i-1,n+i-k+1)}\omega_{n+i-k}\right)^{\intercal}\right]\right\| = \\
&= \left\|\sum_{\substack{1\le j\le n\\ j\neq m'}} \bar{A}_{(n-1,n-j+1)}\Sigma_{\omega,n-j}\bar{A}_{(n+i-1,n-j+1)}^{\intercal}\right\| \le \sum_{j=1}^{\infty}\eta_{A1}^{2j-4+i}\eta_\omega = \frac{\eta_{A2}^{i-2}\eta_\omega}{1-\eta_{A1}^2}.
\end{aligned} \tag{3.126}$$

Here, the equality comes from evaluating the expectation, and the inequality comes from distributing the norm using the triangle inequality and the subadditivity of the spectral norm, bounding the individual terms, and allowing the summation to extend to infinity. Expanding the LHS of (3.125) using (3.53) gives us

$$\begin{aligned}
&\underset{i\to\infty}{\text{p-lim}}\frac{1}{i}\sum_{n=0}^{i-1} -C_n\hat{\delta}_n\omega_{n-m'}^{\intercal}\bar{A}_{(n-1,n-m'+1)}^{\intercal}C_n^{\intercal} = \\
&= \underset{i\to\infty}{\text{p-lim}}\frac{1}{i}\sum_{n=0}^{i-1} C_n\Big(\sum_{j=0}^{n-1}\underline{A}_{(n-1,j+1)}L_j\zeta_j + \sum_{k=1}^{n}\underline{A}_{(n-1,n-k+1)}L_{n-k}C_{n-k}\times \\
&\qquad \times \sum_{\ell=k+1}^{n}\bar{A}_{(n-k-1,n-\ell+1)}\omega_{n-\ell}\Big)\omega_{n-m'}^{\intercal}\bar{A}_{(n-1,n-m'+1)}^{\intercal}C_n^{\intercal} = 0_r.
\end{aligned} \tag{3.127}$$

To prove that (3.125) holds, we use Corollary 5 on (3.127) and show that each term converges to 0_r. Note, by Theorem 6,

$$\underset{i\to\infty}{\text{p-lim}}\frac{1}{i}\sum_{n=0}^{i-1} C_n\Big(\sum_{j=0}^{n-1}\underline{A}_{(n-1,j+1)}L_j\zeta_j\Big)\omega_{n-m'}^{\intercal}\bar{A}_{(n-1,n-m'+1)}^{\intercal}C_n^{\intercal} = 0_r, \tag{3.128}$$

since $\|C_n\bar{A}_{(n-1,n-m'+1)}\| \leq \eta_C$, ζ_n and ω_n are mutually independent, and

$$\left\|\mathbb{E}\left[\sum_{j=0}^{n-1}\sum_{k=0}^{n+i-1}\underline{A}_{(n-1,j+1)}L_j\zeta_j\zeta_k^\intercal L_k^\intercal\underline{A}_{(n+i-1,k+1)}^\intercal\right]\right\| =$$
$$= \left\|\sum_{j=0}^{n-1}\underline{A}_{(n-1,j+1)}L_j\Sigma_{\zeta_j}L_j^\intercal\underline{A}_{(n+i-1,j+1)}^\intercal\right\| \leq \textstyle\sum_{j=0}^{n-1}\eta_{A2}^{2(n-1-j)}\eta_{A2}^i\eta_L^2\eta_\zeta \leq \frac{\eta_{A2}^i\eta_L^2\eta_\zeta}{1-\eta_{A2}^2}. \tag{3.129}$$

Furthermore, considering the portion of $\hat{\delta}_n$ not dependent on $\omega_{n-m'}$, by Theorem 6,

$$\operatorname*{p-lim}_{i\to\infty}\frac{1}{i}\sum_{n=0}^{i-1}C_n\sum_{j=1}^{n}\underline{A}_{(n-1,n-j+1)}L_{n-j}C_{n-j}\sum_{\substack{k=j+1\\k\neq m'}}^{n}\bar{A}_{(n-j-1,n-k+1)}\omega_{n-k}\omega_{n-m'}^\intercal\bar{A}_{(n-1,n-m'+1)}^\intercal C_n^\intercal = 0_r, \tag{3.130}$$

since ω_n are independent, $\|C_n\bar{A}_{(n-1,n-m'+1)}\| \leq \eta_C$, and

$$\left\|\mathbb{E}\left[\left(C_n\sum_{j=1}^{n}\underline{A}_{(n-1,n-j+1)}L_{n-j}C_{n-j}\sum_{\substack{k=j+1\\k\neq m'}}^{n}\bar{A}_{(n-j-1,n-k+1)}\omega_{n-k}\right)\times\right.\right.$$
$$\left.\left.\times\left(C_{n+i}\sum_{j=1}^{n+i}\underline{A}_{(n+i-1,n+i-j+1)}L_{n+i-j}C_{n+i-j}\sum_{\substack{k=j+1\\k\neq m'}}^{n+i}\bar{A}_{(n+i-j-1,n+i-k+1)}\omega_{n+i-k}\right)^\intercal\right]\right\| =$$
$$= \left\|\sum_{j=1}^{n}\sum_{\ell=1}^{n+i}C_n\underline{A}_{(n-1,n-j+1)}L_{n-j}C_{n-j}\sum_{\substack{k=\max\{j+1,\ell+1\}\\k\neq m'}}^{n}\bar{A}_{(n-j-1,n-k+1)}\Sigma_{\omega,n-k}\times\right.$$
$$\left.\times\bar{A}_{(n+i-\ell-1,n-k+1)}^\intercal C_{n+i-\ell}^\intercal L_{n+i-\ell}^\intercal\underline{A}_{(n+i-1,n+i-\ell+1)}^\intercal C_{n+i}^\intercal\right\| \leq$$
$$\leq \sum_{j=1}^{n}\sum_{\ell=1}^{n}\eta_C^4\eta_L^2\eta_A^{\ell+j-2}\sum_{\substack{k=\max\{j+1,\ell+1\}\\k\neq m'}}^{n}\eta_A^{2k-j-\ell-2+i}\eta_\omega \leq$$
$$\leq \eta_A^{i-4}\eta_C^4\eta_L^2\eta_\omega 2\sum_{j=1}^{\infty}\sum_{\ell=j}^{\infty}\sum_{k=\ell+1}^{\infty}\eta^{2k} = \frac{2\eta_A^i\eta_C^4\eta_L^2\eta_\omega}{(1-\eta_A^2)^3}, \tag{3.131}$$

where the first equality comes from evaluating the expectation, the first inequality comes from distributing the norm using the triangle inequality and the submultiplicative property of the spectral norm and then using the individual upper bounds, the second inequality comes from rearranging the sum and allowing the index to go to infinity, and the final equality comes from evaluating the

geometric series. Now if

$$\operatorname*{p-lim}_{i\to\infty}\frac{1}{i}\sum_{n=0}^{i-1}C_n\sum_{j=1}^{m'-1}\underline{A}_{(n-1,n-j+1)}L_{n-j}C_{n-j}\bar{A}_{(n-j-1,n-m'+1)}\omega_{n-m'}\omega_{n-m'}^{\intercal}\bar{A}_{n-1,n-k+1}^{\intercal}C_n^{\intercal}=0_r, \tag{3.132}$$

we have completed the proof. To show this, we show that the trace of the matrix converges to 0 for each value of j.

$$\begin{aligned}&\operatorname*{p-lim}_{i\to\infty}\frac{1}{i}\sum_{n=0}^{i-1}\left(\omega_{n-m'}^{\intercal}\bar{A}_{(n-1,n-m'+1)}^{\intercal}C_n^{\intercal}C_n\underline{A}_{(n-1,n-j+1)}L_{n-j}C_{n-j}\bar{A}_{(n-j-1,n-m'+1)}\omega_{n-m'}\right)\leq\\&\leq\operatorname*{p-lim}_{i\to\infty}\left(\frac{1}{i}\sum_{n=0}^{i-1}\left\|C_n\bar{A}_{(n-1,n-m'+1)}\omega_{n-m'}\right\|^2\right)^{1/2}\left(\frac{1}{i}\sum_{n=0}^{i-1}\left\|C_n\underline{A}_{(n-1,n-j+1)}L_{n-j}C_{n-j}\bar{A}_{(n-j-1,n-m'+1)}\omega_{n-m'}\right\|^2\right)^{1/2}\end{aligned} \tag{3.133}$$

where the inequality follow from the Cauchy Schwarz Inequality. Let $\epsilon,\tau>0$ be chosen arbitrarily. By Markov's Inequality,

$$\begin{aligned}&\mathbb{P}\left(\frac{1}{i}\sum_{n=0}^{i-1}\left\|C_n\bar{A}_{(n-1,n-m'+1)}\omega_{n-m'}\right\|^2\geq\frac{2\eta_C^2\eta_{A1}^{2(m'-1)}\eta_\omega}{1-\tau}\right)\leq\\&\leq\frac{(1-\tau)\mathbb{E}\left[\frac{1}{i}\sum_{n=0}^{i-1}\left\|C_n\bar{A}_{(n-1,n-m'+1)}\omega_{n-m'}\right\|^2\right]}{2\eta_C^2\eta_{A1}^{2(m'-1)}\eta_\omega}\leq\frac{(1-\tau)\eta_C^2\eta_{A1}^{2(m'-1)}\eta_\omega}{2\eta_C^2\eta_{A1}^{2(m'-1)}\eta_\omega}=\frac{1-\tau}{2}.\end{aligned} \tag{3.134}$$

Furthermore by our choice of m', we have that there exists an N such that $i>N$ implies

$$\mathbb{P}\left(\frac{1}{i}\sum_{n=0}^{i-1}\left\|C_{n-j}\bar{A}_{(n-j-1,n-m'+1)}\omega_{n-m'}\right\|^2\leq\frac{\epsilon^2}{2\eta_C^4\eta_{A1}^{2(m'-1)}\eta_{A2}^{2(j-1)}\eta_L^2\eta_\omega}\right)\geq\frac{\tau+1}{2}. \tag{3.135}$$

Finally, applying Theorem 3

$$\begin{aligned}
&\mathbb{P}\left(\left(\frac{1}{i}\sum_{n=0}^{i-1}\left\|C_n\bar{A}_{(n-1,n-m'+1)}\omega_{n-m'}\right\|^2\right)^{1/2}\times\right.\\
&\qquad\left.\times\left(\frac{1}{i}\sum_{n=0}^{i-1}\left\|C_n\underline{A}_{(n-1,n-j+1)}L_{n-j}C_{n-j}\bar{A}_{(n-j-1,n-m'+1)}\omega_{n-m'}\right\|^2\right)^{1/2}\leq\epsilon\right)\geq\\
&\geq\mathbb{P}\left(\frac{1}{i}\sum_{n=0}^{i-1}\left\|C_{n-j}\bar{A}_{(n-j-1,n-m'+1)}\omega_{n-m'}\right\|^2\leq\frac{\epsilon^2}{2\eta_C^4\eta_{A1}^{2(m'-1)}\eta_{A2}^{2(j-1)}\eta_L^2\eta_\omega}\right)+\\
&\qquad+\left(1-\mathbb{P}\left(\frac{1}{i}\sum_{n=0}^{i-1}\left\|C_n\bar{A}_{(n-1,n-m'+1)}\omega_{n-m'}\right\|^2\geq 2\eta_C^2\eta_{A1}^2\eta_\omega\right)\right)-1\geq \qquad (3.136)\\
&\geq\frac{\tau+1}{2}+1-\frac{1-\tau}{2}-1=\tau. \qquad (3.137)
\end{aligned}$$

Therefore (3.132) must hold. ∎

3.3 Dynamic Watermarking for Distributed Control

This section describes the extension of both LTI and LTV dynamic watermarking to distributed control systems of κ agents as described in Hespanhol *et al.* [85] and Porter *et al.* [100]. We focus on the LTV extension which generalizes its LTI counterpart. However, we comment on the differences throughout. We start by defining the distributed LTV model and necessary assumptions in Subsection 3.3.1. Then, in Subsection 3.3.2, we discuss the extension of the real-time statistical test.

3.3.1 Distributed Control Model

Consider an LTV system of κ distributed agents with combined state x_n, process noise w_n

$$\begin{aligned}
x_{n+1} &= A_n x_n + \sum_{i=1}^{\kappa} B_{i,n}\left(K_{i,n}\hat{x}_{i,n} + e_{i,n}\right) + w_n,\\
\hat{x}_{i,n+1} &= M_{i,n}\hat{x}_{i,n} + N_i B_{i,n} e_{i,n} - \sum_{j\in H_i} L_{(i,j),n} s_{(i,j),n},\\
y_{i,n} &= C_i x_n + z_{i,n},
\end{aligned} \qquad (\mathcal{D}3)$$

where agent $i \in \{1,\dots,\kappa\}$ has state observation $\hat{x}_{i,n}$, measurement $y_{i,n}$, measurement noise $z_{i,n}$, watermark $e_{i,n}$ and communicated measurement from agent $j \in \{1,\dots,\kappa\}$ denoted $s_{(i,j),n}$. Commu-

nicated measurements take the form

$$s_{(i,j),n} = y_{j,n} + v_{(i,j),n} \tag{3.138}$$

where $v_{(i,j),n}$ is an additive attack on the communication channel. This style of attack can take the form of a man in the middle, where the attacker intercepts and alters measurements being communicated between agents, or a malicious agent that sends out false measurements.

The distributed model in ($\mathcal{D}3$) has several key characteristics. First, we do not require that all communications are used as facilitated by the set

$$H \subseteq (\{1, \ldots, \kappa\} \times \{1, \ldots, \kappa\}). \tag{3.139}$$

where the communication from agent j to agent i is *active* if $(i, j) \in H$. Moreover, we define the set

$$H_i = \{j | (i, j) \in H\}, \tag{3.140}$$

which contains the indices of agents that send measurements to agent i.

Second, the dimension and covariance of agent specific parameters is not necessarily constant across all agents. As such, we introduce an additional subscript on affected dimensions

$$e_{i,n} \in \mathbb{R}^{q_i} \quad \text{and} \quad y_{i,n},\ z_{i,n} \in \mathbb{R}^{o_i}, \tag{3.141}$$

and covariances

$$e_{i,n} \sim \mathcal{N}(0_{q_i \times 1}, \Sigma_{e_i}) \quad \text{and} \quad z_{i,n} \sim \mathcal{N}(0_{o_i \times 1}, \Sigma_{z_i,n}). \tag{3.142}$$

Third, we assume that each agent observes a linear combination of the combined state vector using a linear functional observer. This observer follows a discrete time version of the observer first introduced in Luenberger [106]. However, since the watermark of other agents is unknown to agent i their input is not included in the update equation ($\mathcal{D}3$).

Remark 39. *In addition to assuming time-invariance, Hespanhol* et al. *[85] also assumed that each agent observes the entire state and that all communications are active.*

We then make the following assumption.

Assumption 40. *The process and measurement noise are mutually independent and Gaussian*

distributed such that

$$w_n \sim \mathcal{N}(0_{p\times 1}, \Sigma_{w,n}), \tag{3.143}$$

$$z_{i,n} \sim \mathcal{N}(0_{o_i\times 1}, \Sigma_{z_i,n}). \tag{3.144}$$

While the linearization of non-linear systems, generally does not result in Gaussian distributed noise, this assumption allows us to derive our proposed method using a statistical basis. Furthermore, this assumption has not hindered the efficacy of LTV dynamic watermarking in non-networked settings [95].

Next we derive the closed loop dynamics. For ease of notation, we combine each agent's measurement noise and watermark such that

$$z_n^\intercal = \begin{bmatrix} z_{1,n}^\intercal & \cdots & z_{\kappa,n}^\intercal \end{bmatrix}, \tag{3.145}$$

$$\Sigma_{z,n} = \text{diag}(\Sigma_{z_1,n}, \ldots, \Sigma_{z_\kappa,n}), \tag{3.146}$$

$$e_n^\intercal = \begin{bmatrix} e_{1,n}^\intercal & \cdots & e_{\kappa,n}^\intercal \end{bmatrix}, \tag{3.147}$$

$$\Sigma_e = \text{diag}(\Sigma_{e_1}, \ldots, \Sigma_{e_\kappa}). \tag{3.148}$$

The additive attacks are also combined such that

$$v_{i,n}^\intercal = \begin{bmatrix} v_{(i,1),n}^\intercal & \cdots & v_{(i,\kappa),n}^\intercal \end{bmatrix}, \tag{3.149}$$

$$v_n^\intercal = \begin{bmatrix} v_{1,n}^\intercal & \cdots & v_{\kappa,n}^\intercal \end{bmatrix}. \tag{3.150}$$

Then we can write the closed loop system as

$$\bar{x}_{n+1} = \bar{A}_n\bar{x}_n + \bar{B}_n e_n + \begin{bmatrix} w_n \\ 0 \\ \vdots \\ 0 \end{bmatrix} - \bar{L}_n \begin{bmatrix} 0 \\ z_n + v_{1,n} \\ \vdots \\ z_n + v_{\kappa,n} \end{bmatrix}, \tag{3.151}$$

where $\bar{x}_n^\intercal = \begin{bmatrix} x_n^\intercal & \hat{x}_{1,n}^\intercal & \cdots & \hat{x}_{\kappa,n}^\intercal \end{bmatrix}$, and

$$\bar{A}_n = \left[\begin{array}{c:c} A_n & B_{1,n}K_{1,n} \cdots B_{\kappa,n}K_{\kappa,n} \\ \hdashline \begin{matrix} -\sum_{j\in H_1} L_{1,j}C_j \\ \vdots \\ -\sum_{j\in H_\kappa} L_{\kappa,j}C_j \end{matrix} & \operatorname{diag}(M_{1,n},\ldots,M_{\kappa,n}) \end{array}\right], \tag{3.152}$$

$$\bar{B}_n = \left[\begin{array}{c} B_{1,n} \quad \cdots \quad B_{\kappa,n} \\ \hdashline \operatorname{diag}(N_1B_{1,n},\ldots,N_\kappa B_{\kappa,n}) \end{array}\right], \tag{3.153}$$

$$\bar{L}_n = \left[\begin{array}{c} 0_{p\times\left(\kappa\sum_{j=1}^{\kappa} r_j\right)} \\ \hdashline \operatorname{diag}(L_{1,n},\ldots,L_{\kappa,n}) \end{array}\right], \tag{3.154}$$

$$L_{i,n} = \begin{bmatrix} L_{(i,1),n} & \cdots & L_{(i,\kappa),n} \end{bmatrix}. \tag{3.155}$$

On the closed loop system we make the following assumption.

Assumption 41. *Consider a closed loop system satisfying* (3.151)*. If $a_{(i,j),n} = 0_{o_i\times 1}$ for all $(i,j) \in H$ and $n \in \mathbb{N}$, there exists positive constants $\eta_1, \eta_2 \in \mathbb{R}$ such that*

$$\|\mathbb{E}[\bar{x}_n\bar{x}_n^\intercal]\| < \eta_1, \tag{3.156}$$

$$\|\left(\mathbb{E}[\bar{x}_n\bar{x}_n^\intercal]^{-1}\right)\| < \eta_2. \tag{3.157}$$

This assumption ensures that, despite being time varying, the covariance of the closed loop system and its inverse are well defined for all time. For the system in this paper, (3.156) holds since the controllers and observers we later derive render the system stable. Furthermore, we note that our system also satisfies (3.157) as a result of the system noise propagating through the state vector.

3.3.2 Statistical Tests

This section derives the statistical tests for networked LTV dynamic watermarking. These tests utilize the difference between the observed state and the measured state called the *measurement residual*. The measurement residual for each $(i,j) \in H$ is formally defined as

$$r_{(i,j),n} = U_{(i,j)}\hat{x}_{i,n} - W_{(i,j)}s_{(i,j),n}, \tag{3.158}$$

where $r_{(i,j),n} \in \mathbb{R}^{p_{(i,j)}}$. Furthermore, the matrices $U_{(i,j)}$ and $W_{(i,j),n}$ are defined such that

$$U_{(i,j)}N_i = W_{(i,j)}C_j \quad \forall (i,j) \in H. \tag{3.159}$$

Here, the matrices U and W are used to select the common state information between the observed state for agent i and measurement from agent j.

The proposed detection scheme focuses on the covariance of the measurement residual $r_{(i,j),n}$ and its correlation with the watermark $e_{i,n-\rho_{(i,j)}-1}$ where $\rho_{(i,j)}$ is a user defined delay. In the case that the delayed watermark is correlated with the common state information of the un-attacked measurement, an attacker cannot scale the true measurement signal without altering this correlation. As a result, monitoring the watermarks correlation with the measurement residual (which is a function of the measurement) allows our proposed algorithm to detect attacks that scale or remove the true measurement. To ensure a correlation between the delayed watermark and the common state information of the measurement, $\rho_{(i,j)}$ is selected to account for the time needed for the watermark to propagate through the system and into the measurement $s_{(i,j),n}$. Though in general there is no guarantee that a given agent's watermark will eventually propagate through the entire system,, given a particular $\rho_{(i,j)}$ we provide a sufficient condition for a non-zero correlation in the following proposition.

Proposition 42. *[100, Proposition A.1] Consider a closed loop LTV system satisfying* (3.151)*. If for some* $\rho_{(i,j)} \in \mathbb{N}$

$$\begin{bmatrix} W_{(i,j)}C_j & 0_{o_i \times t} \end{bmatrix} \bar{A}_{n-1} \cdots \bar{A}_{n-\rho_{(i,j)}} \bar{B}_{n-\rho(i,j)-1} \begin{bmatrix} 0_{q_i \times \left(p+\sum_{j=1}^{i-1} p_j\right)} & I_{q_i} & 0_{q_i \times \left(\sum_{j=i+1}^{\kappa} p_j\right)} \end{bmatrix}^{\mathsf{T}} \neq 0_{p_{(i,j)} \times q_i} \tag{3.160}$$

then

$$\mathbb{E}\left[W_{(i,j)} s_{(i,j),n} e_{i,n-\rho_{(i,j)}-1}^{\mathsf{T}} \right] \neq 0_{p_{(i,j)} \times q_i} \tag{3.161}$$

Proof. **(Proposition 42)** Due to the assumption that the watermark is mutually independent and zero mean, only terms that are a function of the watermark at that particular step have non-zero expectation. For ease of notation, these terms are removed without loss of generality in this proof. Expanding the communicated measurement on the left side of (3.161) first using (3.138) then (3.151) results in

$$\begin{aligned} \mathbb{E}\left[W_{(i,j)} s_{(i,j),n} e_{i,n-\rho_{(i,j)}-1}^{\mathsf{T}} \right] &= \mathbb{E}\left[W_{(i,j)} C_j x_n e_{i,n-\rho_{(i,j)}-1}^{\mathsf{T}} \right] = \\ &= \mathbb{E}\Big[\begin{bmatrix} W_{(i,j)}C_j & 0_{o_i \times t} \end{bmatrix} \bar{A}_{n-1} \cdots \bar{A}_{n-\rho_{(i,j)}} \bar{B}_{n-\rho_{(i,j)}-1} e_{n-\rho_{(i,j)}-1} e_{i,n-\rho_{(i,j)}-1}^{\mathsf{T}} \Big] = \\ &= \begin{bmatrix} W_{(i,j)}C_j & 0_{o_i \times t} \end{bmatrix} \bar{A}_{n-1} \cdots \bar{A}_{n-\rho_{(i,j)}} \bar{B}_{n-\rho_{(i,j)}-1} \begin{bmatrix} 0_{q_i \times \left(p+\sum_{j=1}^{i-1} p_j\right)} & I_{q_i} & 0_{q_i \times \left(\sum_{j=i+1}^{\kappa} p_j\right)} \end{bmatrix}^{\mathsf{T}} \Sigma_{e_i}. \end{aligned} \tag{3.162}$$

Since Σ_{e_i} is full rank and (3.160) holds, (3.161) holds as well. ∎

Remark 43. *In our initial extension of LTI dynamic watermarking to distributed control systems, a*

method for designing the controller to ensure the existence of $\rho_{(i,j)}$ satisfying (3.161) *was provided [85, Algorithm 1, Algorithm 2].*

To monitor both the covariance of the measurement residual and its correlation with the watermark, we aim to observe the normalized outer product of the vector $[e^{\intercal}_{i,n-\rho_{(i,j)}-1}\ r^{\intercal}_{(i,j),n}]^{\intercal}$, which follows a Gaussian distribution when the platoon is un-attacked such that

$$\begin{bmatrix} r_{(i,j),n} \\ e_{i,n-\rho_{(i,j)}-1} \end{bmatrix} \sim \mathcal{N}\left(0_{(p_{(i,j)}+q_i)\times 1}, \Sigma_{(i,j),n}\right). \tag{3.163}$$

To this end, we start by defining a matrix normalizing factor similar to that of Porter *et al.* [96]

$$V_{(i,j),n} = \Sigma^{-\frac{1}{2}}_{(i,j),n}, \tag{3.164}$$

to create a new vector with constant covariance

$$\Psi_{(i,j),n} = V_{(i,j),n} \begin{bmatrix} r_{(i,j),n} \\ e_{i,n-\rho_{(i,j)}-1} \end{bmatrix} \sim \mathcal{N}(0_{(p_{(i,j)}+q_i)\times 1}, I). \tag{3.165}$$

Next, we form the matrix

$$P_{(i,j),n} = \begin{bmatrix} \Psi_{(i,j),n-\ell_{(i,j)}+1} & \cdots & \Psi_{(i,j),n} \end{bmatrix}, \tag{3.166}$$

which, under the assumption of no attack, is distributed as

$$P_{(i,j),n} \sim \mathcal{N}\left(0_{(p_{(i,j)}+q_i)\times \ell_{(i,j)}}, I_{p_{(i,j)}+q_i}, G_{(i,j),n}\right), \tag{3.167}$$

where

$$G_{(i,j),n} = \frac{\mathbb{E}[P^{\intercal}_{(i,j),n} P_{(i,j),n}]}{p_{(i,j)} + q_i}. \tag{3.168}$$

Note that the values of $V_{(i,j),n}$ and $G_{(i,j),n}$ can be derived as functions of the dynamics and the covariances of the watermarks, process noise, and measurement noise. However, in real word applications these covariances must be approximated. To avoid compounding error, we derive methods for approximating both $V_{(i,j),n}$ and $G_{(i,j),n}$ in Section 3.4. Finally, we form the matrix

$$S_{(i,j),n} = P_{(i,j),n} G^{-1}_{(i,j),n} P^{\intercal}_{(i,j),n}, \tag{3.169}$$

which is distributed according to a Wishart distribution with scale matrix $I_{p_{(i,j)}+q_i}$ i.e.

$$S_{(i,j),n} \sim \mathcal{W}(\ell_{(i,j)}, I_{p_{(i,j)}+q_i}). \tag{3.170}$$

Using this distribution, a statistical test can be implemented using the negative log likelihood of the scale matrix given the observation $S_{(i,j),n}$

$$\mathcal{L}_{(i,j)} = (q_i + p_{(i,j)} + 1 - \ell_{(i,j)}) \log(|S_{(i,j),n}|) + \text{trace}(S_{(i,j),n}) + \log\left(2^{(q_i+p_{(i,j)})\ell/2}\Gamma_{(q_i+p_{(i,j)})}\left(\frac{\ell_{(i,j)}}{2}\right)\right). \tag{3.171}$$

If $\mathcal{L}_{(i,j)}$ exceeds a user defined threshold, agent i signals that an attack has likely occurred. The threshold can be set to satisfy a desired performance metric such as the false alarm rate.

Remark 44. *In our initial extension of LTI dynamic watermarking to distributed control systems in Hespanhol* et al. *[85], we used the sum $\mathcal{L}_i = \sum_{j=1}^{\kappa} \mathcal{L}_{(i,j)}$ as opposed to the individual values. This summation can greatly reduce the total number of tests. However, keeping them separate allows for each communication channel to be monitored separately.*

3.4 System Dependent Parameter Estimation

This section describes methods for estimating the matrix normalizing factor V and the autocorrelation normalizing factor G. First, a general approach is given. Then we derive a practical solutions to the issue of drift along the trajectory which arises for LTV systems. More specifically, we devise a method for allowing the linearization to drift along the trajectory by constructing the discretized trajectory in a receding horizon fashion. Furthermore, we derive an approximation scheme for the normalization matrices used in the statistical tests that accommodates this drift as well.

3.4.1 General Approach

Consider sequences of measurement residuals from several unattacked realizations of a system satisfying ($\mathcal{D}1$), ($\mathcal{D}2$), or ($\mathcal{D}3$) denoted $r_n^{(j)}$ where n is the step number and j is an index for the realization. Without loss of generality, we omit the additional subscripts for distributed systems since the process is the same for each communication channel. For an LTI system the matrix normalizing factor and auto-correlation normalizing factor can then be approximated as

$$V \approx \left(\frac{1}{NJ}\sum_{n}^{N}\sum_{j=1}^{J} r_n r_n^{\mathsf{T}}\right)^{-1/2} \tag{3.172}$$

and

$$G \approx \frac{1}{(N-\ell-\rho-1)J} \sum_{n=\ell+\rho+1}^{N} \sum_{j=1}^{J} \frac{\left(P_n^{(j)}\right)^{\mathsf{T}} P_n^{(j)}}{q+r} \tag{3.173}$$

where N is the number of steps in the realization and J is the total number of realizations used. Similarly for an LTV system the matrix normalizing factor can be approximated as

$$V_n \approx \left(\frac{1}{J}\sum_{j=1}^{J} r_n r_n^{\mathsf{T}}\right)^{-1/2} \tag{3.174}$$

and

$$G_n \approx \frac{1}{J} \sum_{j=1}^{J} \frac{\left(P_n^{(j)}\right)^{\mathsf{T}} P_n^{(j)}}{q+r} \tag{3.175}$$

Finally, for a distributed system the matrix normalizing factor can be approximated as

$$V_n \approx \left(\frac{1}{J}\sum_{j=1}^{J} \begin{bmatrix} r_n^{(j)} \\ e_{i,n-\rho-1}^{(j)} \end{bmatrix} \begin{bmatrix} r_n^{(j)} \\ e_{i,n-\rho-1}^{(j)} \end{bmatrix}^{\mathsf{T}}\right)^{-1/2} \tag{3.176}$$

and

$$G_n \approx \frac{1}{J} \sum_{j=1}^{J} \frac{\left(P_n^{(j)}\right)^{\mathsf{T}} P_n^{(j)}}{q+r} \tag{3.177}$$

3.4.2 Accommodating Drift

We start by creating a *high-resolution* trajectory which uses a step size considerably smaller than that of the step size used when discretizing the dynamics, controllers, and observers. In practice we have chosen to use a step size 10 times smaller. At the n^{th} control step of the system, vehicle i finds its closest point on the high-resolution trajectory denoted $h(n)$, then uses the corresponding linearization.

Similarly to the discretized trajectory, the matrix normalizing factor V_n and the auto-correlation normalizing factor G_n are selected in a receding horizon fashion. We accomplish this by approximating both normalizing factors for each step in the high resolution trajectory then selecting the appropriate normalizing factors at each control step of the system using the index of the high resolution trajectory $h(n)$. To this end, we denote the sample covariance at index k of the high-resolution

trajectory using the ensemble average

$$\bar{\Sigma}_k = \frac{1}{f_k} \sum_{j=1}^{J} \sum_{h^{(j)}(n)=k} \begin{bmatrix} e_{i,n-\rho-1}^{(j)} \\ r_n^{(j)} \end{bmatrix} \begin{bmatrix} e_{i,n-\rho-1}^{(j)} \\ r_n^{(j)} \end{bmatrix}^{\mathsf{T}} \tag{3.178}$$

where the superscript (j) denotes the realization number, J is the total number of realizations, and $f_k = \text{card}(\{(j,n) \mid j \in \{1,\ldots,J\}, h^{(j)}(n) = k\})$. In the event that no samples are available for a given k, we set $\bar{\Sigma}_k = 0_{p+q}$. Since we are limited to having a finite number of realizations, there is no guarantee that the sample covariance matrices all have a sufficient number of samples to be invertable. To overcome this obstacle, we take a weighted average such that

$$\bar{V}_k = \left(\frac{1}{b_k} \sum_{\epsilon=k-10}^{k+10} \sigma^{|k-\epsilon|} \bar{\Sigma}_k \right)^{-\frac{1}{2}}, \tag{3.179}$$

where $0 < \sigma < 1$ is used to reduce the weight of samples that are farther away (we use $\sigma = 0.8$) and b_k is the sum of the weights for which samples exist

$$b_{i,k} = \sum_{\substack{\epsilon=k-10,\ldots k+10 \\ f_\epsilon > 0}} \sigma^{|k-\epsilon|}. \tag{3.180}$$

Here, the range of indices in the summation was chosen such that the number of realizations needed to ensure invertability of $\bar{V}_n$ should be no more than the dimension of the sample covariance matrix. However, the number of samples should exceed this value to ensure a good approximation. Finally, the matrix normalizing factor is approximated at each step as

$$V_n \approx \bar{V}_{h(n)}. \tag{3.181}$$

The auto-correlation normalizing matrices G_n can then be approximated using (3.165)-(3.166) and the approximate matrix normalization factor using the ensemble average

$$\begin{aligned} \bar{G}_k = \frac{1}{(p+q)g_k} \sum_{j=1}^{J} \sum_{\substack{h^{(j)}(n) \le k \\ h^{(j)}(n+1) > k}} \Bigg(& \frac{1}{|h^{(j)}(n+1) - h^{(j)}(n)|} \times \\ & \times \left(|k - h^{(j)}(n+1)| \left(P_n^{(j)}\right)^{\mathsf{T}} P_n^{(j)} + |k - h^{(j)}(n)| \left(P_{n+1}^{(j)}\right)^{\mathsf{T}} P_{n+1}^{(j)} \right) \Bigg), \end{aligned} \tag{3.182}$$

where the superscript (j) denotes the realization number, J is the total number of realizations, and $g_k = \text{card}(\{(j,n) | j \in \{1,\ldots,J\},\ h^{(j)}(n) \le k,\ h^{(j)}(n+1) > k\})$. The auto-correlation normalizing

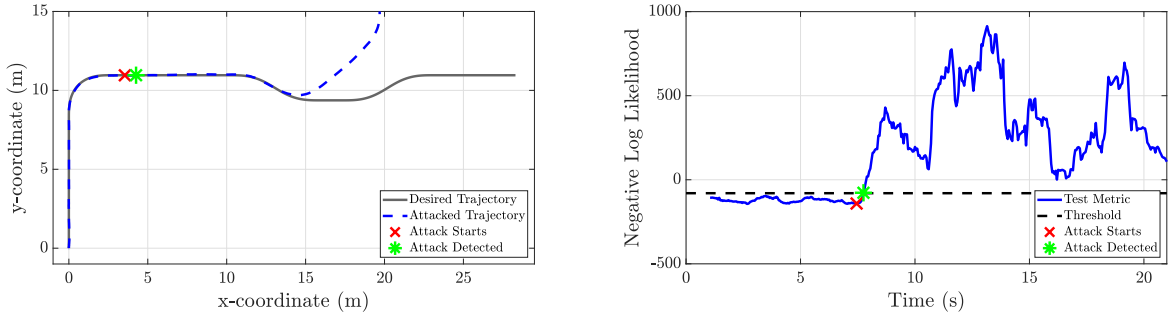


Figure 3.1: Desired and attacked trajectory of an LTV car model showing attack start and detection (Left); Corresponding LTV Dynamic Watermarking test metric showing attack start and detection (Right)

matrix is then approximated at each step as

$$G_n \approx \bar{G}_{h(n)}. \tag{3.183}$$

3.5 Discussion

The use of LTV instead of LTI dynamic watermarking can greatly increase the number of trials necessary to obtain the parameters V and G and increases the complexity of the algorithm. Since the linear model is often approximated, it may be tempting to oversimplify the model to reduce the complexity. However, this can result in adverse effects. In this section, we provide examples to illustrate the potential effects of using LTI instead of LTV dynamic watermarking and of adding the auto-correlation normalizing factor.

Measurement	Expected Std Dev	Over-approx. Std Dev
(x, y)	≤ 3 cm [107]	$2\sqrt{10} \approx 6.3$ cm
ψ	$< 3 \times 10^{-3}$ rad [108]	6×10^{-3} rad
s	0.2 cm/s to 5 cm/s [109]	$\sqrt{10}s_n \approx 3.2s_n$ cm/s
$\dot{\psi}$	2×10^{-4} rad/s [110]	4×10^{-4} rad/s

Table 3.2: The standard deviation of measurement noise from a real-world RTK GNSS and an IMU system and the standard deviation of measurement noise used in the experiment. Note that the measurement noise used in the experiment over-approximates the noise one would expect to see in the real-world.

3.5.1 Comparing LTI to LTV Dynamic Watermarking

To provide proof of concept, we use a simplified car model

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\psi} \\ \dot{s} \\ \dot{\psi} \end{bmatrix}^{\mathsf{T}} = \begin{bmatrix} s\cos(\psi) \\ s\sin(\psi) \\ \dot{\psi} \\ a \\ \ddot{\psi} \end{bmatrix}^{\mathsf{T}}, \tag{3.184}$$

where the car has ground plane coordinates (x, y), heading ψ, forward velocity s, and angular velocity $\dot{\psi}$. Using the desired trajectory shown in Figure 3.1, (3.184) is linearized and discretized using a step size of 0.05 and zero order hold on the current state and input. Note, for the discretized system, Assumption 24 holds. The controller and observer for the resulting LTV system are found using a linear quadratic regulator (LQR) to stabilize the system, by enforcing a bound on $\bar{A}$ and $\underline{A}$ as stated in Assumptions 19 and 20. While linearizing non-linear stochastic systems often results in noise that is not independent zero mean Gaussian distributed, for this example we approximate it as such where $w_n \sim \mathcal{N}(0_{5\times1}, 10^{-5}I_5)$, $z_n \sim \mathcal{N}(0_{5\times1}, \text{diag}(4I_2\times10^{-3}, 3.6\times10^{-5}, s_n^2\times10^{-3}, 1.6\times10^{-7}))$. Note that the vehicle maintain a speed of 1.5 m/s to 3 m/s. As a result, this measurement noise over-approximates that of a vehicle relying upon RTK GNSS (within 20 km of a base station and using multiple antenna spaced 1 m apart) for measuring the ground plane positioning, heading, and velocity and an IMU for measuring angular velocity. Table 3.2 shows both the expected and over-approximated standard deviations of the measurement noise.

To compare LTI and LTV Dynamic Watermarking, a time invariant matrix normalization factor is calculated using the average of the residual covariance, while the time-varying matrix normalization factor is calculated using (3.174) with 100 realizations. For both cases, we run 100 simulations with a window size of 20 and calculate the test metric and the average test metric as shown in Figure 3.2. Note, while the LTV Dynamic Watermarking metric remains consistent over the simulation, the LTI counterpart has a repeatable time-varying pattern.

Using the un-attacked data, a threshold for the LTV case is found such that the rate at which false alarms occur does not exceed once per every 50 seconds of run time. Next consider an attack model satisfying (6.15)-(3.53), with α equal to -1 and the measurement and process noise matching that of the true system. The results of this attack on the system, and the ability of LTV Dynamic Watermarking to quickly detect it, are shown in Figure 3.1.

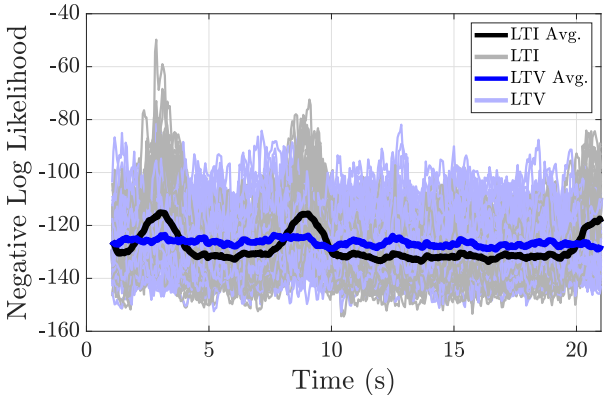


Figure 3.2: Simulated LTI and LTV dynamic watermarking test metrics for LTV car model under no attack

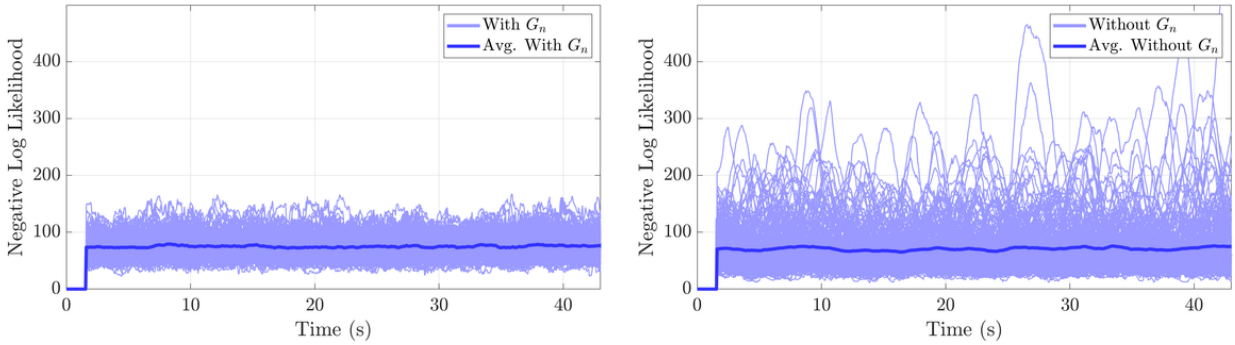


Figure 3.3: An example LTV system is simulated 200 times and the negative log likelihood is generated with the auto-correlation normalizing factor, G_n, (Left) and without the auto-correlation normalizing factor (Right).

3.5.2 Effect of Auto-Correlation

To illustrate the effect of the auto-correlation normalizing factor we present the following example [95, Example V.1]. Consider an LTV system satisfying the dynamics in ($\mathcal{D}2$) where $v_n = 0$ for all n, $w_n \sim \mathcal{N}(0_{3\times1}, 1\times10^{-3}I_3)$, $z_n \sim \mathcal{N}(0_{2\times1}, 1\times10^{-3}I_2)$, $e_n \sim \mathcal{N}(0, 1\times10^{-3})$,

$$A_n = \begin{bmatrix} 1 & 1+\frac{1}{2}\sin(\frac{n}{100}) & 0 \\ 0 & 1 & 0.1 \\ 0 & 0 & 1 \end{bmatrix}, \quad B_n = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}^{\mathsf{T}}, \quad C_n = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}, \tag{3.185}$$

and the controller and observer gains satisfy

$$K_n = \begin{bmatrix} -4\times10^{-4} \\ -3.65\times10^{-2} \\ -1.05\times10^{-1} \end{bmatrix}^{\mathsf{T}}, \quad L_n = \begin{bmatrix} -7\times10^{-2} & -1 \\ -2.2\times10^{-3} & -1.4\times10^{-1} \\ -1.6\times10^{-3} & -4.5\times10^{-2} \end{bmatrix}. \tag{3.186}$$

Note, for this system ρ is 2. The test metric was generated for 200 simulated realizations both with and without the auto-correlation normalizing factor G_n for a window size ℓ of 20.

As illustrated in Figure 3.3, the addition of the auto-correlation normalizing factor has little effect on the average of the negative log likelihood. However, this normalizing factor does improve the consistency by removing anomalies in many of the realizations caused by auto-correlation.

Chapter 4

Tools for Selecting User-Defined Parameter

Attack detection algorithms in literature differ in many ways. However, from an outside perspective they operate in much the same way. To detect an attack, the detector evaluates a test statistic that is a function of the measurement residual. If this statistic's value rises above some user-specified threshold, then the detector triggers an alarm.

The choice of threshold, in addition to any algorithm specific parameters have a direct affect on the performance of the detector. To evaluate and design the threshold for a detector, researchers have proposed the following three metrics: first, the *attack capability* or the amount of perturbation to the state of the system that an attack can induce without either inducing an alarm [68] or without increasing the rate of alarms [111], [112]; second, the rate of false alarms (R_{FA}) given by the detector when no attack is occurring; and third, the ability of the detector to reliably detect specific attack models. For an open-loop stable system, the attack capability can be evaluated by computing the reachable set of the error in the observed state. Since computing this reachable set can be challenging, researchers have instead attempted to evaluate surrogates for the attack capability such as the expected value of the state vector [68] or the norm of the largest time invariant normalized residual [69]. However, these surrogates are unable to accurately characterize the attack capability of attacks that have large residuals for short amounts of time. Note that, by reducing the threshold in any detector, one can reduce the attack capability; however, this can increase the rate of false alarms. To compute this false alarm rate for classic anomaly detectors, it is typically assumed that the residuals are independent [68], [69]. However, in real-world systems, this assumption does not hold. As such we compute the false alarm rate empirically. Though dynamic watermarking is proven to be capable of detecting a larger class of attacks when compared to prior detection algorithms, to the best of our knowledge, no one has conducted a real-world evaluation of any of these attack detection algorithms. Moreover, no one has evaluated the attack capability of a system employing a dynamic watermarking scheme as a function of false alarm rate or developed a technique to design a detector using dynamic watermarking that achieves a user-specified false

alarm rate.

In this chapter we start by introducing each of the additional detection algorithms discussed in this chapter in Section 4.1. Then we derive a method for measuring the attack capability in Section 4.2. Our proposed method can be applied to both dynamic watermarking and all previous algorithms as well. Using the attack capability we provide a comparison of each detection algorithm. Then in Section 4.3, we further compare each detection algorithm by subjecting a real-world system to a variety of attack models and applying each of detection algorithm. Video of these experiments is available at Porter *et al.* [99].

4.1 Other Detection Schemes

This section describes the technical details of each of the additional detection algorithms explored in this chapter. Namely we describe three detectors which were originally proposed for anomaly detection in quality control applications [113]: the χ^2, cumulative sum (CUSUM), and multivariate exponentially weighted moving average (MEWMA) detectors [68], [69], [114]. At each time step, each detector computes a test statistic, $a_*(R_n)$, based on current and previous residuals. The subscript $*$ is a placeholder for the detector designation, and n is the discrete time step. If, for a given detector, the test statistic exceeds a predefined threshold, then the detector raises an alarm. These thresholds are denoted by τ_*.

4.1.1 χ^2 Detector

The χ^2 detector uses the normalized residual $\bar{r}_n$ to develop a statistical test. Since, under the assumption of no attack, $\bar{r}_n \sim \mathcal{N}(0, I)$, the χ^2 *detector* is defined as:

$$a_{\chi^2}(R_n) = \bar{r}_n^T \bar{r}_n < \tau_{\chi^2}, \tag{4.1}$$

where the test statistic $a_{\chi^2}(R_n) \sim \chi^2(q)$ when the system is not under attack. A change to the distribution of the residual, as a result of an attack, may change the resulting distribution of the χ^2 test value, but this is not true for all attacks. For instance an attack could replace the residual vectors with:

$$r'_n = \Sigma_r^{1/2} \bar{r}'_n = \Sigma_r^{1/2} \begin{bmatrix} \sqrt{c_n} & 0 & \dots & 0 \end{bmatrix}^T \tag{4.2}$$

where $c_n \sim \chi^2(q)$. Then $\bar{r}_n'^T \bar{r}'_n = c_n \sim \chi^2(q)$. Similarly some attacks, such as the one described in Section 4.3, can generate a false set of residuals that have the same distribution as the residual when no attack is taking place. Such attacks would be indistinguishable by the χ^2 detector while

increasing the error in the observed state. Furthermore, the χ^2 detector is memoryless, which can make detecting small increases in the norm of the residuals difficult.

4.1.2 CUSUM Detector

The CUSUM detector addresses the difficulty in detecting small but persistent increases in the norm of the normalized residual by introducing dynamics to its test statistic:

$$a_C(R_n) = \max(a_C(R_{n-1}) + \bar{r}_n^T \bar{r}_n - \gamma, 0) < \tau_C, \tag{4.3}$$

where $a_C(R_{-1}) = 0$, and γ is a parameter called the *forgetting factor*. To ensure that the test statistic is stable, $\gamma > q$ where q is the dimension of the residual [68, Theorem 1]. Similar to the χ^2 detector, the CUSUM detector bounds the norm of the normalized residual under the assumption of no alarms:

$$\bar{r}_n^T \bar{r}_n - \gamma \leq a_C(R_{n-1}) + \bar{r}_n^T \bar{r}_n - \gamma < \tau_C. \tag{4.4}$$

For the CUSUM detector, a persistent increase in the norm of the normalized residual, increases the likelihood that $\bar{r}_n^T \bar{r}_n > \gamma$. When this is true for several steps, the CUSUM test value increases cumulatively, triggering an alarm.

4.1.3 MEWMA Detector

The MEWMA detector test statistic also incorporates dynamics. The MEWMA detector uses the exponentially weighted moving average of the normalized residual:

$$G_n = \beta \bar{r}_n + (1 - \beta) G_{n-1} \tag{4.5}$$

where $G_{-1} = 0$ and the parameter $\beta \in (0, 1]$ is also called the *forgetting factor*. The MEWMA detector is then defined as:

$$a_M(R_n) = \frac{2 - \beta}{\beta} G_n^T G_n < \tau_M. \tag{4.6}$$

When $\beta = 1$ the test statistic is equal to the χ^2 detector's test statistic. For smaller β, one gets a similar effect to that of the CUSUM detector, because, for a forgetting factor $\beta \in (0, 1)$, a persistent increase in the norm of the residual results in a larger value of G which results in a higher test statistic value. However, the MEWMA test statistic does not increase for all persistent changes. For instance, if the covariance of the residuals under attack are $\alpha \Sigma_r$ for some $\alpha \in (0, 1)$, we expect

a lower test value.

4.2 Attack Capability

Assuming the A matrix is Schur Stable, the capability of an attack can be measured by its ability to affect the observer error δ_n. A reachable set of the observer error can evaluate the attack capability, but, since the noise is supported over an infinitely large set, this reachable set would have infinite volume. As a result, this work focuses on computing the volume of the reachable set of the portion of the observer error corresponding to the residual under the condition of no alarms being raised. To provide a rigorous definition of this set, we introduce some additional notation and definitions.

Using superposition, one can split the observer error described in (3.5) into two pieces:

$$\delta_{n+1}^{(a)} = (A + LC)\delta_n^{(a)} - Lz_n - Lv_n \tag{4.7}$$

$$\delta_{n+1}^{(b)} = (A + LC)\delta_n^{(b)} - w_n. \tag{4.8}$$

The observer error is then $\delta_n = \delta_n^{(a)} + \delta_n^{(b)}$. Here $\delta_n^{(a)}$ is the portion related to the residual, which can be seen by applying (3.3) to (4.7):

$$\delta_{n+1}^{(a)} = A\delta_n^{(a)} + L\Sigma_r^{1/2}\bar{r}_n. \tag{4.9}$$

where $\bar{r}_n$ is the normalized measurement residual

$$\bar{r}_n = Vr_n = \Sigma_r^{-1/2} r_n. \tag{4.10}$$

Furthermore, we define the vector of current and previous normalized residuals:

$$R_n = [\bar{r}_n^T \dots \bar{r}_0^T]^T. \tag{4.11}$$

Since an attack is only able to affect the $\delta_n^{(a)}$ portion of the observer error, the other portion is ignored while evaluating attack capability. For each $n \in \mathbb{N}$, denote the *reachable set of* $\delta_n^{(a)}$ *at a given time step n under the condition of no alarms for a threshold* τ_* as $\mathcal{R}_n^{\tau_*}$ and define it as:

$$\mathcal{R}_n^{\tau_*} = \{\delta_n^{(a)} \mid \delta_n^{(a)} = \bar{A}_{n-1}R_{n-1},\ R_{n-1} \in \Omega_n^{\tau_*}\} \tag{4.12}$$

where:

$$\Omega_{n-1}^{\tau_*} = \{R_{n-1} \mid a_*(R_{n-1}) < \tau_*\ \forall i < n\}, \tag{4.13}$$

and:

$$\bar{A}_n = \begin{bmatrix} L\Sigma_r^{1/2} & AL\Sigma_r^{1/2} \dots & A^n L\Sigma_r^{1/2} \end{bmatrix}. \tag{4.14}$$

Furthermore, we denote the *steady state reachable set under the condition of no alarms for a threshold* τ_* as $\mathcal{R}^{\tau_*}$ and define it as:

$$\mathcal{R}^{\tau_*} = \{\delta^{(a)} \mid \forall n \in \mathbb{N},\ \exists m \in \mathbb{N} \text{ s.t. } m > n, \delta^{(a)} \in \mathcal{R}_m^{\tau_*}\}. \tag{4.15}$$

Finally, we evaluate the attack capability by measuring the volume of the steady state reachable set under the condition of no alarms under a threshold τ_*, which is defined as:

$$V_{RS}(\tau_*) = \mu(\mathcal{R}^{\tau_*}), \tag{4.16}$$

where μ denotes the Lebesgue measure.

Calculating the set $\mathcal{R}^{\tau_*}$ can be difficult, so we first derive a method for calculating $\mathcal{R}_n^{\tau_*}$:

Theorem 45. *[61, Theorem 2] Suppose* $\tau_* \in \mathbb{R}$ *and* $\bar{A}_{n-1}$, R_{n-1}, $\mathcal{R}_n^{\tau_*}$, *and* $\Omega_{n-1}^{\tau_*}$ *are as in* (4.14), (4.11), (4.12), *and* (4.13), *respectively. Suppose* $v : \mathbb{R}^q \to \mathbb{R}$ *is the solution to:*

$$\inf_{v \in C} \int v(\delta)\, d\delta \tag{4.17}$$

$$s.t.\quad v(\delta) \geq 0 \qquad \delta \in \mathbb{R}^q \tag{4.18}$$

$$v(\bar{A}_{n-1}R_{n-1}) - 1 \geq 0 \qquad R_{n-1} \in \Omega_{n-1}^{\tau_*} \tag{4.19}$$

where C *is the space of continuous functions. Then the 1 super-level set of* v *is an outer approximation to* $\mathcal{R}_n^{\tau_*}$

Proof. **(Theorem 45)** Let $\delta \in \mathcal{R}_n^{\tau_*}$. Then, from (4.12), there exists an $R_{n-1} \in \Omega_{n-1}^{\tau_*}$ such that $\delta = \bar{A}_{n-1}R_{n-1}$. The constraint in (4.19) then gives $v(\bar{A}_{n-1}R_{n-1}) = v(\delta) \geq 1$. ∎

To make this problem computationally tractable, we optimize over polynomial functions of fixed degree instead of continuous functions, and describe the positivity constraint, (4.19), with a Sums-of-Squares constraint. We then apply Sums-of-Squares Programming to generate an outer approximation to the reachable set. To replace (4.19) with a Sums of Squares constraint, $\Omega_n^{\tau_*}$ must first be replaced with a semi-algebraic set [115, Theorem 2.14]. To simplify our exposition, we denote by $\Theta_n^{\tau_*}$ a collection of semi-algebraic constraints such that $\Omega_n^{\tau_*} \subseteq \Theta_n^{\tau_*}$. In fact, as we show next, for many detectors, $\Omega_n^{\tau_*} = \Theta_n^{\tau_*}$.

For the χ^2 detector, the constraint of no alarms is a quadratic constraint on the residual, so:

$$\Theta_n^{\tau_{\chi^2}} = \left\{R_n \mid R_n^T Q_{(i,n)}^{\tau_{\chi^2}} R_n < 1 \; i = 0, ..., n\right\} \tag{4.20}$$

where:

$$Q_{(i,n)}^{\tau_{\chi^2}} = \frac{1}{\tau_{\chi^2}} \begin{bmatrix} 0_{o(n-i)} & 0 & 0 \\ 0 & I_o & 0 \\ 0 & 0 & 0_{o(i)} \end{bmatrix}. \tag{4.21}$$

Note that $\Theta_n^{\tau_{\chi^2}} = \Omega_n^{\tau_{\chi^2}}$ since $\frac{a_{\chi^2}(R_i)}{\tau_{\chi^2}} = R_n^T Q_{(i,n)}^{\tau_{\chi^2}} R_n$ for all $i \leq n$.

For the CUSUM detector:

$$\Theta_n^{\tau_C} = \left\{R_n \mid R_n^T Q_{(i,j,n)}^{\tau_C} R_n < 1 \; i = 0, ..., n \; j \leq i\right\} \tag{4.22}$$

where:

$$Q_{(i,j,n)}^{\tau_C} = \frac{1}{\tau_C + \gamma(j+1)} \begin{bmatrix} 0_{o(n-i)} & 0 & 0 \\ 0 & I_{o(j+1)} & 0 \\ 0 & 0 & 0_{o(i-j)} \end{bmatrix}. \tag{4.23}$$

Note that $\Theta_n^{\tau_C} = \Omega_n^{\tau_C}$, since:

$$a_C(R_i) = \max\left(\left\{\sum_{h=i-j}^{i} (\bar{r}_h^T \bar{r}_h - \gamma) \mid j \leq i\right\}, 0\right) \tag{4.24}$$

and:

$$R_n^T Q_{(i,j,n)} R_n = \frac{1}{\tau_C + \gamma(j+1)} \sum_{h=i-j}^{i} \bar{r}_h{}^T \bar{r}_h < 1 \tag{4.25}$$

can be rearranged to form:

$$\sum_{h=i-j}^{i} (\bar{r}_h^T \bar{r}_h - \gamma) < \tau_C. \tag{4.26}$$

For the MEWMA detector note that:

$$\Theta_n^{\tau_M} = \left\{R_n \mid R_n^T Q_{(i,n)}^{\tau_M} R_n < 1 \; i = 0, ...n\right\} \tag{4.27}$$

where:

$$Q^{\tau_M}_{(i,n)} = \frac{2-\beta}{\beta\tau_M} \begin{bmatrix} 0_{o(n-i)\times o} \\ \beta I_o \\ (1-\beta)\beta I_o \\ \vdots \\ (1-\beta)^i \beta I_o \end{bmatrix} \begin{bmatrix} 0_{o(n-i)\times o} \\ \beta I_o \\ (1-\beta)\beta I_o \\ \vdots \\ (1-\beta)^i \beta I_o \end{bmatrix}^T . \tag{4.28}$$

Note that $\Theta^{\tau_M}_n = \Omega^{\tau_M}_n$ since $\frac{a_M(R_i)}{\tau_M} = R_n^T Q^{\tau_M}_{(i,n)} R_n$ for all $i \leq n$.

While $\Omega^{\tau_*}_n$ is already a semi-algebraic set for the χ^2, CUSUM and the MEWMA detectors, this is not true for the Dynamic Watermarking detector due to the log function in (3.38). Therefore, we consider an outer approximation to $\Omega^{\tau_\mathcal{D}}_n$ described via a quadratic constraint:

Theorem 46. *[61, Theorem 3] Suppose $\Omega^{\tau_\mathcal{D}}_n$ is as in* (4.13)*, ℓ is the window size of the Dynamic Watermarking detector, $\tau_\mathcal{D}$ is the threshold of the detector, ρ satisfies* (3.11)*, o is the dimension of the residual, q is the dimension of the input signal, and:*

$$\Theta^{\tau_\mathcal{D}}_n = \left\{ R_n \mid R_n^T Q^{\tau_\mathcal{D}}_{(i,n)} R_n < 1 \; i = \ell + k', ..., n \right\}, \tag{4.29}$$

where:

$$Q^{\tau_\mathcal{D}}_{(i,n)} = \frac{1}{(q+o)\epsilon} \begin{bmatrix} 0_{o(n-i)} & 0 & 0 \\ 0 & I_{o\ell} & 0 \\ 0 & 0 & 0_{o(i-\ell)} \end{bmatrix} \tag{4.30}$$

and where $\epsilon > \ell - 1 - o - q$ is a solution to:

$$\tau_\mathcal{D} = \frac{(o+q)\epsilon}{2} + \frac{(o+q+1-\ell)}{2} \log(\epsilon^{o+q}) + \log\left(2^{(o+q)\ell/2} \Gamma_{(o+q)}\left(\frac{\ell}{2}\right)\right). \tag{4.31}$$

Then $\Omega^{\tau_\mathcal{D}}_n \subset \Theta^{\tau_\mathcal{D}}_n$.

To prove this theorem, consider the following lemma:

Lemma 47. *[61, Lemma 1] Suppose $(g_i)_{i\in\mathbb{N}}$ is a sequence of vectors where $g_i \in \mathbb{R}^q$, $\tau \in \mathbb{R}$ such that $\tau > 0$, and $\ell \in \mathbb{N}$ such that $\ell > o + 1$. Furthermore suppose that:*

$$\mathcal{L}^\ell_o \left(\sum_{i=1}^{\ell} g_i g_i^T \right) < \tau \tag{4.32}$$

where the function $\mathcal{L}_o^\ell$ is a generalization of (3.38) *such that*

$$\mathcal{L}_i^j(X) = \frac{(i+1-j)}{2} \cdot \log(|X|) + \frac{1}{2}\text{trace}\,(X) + \log\left(2^{ij/2}\Gamma_{(i)}\left(\frac{j}{2}\right)\right). \tag{4.33}$$

Then:

$$\sum_{i=1}^{\ell} g_i^T g_i < \epsilon o, \tag{4.34}$$

where $\epsilon > \ell - 1 - o$ is a solution to:

$$\tau = \frac{(o)\epsilon}{2} + \frac{(o+1-\ell)}{2}\log(\epsilon^o) + \log\left(2^{(o)\ell/2}\Gamma_{(o)}(\frac{\ell}{2})\right). \tag{4.35}$$

Proof. **(Lemma 47)** Denote the eigenvalues of $\sum_{i-1}^{\ell} g_i g_i^T$ as $\lambda_1, ...\lambda_o$. The eigenvalues are all non-negative due to the construction of the matrix. Note that we can rewrite (4.33) as a new function $\mathfrak{Q}_i^j$ in terms of these eigenvalues:

$$\mathcal{L}_o^\ell\left(\sum_{i-1}^{\ell} g_i g_i^T\right) = \mathfrak{Q}_o^\ell(\lambda_1, ..., \lambda_o) \tag{4.36}$$

$$= \sum_{i=1}^{o} \frac{(o+1-\ell)}{2} \cdot \log(\lambda_i) + \frac{\lambda_i}{2} + \log\left(2^{(o)\ell/2}\Gamma_{(o)}\left(\frac{\ell}{2}\right)\right). \tag{4.37}$$

Furthermore we have that $\mathfrak{Q}_o^\ell$ is convex since:

$$\nabla^2\mathfrak{Q}_o^\ell(\lambda_1, ..., \lambda_o) = \begin{bmatrix} \frac{(\ell-1-o)}{2\lambda_1^2} & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \frac{\ell-1-o}{2\lambda_o^2} \end{bmatrix} \tag{4.38}$$

is positive definite for $\lambda_i > 0$. Also note that the function achieves a global minimum at $\lambda_i = \ell - 1 - o,\ i = 1, ..., o$ since:

$$\nabla\mathfrak{Q}_o^\ell(\lambda_1, ..., \lambda_o) = \begin{bmatrix} \frac{o+1-\ell}{2\lambda_1} + \frac{1}{2} \\ \vdots \\ \frac{o+1-\ell}{2\lambda_o} + \frac{1}{2} \end{bmatrix} \tag{4.39}$$

is zero at this point. If we consider the particular case where $\lambda_1 = ... = \lambda_o = \epsilon$. Then $(\epsilon, ..., \epsilon)$ is a

boundary point to the τ level set of $\mathfrak{Q}_o^\ell$. Furthermore we have that the derivative at that point is:

$$\nabla \mathfrak{Q}_o^\ell(\epsilon, ..., \epsilon) = \begin{bmatrix} \frac{o+1-\ell}{2\epsilon} + \frac{1}{2} \\ \vdots \\ \frac{o+1-\ell}{2\epsilon} + \frac{1}{2} \end{bmatrix} \tag{4.40}$$

which is some positive scalar times the vector $\begin{bmatrix} 1 & \dots & 1 \end{bmatrix}^T$. Since the tangent plane at this point is a supporting hyperplane to the τ sublevel set of $\mathfrak{Q}_o^\ell$ we then have that:

$$\sum_{i=1}^{\ell} g_i^T g_i = \sum_{i=1}^{o} \lambda_i < \epsilon o \tag{4.41}$$

for all g_i such that $\mathcal{L}_o^\ell(\sum_{i=1}^{\ell} g_i g_i^T) < \tau$. ■

Now we return to the prove the theorem

Proof. **(Theorem 46)** For a given $R \in \Omega_n^{\tau_{\mathcal{D}}}$:

$$a_{\mathcal{D}}(R_i) = \mathcal{L}_{(m+o)}^\ell \left(\sum_{j=i-\ell+1}^{i} \psi_j \psi_j^T \right) < \tau_{\mathcal{D}} \tag{4.42}$$

for $i = \ell + \rho, ..., n$. Lemma 47 then gives us that:

$$\sum_{j=i-\ell+1}^{i} \psi_j^T \psi_j = \sum_{j=i-\ell+1}^{i} \bar{r}_j^T \bar{r}_j + \sum_{j=i-\ell+1}^{i} e j^T e_j < (m+o)\epsilon \tag{4.43}$$

for $i = \ell + \rho, ..., n$. Furthermore we have that:

$$R_n^T Q_{i,n}^{\tau_{\mathcal{D}}} R_n = \sum_{j=i-\ell+1}^{i} \bar{r}_j^T \bar{r}_j < (q+o)\epsilon \tag{4.44}$$

for $i = \ell + \rho, ..., n$. Therefore $R \in \Theta_n^{\tau_{\mathcal{D}}}$. ■

Now, we construct an outer approximation to $\mathcal{R}_n^{\tau_*}$ using the constraint sets $\Theta_n^{\tau_*}$:

Theorem 48. *[61, Theorem 4] Suppose $\bar{A}_{n-1}$ and R_{n-1} are defined as in* (4.14) *and* (4.11) *respectively, $\mathcal{R}_n^{\tau_*}$ is the set in* (4.12)*, Φ is a compact semi-algebraic set such that $\mathcal{R}_n^{\tau_*} \subset \Phi$, $\Theta_{n-1}^{\tau_*}$ is defined based on the choice of detector and:*

$$H_n^{\tau_*} = \frac{1}{1-c} H \tag{4.45}$$

where H and c are the solution to:

$$\inf_{H\in S\ c\in\mathbb{R}} \quad \int_{\Phi} \left(\delta^T H\delta + c\right)\, d\delta \tag{4.46}$$

$$s.t.\ \ \delta^T H\delta + c \geq 0 \qquad \delta \in \Phi \tag{4.47}$$

$$R_{n-1}^T \bar{A}_{n-1}^T H \bar{A}_{n-1} R_{n-1} + c - 1 \geq 0 \qquad R_{n-1} \in \Theta_n^{\tau_*} \tag{4.48}$$

where $S \subset \mathbb{R}^{p\times p}$ is the set of symmetric matrices. Then $\mathcal{R}_n^{\tau_*} \subseteq \{\delta \mid \delta^T H_n^{\tau_*}\delta \leq 1\}$.

Proof. **(Theorem 48)** Let $\delta \in \mathcal{R}_n^{\tau_*}$. Then, from (4.12), we have that there exists an $R_{n-1} \in \Omega_{n-1}^{\tau_*} \subseteq \Theta_n^{\tau_*}$ such that $\delta = \bar{A}_{n-1}R_{n-1}$. Constraint (4.48) then gives $R_{n-1}^T \bar{A}_{n-1}^T H \bar{A}_{n-1} R_{n-1} + c \geq 1$. Furthermore $c > 1$ since $0 \in \Omega_{n-1}^{\tau_*}$, so we can rearrange the inequality resulting in $\delta^T \frac{1}{1-c} H\delta = \delta^T H_n^{\tau_*}\delta \leq 1$. ■

One can solve the program in Theorem 48 using the Spotless optimization toolbox [116] which formulates the problem as a Semi-Definite Program that can be solved using commercial solvers such as MOSEK [117]. This program assumes that we can find a compact semi-algebraic set Φ that outer approximates $\mathcal{R}_n^{\tau_*}$, which can be done using the following lemma under the specific case that $N = n$:

Lemma 49. *[61, Lemma 2] Suppose $N, n \in \mathbb{N}$, such that $N \geq n$ and if applicable, suppose $N > \ell+\rho$ if the detector is the dynamic watermarking detector. Furthermore suppose $\tau_* \in \mathbb{R}$ such that $\tau_* > 0$, $\bar{A}_{n-1}$ and R_{N-1} are as in* (4.14),(4.11)*. Then there exists a $\eta \in \mathbb{R}$ such that:*

$$\{\delta = [0_{o\times o(N-n)}\ \bar{A}_{n-1}]R_{N-1} \mid R_{N-1} \in \Theta_{N-1}^{\tau_*}\} \subset \mathcal{B}_\eta. \tag{4.49}$$

Proof. **(Lemma 49)** First we show that $\Theta_{N-1}^{\tau_*}$ is bounded. We denote the upper bounds for the norm of elements in $\Theta_{N-1}^{\tau_*}$ as σ^{τ_*}, and we use the decomposition of $R_{N-1} = [\bar{r}_{N-1}^T\ \dots\ \bar{r}_0^T]^T \in \Theta_{N-1}^{\tau_*}$. For the χ^2 detector we have that $\sigma^{\tau_{\chi^2}} = \sqrt{N\tau_{\chi^2}}$ since:

$$\|[\bar{r}_{N-1}^T\ \dots\ \bar{r}_0^T]^T\| = \sqrt{\sum_{i=0}^{N-1} \bar{r}_i^T \bar{r}_i} \leq \sqrt{N\tau_{\chi^2}}. \tag{4.50}$$

Similarly for the CUSUM detector we have that $\sigma^{\tau_C} = \sqrt{N(\tau_C + \delta)}$ since:

$$\|[\bar{r}_{N-1}^T\ \dots\ \bar{r}_0^T]^T\| = \sqrt{\sum_{i=0}^{N-1} \bar{r}_i^T \bar{r}_i} \leq \sqrt{N(\tau_C + \delta)}. \tag{4.51}$$

In the case of the MEWMA detector we have that $\sigma^{\tau_M} = \sqrt{\frac{N\tau_M(2-\beta)}{\beta}}$ since:

$$\|G_i\| = \|\beta\bar{r}_i + (1-\beta)G_{i-1}\| \leq \sqrt{\frac{\tau_M\beta}{2-\beta}} \tag{4.52}$$

and:

$$\beta\|\bar{r}_i\| - (1-\beta)\sqrt{\frac{\tau_M\beta}{2-\beta}} \leq \|\beta\bar{r}_i + (1-\beta)G_{i-1}\|. \tag{4.53}$$

Combining (4.52) and (4.53) we get:

$$\|\bar{r}_i\| \leq \sqrt{\frac{\tau_M(2-\beta)}{\beta}}. \tag{4.54}$$

Then:

$$\|[\bar{r}_{N-1}^T \ \dots \ \bar{r}_0^T]^T\| = \sqrt{\sum_{i=0}^{N-1}\bar{r}_i^T\bar{r}_i} \leq \sqrt{\frac{N\tau_M(2-\beta)}{\beta}}. \tag{4.55}$$

In the case of the Dynamic Watermarking detector, we have that $\sigma^{\tau_{\mathcal{D}}} = \sqrt{N(q+o)\epsilon}$, where $\epsilon > \ell - 1 - o - q$ is the solution to (4.31), since:

$$\|[\bar{r}_{N-1}^T \ \dots \ \bar{r}_0^T]^T\| = \sqrt{\sum_{i=0}^{N-1}\bar{r}_i^T\bar{r}_i} \leq \sqrt{N(q+o)\epsilon}. \tag{4.56}$$

Then, since:

$$\|[0_{o\times o(N-n)}\ \bar{A}_{n-1}]R_{N-1}\| \leq \|[0_{o\times o(N-n)}\ \bar{A}_{n-1}]\|\ \|R_{N-1}\|, \tag{4.57}$$

let $\eta = \|[0_{o\times o(N-n)}\ \bar{A}_{n-1}]\|\sigma^{\tau_*}$. Then:

$$\{\delta = [0_{o\times o(N-n)}\ \bar{A}_{n-1}]R_{N-1} \mid R_{N-1} \in \Theta_{N-1}^{\tau_*}\} \subset \mathcal{B}_\eta. \tag{4.58}$$

■

The program in Theorem 48 gives an upper bound to $\mathcal{R}_n^{\tau_*}$, which we denote by:

$$\mathcal{T}_n^{\tau_*} = \{\delta \mid \delta^T H_n^{\tau_*} \delta \leq 1\}. \tag{4.59}$$

We dilate $\mathcal{T}_n^{\tau_*}$ to obtain an outer approximation to $\mathcal{R}^{\tau_*}$:

Theorem 50. *[61, Theorem 5] Suppose $\tau_* \in \mathbb{R}$ such that $\tau_* > 0$, $\mathcal{R}^{\tau}$ is as in* (4.15)*, $\mathcal{T}_n^{\tau_*}$ is as in* (4.59)*, and:*

$$\mathcal{E}_n^{\tau_*} = \mathcal{T}_n^{\tau_*} \oplus \mathcal{B}_\epsilon, \tag{4.60}$$

where:

$$\epsilon = \frac{\|A^n\|}{\sqrt{s_1(H_n^{\tau_*})}(1 - \|A^n\|)}. \tag{4.61}$$

Then $\mathcal{R}^{\tau_*} \subset \mathcal{E}_n^{\tau_*}$.

To prove this result we must first consider the lemma:

Lemma 51. *[61, Lemma 3] Suppose $n, N, h \in \mathbb{N}$ such that $0 < n \leq h \leq N$, $R = [r_N^T ... r_0^T]^T \in \Theta_N^{\tau_*}$ where $\Theta_N^{\tau_*}$ is defined based on the choice of detector. Then $R' = [r_h^T ... r_{h-n}^T]^T \in \Theta_n^{\tau_*}$.*

Proof. **(Lemma 51)** To prove that R' is in $\Theta_n^{\tau_*}$ we show that each of the constraints associated with $\Theta_n^{\tau_*}$ are included as a constraint associated with $\Theta_N^{\tau_*}$ or that there exists a more restrictive constraint in $\Theta_N^{\tau_*}$. For the χ^2 test we have the inclusion of all constraints since using (4.20) and (4.21) we have:

$$R'^T Q_{(i,n)}^{\tau_{\chi^2}} R' = R^T Q_{(i+h-n,N)}^{\tau_{\chi^2}} R < 1 \ \forall i = 0, ..., n. \tag{4.62}$$

Similarly for the CUSUM detector we have that using (4.22) and (4.23) we have:

$$R'^T Q_{(i,j,n)}^{\tau_C} R' = R^T Q_{(i+h-n,j,N)}^{\tau_C} R < 1 \ \forall \ i = 0, ..., n \text{ and } j = 0, ..., i. \tag{4.63}$$

For the MEWMA we have that $\Theta_N^{\tau_*}$ has more restrictive constraints since using (4.27) and (4.28) we have:

$$R'^T Q_{(i,n)}^{\tau_M} R' \leq R^T Q_{(i+h-n,N)}^{\tau_M} R < 1 \ \forall i = 0, ..., n. \tag{4.64}$$

For The Dynamic Watermarking Detector we have the inclusion of all constraints since for (4.29)

and (4.30) we have:

$$R'^T Q^{\tau_{\mathcal{D}}}_{(i,n)} R' = R^T Q^{\tau_{\mathcal{D}}}_{(i+h-n,N)} R < 1 \; \forall i = \ell + \rho, ..., n. \tag{4.65}$$

■

Now we return to proving Theorem 50.

Proof. **(Theorem 50)** Let $\delta' \in \mathcal{R}^{\tau_*}$, and assume that $\delta' \notin \mathcal{E}^{\tau_*}_n$. Furthermore let:

$$\epsilon_1 = \inf\{\|\delta - \delta'\| \mid \delta \in \mathcal{E}^{\tau_*}_n\}. \tag{4.66}$$

Now consider that, for a given $N > n$:

$$\mathcal{R}^{\tau_*}_N \subseteq \{\delta \mid \delta = A_{N-1}X, \; X \in \Theta^{\tau_*}_{N-1}\}. \tag{4.67}$$

Using Minkowski sums we over-approximate this set further as:

$$\begin{aligned}
\mathcal{R}^{\tau_*}_N \subseteq & \{\delta = [\bar{A}_{n-1} \; 0_{p\times p(N-n)}]R_{N-1} \mid R_{N-1} \in \Theta^{\tau_*}_{N-1}\} \oplus \\
& \oplus \left(\bigoplus_{i=1}^{j} A^{ni}\{\delta = [0_{p\times pi} \; \bar{A}_{n-1} \; 0_{p\times p(N-n-i)}]R_{N-1} \mid R_{N-1} \in \Theta^{\tau_*}_{N-1}\} \right) \oplus \\
& \oplus A^{n(j+1)}\{\delta = [0_{p\times p(N-nj)} \; \bar{A}_h]R_{N-1} \mid R_{N-1} \in \Theta^{\tau_*}_{N-1}\}.
\end{aligned} \tag{4.68}$$

where N is evenly divisible by n, $j+1$ times and h is the remainder. Applying Lemma 49 and 51, we have:

$$\begin{aligned}
\mathcal{R}^{\tau_*}_N \subseteq & \{\delta = \bar{A}_{n-1}R_{n-1} \mid R_{n-1} \in \Theta^{\tau_*}_{n-1}\} \oplus \\
& \oplus \left(\bigoplus_{i=1}^{j} A^{ni}\{\delta = \bar{A}_{n-1}R_{n-1} \mid R_{n-1} \in \Theta^{\tau_*}_{n-1}\} \right) \oplus \\
& \oplus \mathcal{B}_{\eta\|A^{n(j+1)}\|}
\end{aligned} \tag{4.69}$$

where η is the maximum radius when applying Lemma 49 for $h = 0, ..., n$. Let $\sigma = \frac{1}{\sqrt{s_1(H^{\tau_*}_n)}}$ then:

$$\{\delta = \bar{A}_{n-1}R_{n-1} \mid R_{n-1} \in \Theta^{\tau_*}_{n-1}\} \subset \mathcal{T}^{\tau_*}_i \tag{4.70}$$

$$= \{\delta \mid \delta^T H^{\tau_*}_n \delta \leq 1\} \subset \mathcal{B}_\sigma. \tag{4.71}$$

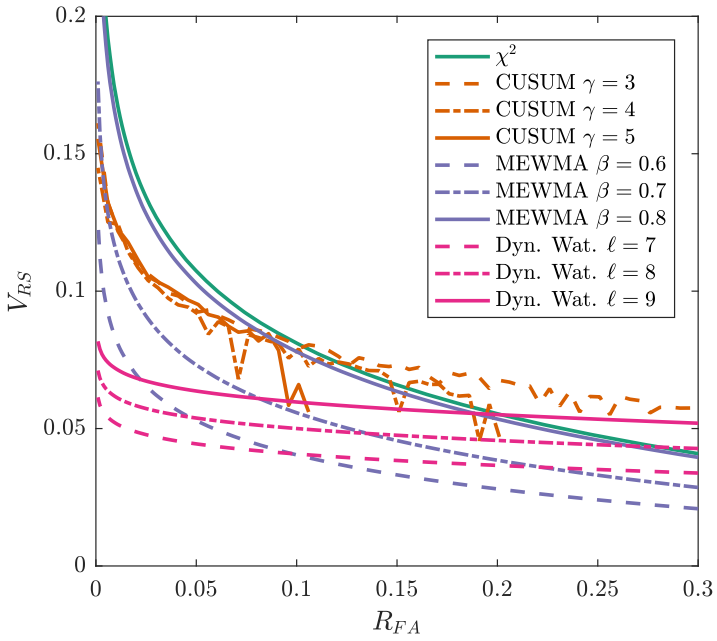


Figure 4.1: Approximate reachable set volume for varying false alarm rates for the example system in section 4.2.1

This means that:

$$\mathcal{R}_N^{\tau_*} \subseteq \mathcal{T}_n^{\tau_*} \oplus \left(\bigoplus_{i=1}^{j} \mathcal{B}_{\sigma\|A^{ni}\|} \right) \oplus \mathcal{B}_{\eta\|A^{n(j+1)}\|}. \tag{4.72}$$

Since the Minkowski sum of balls is a ball with its radius as the sum of the radii, we can increase the outer approximation by allowing the summation to extend towards infinity:

$$\mathcal{R}_N^{\tau_*} \subseteq \mathcal{T}_n^{\tau_*} \oplus \mathcal{B}_\epsilon \oplus \mathcal{B}_{\eta\|A^{n(j+1)}\|}, \tag{4.73}$$

where:

$$\epsilon = \frac{\sigma\|A^n\|}{(1 - \|A^n\|)} \geq \sum_{i=1}^{\infty} \sigma\|A^{ni}\|. \tag{4.74}$$

Since $j + 1 > \frac{N}{n}$, there exists an N_2 such that for $N > N_2$ we have that $\eta\|A^{n(j+1)}\| < \epsilon_1$ which contradicts $\delta \in \mathcal{R}^{\tau_*}$.

■

4.2.1 Simulation-Based Comparison of Attack Capability

To illustrate the trade-off between the rate of false alarms and attack capability, we provide a comparison of each of the detection algorithms using a 2 dimensional model from Murguia and Ruths [111]:

$$A = \begin{bmatrix} 0.84 & 0.23 \\ -0.47 & 0.12 \end{bmatrix} \qquad B = \begin{bmatrix} 0.07 & -0.32 \\ 0.23 & 0.58 \end{bmatrix}$$

$$C = \begin{bmatrix} 1 & 0 \\ 2 & 1 \end{bmatrix} \qquad K = \begin{bmatrix} 1.404 & -1.042 \\ 1.842 & 1.008 \end{bmatrix}$$

$$L = \begin{bmatrix} 0.0276 & 0.0448 \\ -0.01998 & -0.0290 \end{bmatrix} \qquad \Sigma_z = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}$$

$$\Sigma_w = \begin{bmatrix} 0.035 & -0.011 \\ -0.011 & 0.02 \end{bmatrix} \qquad \Sigma_r = \begin{bmatrix} 2.086 & 0.134 \\ 0.134 & 2.230 \end{bmatrix}$$

with the addition of a watermark with covariance $\Sigma_e = 10^{-2}I$. Thresholds for the false alarm rates between 0.01 and 0.3 were found by running the simulation under no attack for 10^6 time steps. Using these values, the reachable sets at time step $n = 12$ were outer approximated using the optimization program stated in Theorem 48 and dilated as stated in Theorem 50 to provide outer approximations of the steady state reachable sets. The resulting approximations for the V_{RS}, defined in (4.16), are plotted against the false alarm rate in Figure 4.1 for each of the detectors using various detector specific parameter selections.

One may note that while these methods provide smooth curves for the χ^2, MEWMA, and Dynamic Watermarking detectors, the curves for the CUSUM detector appears discontinuous, and do not span the entire range of R_{FA} values. The apparent discontinuity is attributed to the fact that, for the χ^2, MEWMA, and Dynamic Watermarking detectors, increasing the threshold τ_* results in a proportional scaling of the outer approximation of $\mathcal{R}_n^{\tau_*}$. However, For the CUSUM detector, changing the threshold does not have this affect. In fact, changing the threshold for the CUSUM detector alters the shape of $\Theta_n^{\tau_C}$, resulting in the outer approximation of $\mathcal{R}_n^{\tau_C}$ being less conservative for certain threshold values. Furthermore, the shortened span of the curves for the CUSUM detector are a result of certain R_{FA} values being un-achievable for a given forgetting factor.

To determine whether the outer approximation is tight for the χ^2, CUSUM, and MEWMA detectors, simulations were run for 60 R_{FA} values uniformly spaced between 0.01 and 0.3. In these simulations, the portion of the observer related to the residual was propagated forward for 10^5 steps using the dynamics (4.9). The residuals were sampled from a normal distribution with 0 mean and covariance $5I$ and scaled if necessary, to avoid alarms. The area of the convex hull of the observer

Method	R_{FA}	Threshold
χ^2	0.05	12.08
	0.03	14.29
	0.01	21.38
CUSUM γ=(15/17/19)	0.05	(51.04 / 12.27 / 2.21)
	0.03	(655.81 / 384.73 / 158.68)
	0.01	(1535.94 / 1445.58 / 1355.22)
MEWMA β=(0.6/0.7/0.8)	0.05	(13.09 / 15.00 / 16.85)
	0.03	(15.05 / 17.59 / 20.17)
	0.01	(20.68 / 24.85 / 28.88)
Dyn. Wat. ℓ=(20/25/30)	0.05	(99.57 / 103.74 / 105.59)
	0.03	(103.05 / 106.73 / 108.58)
	0.01	(108.58 / 110.79 / 113.94)

Table 4.1: Experimentally Found Thresholds for Various False Alarm Rates and Detector Specific Parameters for the Real-World Implementation

error for the entire simulation was calculated. The difference between the over approximated area and the simulated area ranged from $0.0156 - 0.1408$ for the χ^2, $0.0032 - 0.1143$ for the CUSUM, and $0.0075 - 0.1301$ for the MEWMA. The results indicate that the attack capability under the Dynamic Watermarking detector is comparable to the classic anomaly detectors as a function of false alarm rate.

4.3 Detect Specific Attacks

In this section, we evaluate the ability of each of the anomaly detection schemes to detect attacks using a Segway Robotics Mobility Platform performing a path-following task. In addition, we illustrate that adding a watermark to the system leads to an imperceptible reduction in performance, while significantly improving the detectability of an attack that was missed by classic anomaly detectors. Localization was provided by Google Cartographer [98] using planar LiDAR and wheel odometry measurements. For the purpose of control, a LTV model was fit to the observed data yielding:

$$\begin{bmatrix} e_{\ell,n+1} \\ e_{s,n+1} \\ e_{\theta,n+1} \\ e_{v,n+1} \\ e_{\dot{\theta},n+1} \end{bmatrix} = \begin{bmatrix} e_{\ell,n} + (0.0478\tilde{v}_n)e_{\theta,n} - (0.045\dot{\tilde{\theta}}_n)e_{s,n} \\ e_{s,n} + (0.0478)e_{v,n} + (0.045\dot{\tilde{\theta}}_n)e_{\ell,n} \\ e_{\theta,n} + 0.045e_{\dot{\theta},n} \\ e_{v,n} - 0.1e_{v,n-4} + 0.1u_{v,n} \\ 0.6e_{\dot{\theta},n} + 0.15e_{\dot{\theta},n-4} + 0.24u_{\dot{\theta},n} \end{bmatrix} \tag{4.75}$$

where the state is represented in trajectory error coordinates for a given nominal trajectory where $e_{\ell,n}, e_{s,n},$ and $e_{\theta,n}$ are the lateral, longitudinal, and heading error, $e_{v,n}, e_{\dot{\theta},n}$ are the error in the velocity and angular velocity, $\tilde{v}_n, \dot{\tilde{\theta}}_n$ are the nominal velocity and angular velocity and $u_{v,n}, u_{\dot{\theta},n}$ are the deviation from the nominal inputs.

For a constant nominal velocity of 0.6 m/s and angular velocity of 0 rad/s, this model can be represented as an LTI model with state vector:

$$x = \begin{bmatrix} e_{\ell,n} & e_{s,n} & e_{\theta,n} & e_{v,n} & e_{\dot{\theta},n} & e_{v,n-1} & e_{v,n-2} & e_{v,n-3} & e_{\dot{\theta},n-1} & e_{\dot{\theta},n-2} & e_{\dot{\theta},n-3} \end{bmatrix}^T. \tag{4.76}$$

For the sake of brevity, the A and B matrices are not stated explicitly but can be found by expanding (4.75). The feedback gain matrix K was found to make the closed loop system Schur Stable and is approximately:

$$K = \begin{bmatrix} 0 & 0 & -0.313 & -0.212 & 0 & 0.019 & 0.020 \\ -1.639 & -1.984 & 0 & 0 & -0.384 & 0 & 0 \\ 0.021 & 0.022 & 0 & 0 & 0 & 0 \\ 0 & 0 & -0.039 & -0.043 & -0.052 & -0.065 \end{bmatrix}^\intercal \tag{4.77}$$

Similarly the observer gain matrix L was found to make the observer Schur Stable and is approximately:

$$L = \begin{bmatrix} -0.791 & -0.016 & 0 & 0 & 0 \\ -0.002 & -0.501 & 0 & 0 & -0.022 \\ 0 & 0 & -0.272 & -0.025 & 0 \\ 0 & 0 & -0.011 & -0.252 & 0 \\ 0 & -0.003 & 0 & 0 & -0.240 \\ 0 & 0 & -0.013 & -0.258 & 0 \\ 0 & 0 & -0.015 & -0.187 & 0 \\ 0 & 0 & -0.015 & -0.133 & 0 \\ 0 & 0 & -0.014 & -0.091 & 0 \\ 0 & -0.004 & 0 & 0 & -0.395 \\ 0 & -0.010 & 0 & 0 & -0.145 \\ 0 & -0.008 & 0 & 0 & -0.054 \\ 0 & -0.005 & 0 & 0 & -0.024 \end{bmatrix} \tag{4.78}$$

The steady state covariance of the residuals, Σ_r, was approximated using the sample covariance from data generated by the Segway following a straight line down a 16 m hallway 40 times. To

avoid the effects of the transient behavior at the start of each run, the beginning of each run was ignored. This experiment was repeated a second time after the introduction of a watermark into the control input with covariance:

$$\Sigma_e = \begin{bmatrix} 0.02 & 0 \\ 0 & 0.03 \end{bmatrix}, \tag{4.79}$$

in order to approximate Σ_ψ. The average location error was 0.0262 m for the non-watermarked runs and 0.0506 m for the watermarked runs. While adding the watermark increased the location error, the average error did not hinder overall performance during the lane-following task.

Threshold values for the false alarm rates of 0.01, 0.03, and 0.05 were approximated for the χ^2, CUSUM, and MEWMA detectors using the residuals from the un-watermarked runs. Thresholds for the Dynamic Watermarking detector and the same false alarm rates were found using the residuals from the watermarked runs. The resulting threshold values are displayed in Table 4.1.

Two attacks, following differing models, were then applied. Attack model 1 assumes that the attacker adds random noise to the system such that $v_n \sim \mathcal{N}(0, 10^{-5}I)$. Attack model 2 takes the form:

$$v_n = -(Cx_n + z_n) + C\xi_n + \zeta_n. \tag{4.80}$$

For this model, the attack measurement noise ζ_n is added to the false state such that $\zeta_n \sim \mathcal{N}(0, \Sigma_\zeta)$ and the false state $\xi_n \in \mathbb{R}^p$ is updated according to the closed loop dynamics of the system:

$$\xi_{n+1} = (A + BK)\xi_n + \omega_n \tag{4.81}$$

with attack process noise $\omega_n \sim \mathcal{N}(0, \Sigma_\omega)$. The attack process and measurement noise were chosen to leave the distribution of the residuals unchanged.

For each attack, 10 experimental runs were completed without a watermark and 10 with a watermark for a total of 20 experimental runs per attack model or 40 total experimental runs. The runs with a watermark were used in evaluating the Dynamic Watermarking detector, while all other detectors used the un-watermarked data. The resulting detection rates, defined as the number of alarms divided by the total number of time steps in the attacked runs, are displayed in Table 4.2.

For attack model 1, all of the detectors are able to reliably detect the attack, confirming that the implementation of the detectors is correct. For attack model 2 the detection rate decrease from the R_{FA} for the χ^2, CUSUM, and MEWMA detectors. This may be due to the residuals for the un-attacked system not being distributed as a Gaussian distribution resulting in higher threshold values. Since attack model 2 replaces the feedback completely, the resulting residuals, when under

Method	R_{FA}	Attack Model 1 Detection Rates	Attack Model 2 Detection Rates
χ^2	0.05	0.69	0.03
	0.03	0.62	0.01
	0.01	0.47	0.00
CUSUM γ=(15/17/19)	0.05	(0.98 / 0.99 / 0.99)	(0.00 / 0.00 / 0.00)
	0.03	(0.81 / 0.86 / 0.92)	(0.00 / 0.00 / 0.00)
	0.01	(0.58 / 0.54 / 0.51)	(0.00 / 0.00 / 0.00)
MEWMA β=(0.6/0.7/0.8)	0.05	(0.51 / 0.57 / 0.62)	(0.03 / 0.03 / 0.03)
	0.03	(0.45 / 0.50 / 0.54)	(0.01 / 0.01 / 0.01)
	0.01	(0.32 / 0.35 / 0.39)	(0.00 / 0.00 / 0.00)
Dyn. Wat. ℓ=(20/25/30)	0.05	(**1.00** / **1.00** / **1.00**)	(0.98 / **1.00** / **1.00**)
	0.03	(**1.00** / **1.00** / **1.00**)	(0.97 / 0.99 / **1.00**)
	0.01	(**1.00** / **1.00** / **1.00**)	(0.95 / 0.98 / **1.00**)

Table 4.2: Experimentally Found Alarm Rates For Various Detector Specific Parameters

attack, do follow a Gaussian distribution which then results in lower detection rates. The Dynamic Watermarking detector in the presence of the second attack provides a high detection rates for each set of parameters, and in some cases achieves a perfect detection rate.

Chapter 5

Single Autonomous Vehicle Applications

CAVs have been touted as a way to increase safety by removing driver error. However, like other CPSs, CAVs are vulnerable to cyber-attack, that give rise to additional safety concerns [15]–[17]. Some cyber-attacks can even compromise the control systems of the CAV.

In this chapter we illustrate how LTV dynamic watermarking, as described in Section 3.2, can be applied to a single CAV to detect altered measurement signals. To this end, we consider two case studies. First, we use CarSim to simulate a high fidelity vehicle model in Section 5.1. For this system, we fit a lower fidelity model then linearize to generate an LTV model. While dynamic watermarking is applied using the LTV model, the simulation is still run on using CarSim and the high fidelity model. Second, we use a $1/10^{\text{th}}$ scale autonomous rover shown in Figure 5.3 to illustrate the effectiveness of dynamic watermarking on a real-world system.

5.1 CarSim Example

In our simulations of a single autonomous vehicle we use the simulation environment CarSim with a high fidelity non-linear model [118]. CarSim is a widely used software for accurately modeling the behavior of vehicles using high dimensional multi-body dynamics. As such, we treat the output of the simulation as the ground truth. However, to make linearization tractable we fit a simplified vehicle model.

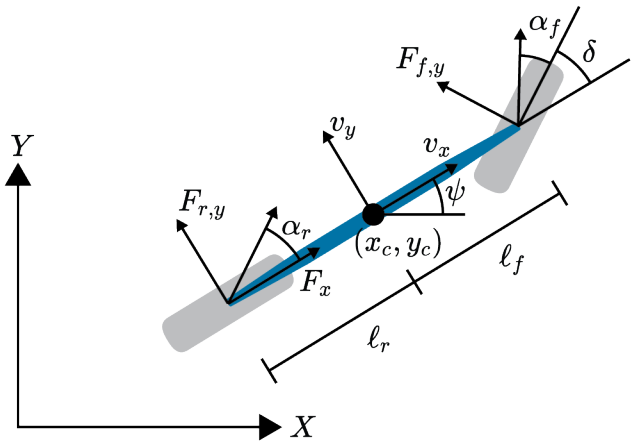


Figure 5.1: Bicycle model with tire forces and slip angles used to approximate dynamics of CarSim simulation

5.1.1 Vehicle Model

We start with the dynamic bicycle model illustrated in Figure 5.1. This model satisfies Liniger *et al.* [119, Equation (1)]:

$$\begin{bmatrix} \dot{x}_c \\ \dot{y}_c \\ \dot{\psi} \\ \dot{v}_x \\ \dot{v}_y \\ \ddot{\psi} \end{bmatrix} = \begin{bmatrix} v_x \cos(\psi) - v_y \sin(\psi) \\ v_x \sin(\psi) + v_y \cos(\psi) \\ \dot{\psi} \\ \frac{1}{m} F_x - \frac{1}{m} F_{f,y} \sin(\delta) + v_y \dot{\psi} \\ \frac{1}{m} F_{f,y} \cos(\delta) + \frac{1}{m} F_{r,y} - v_x \dot{\psi} \\ \frac{\ell_f}{I_z} F_{f,y} \cos(\delta) - \frac{\ell_r}{I_z} F_{r,y} \end{bmatrix} \tag{5.1}$$

where (x_c, y_c) are the ground plane coordinates of the center of mass, ϕ is the heading angle, v_x is the longitudinal velocity, v_y is the lateral velocity, $\dot{\psi}$ is the angular velocity, m is the mass, I_z is the moment of inertia, ℓ_f and ℓ_r are the distances from the center of mass to the front and rear tire respectively, δ is the steering angle, F_x is the longitudinal force at the back tire, and $F_{f,y}$ and $F_{r,y}$ are the lateral forces at the front and rear tire respectively.

For (5.1), the longitudinal force F_x is modeled empirically using a 4^{th} order polynomial in terms of the desired throttle u_a, and braking u_d inputs

$$F_x = \sum_{\substack{i,j \in \{0,1,2,3,4\} \\ i+j \leq 4}} v_x^i \left(c_{a,i,j} u_a^j + c_{d,i,j} u_d^j \right). \tag{5.2}$$

Note that we assume that the throttle and braking inputs cannot both be non-zero at any given time. The Lateral forces are then approximated using a simplified version of the widely used

Pacejka "Magic" tire model [120][119, Equations 2(a), 2(b)]

$$F_{f,y} = F_{f,z} D_f \sin(C_f \arctan(B_f \alpha_f)), \tag{5.3}$$

$$F_{r,y} = F_{r,z} D_r \sin(C_r \arctan(B_r \alpha_r)), \tag{5.4}$$

where the front and rear slip angles α_f and α_r satisfy

$$\alpha_f = \delta - \arctan\left(\frac{\ell_f \dot{\psi} + v_y}{v_x}\right), \tag{5.5}$$

$$\alpha_r = \arctan\left(\frac{v_y - \ell_r \dot{\psi}}{v_x}\right), \tag{5.6}$$

and the front and rear vertical tire force $F_{f,z}$ and $F_{r,z}$ satisfy

$$F_{f,z} = \frac{mg\ell_r}{\ell_f + \ell_r}, \tag{5.7}$$

$$F_{r,z} = \frac{mg\ell_f}{\ell_f + \ell_r}. \tag{5.8}$$

Though (5.1) and the corresponding simulation environment treats the steering angle directly as an input, a real-world vehicle cannot change the steering angle instantly. Therefore we instead treat δ as a state of the system and use the rate of change of the steering angle $\dot{\delta}$ as an input.

To linearize the system let $x = \begin{bmatrix} x_c & y_c & \psi & v_x & v_y & \dot{\psi} \end{bmatrix}^\intercal$, $u = \begin{bmatrix} u_a & u_b & \dot{\delta} \end{bmatrix}^\intercal$, and let f be a function such that $\dot{x} = f(x, u)$. This model is then linearized about the desired trajectory $\bar{x}$ with corresponding nominal inputs $\bar{u}$ such that

$$\dot{x} - \dot{\bar{x}} = \left.\frac{\partial f}{\partial x}\right|_{\substack{x=\bar{x} \\ u=\bar{u}}} (x - \bar{x}) + \left.\frac{\partial f}{\partial u}\right|_{\substack{x=\bar{x} \\ u=\bar{u}}} (u - \bar{u}). \tag{5.9}$$

Then we discretize the system for a time step of 0.05 s under the assumption of a zero order hold on the inputs and states to generate an LTV model of the form ($\mathcal{D}2$) where

$$A_n = \operatorname{expm}\left(0.05 \left.\frac{\partial f}{\partial x}\right|_{\substack{x_n=\bar{x_n} \\ u_n=\bar{u_n}}}\right), \tag{5.10}$$

$$B_n = \int_{t=0}^{0.05} \operatorname{expm}\left(t \left.\frac{\partial f}{\partial x}\right|_{\substack{x_n=\bar{x_n} \\ u_n=\bar{u_n}}}\right) dt, \tag{5.11}$$

and the state and inputs of the discretized system are the deviation from the nominal trajectory and inputs.

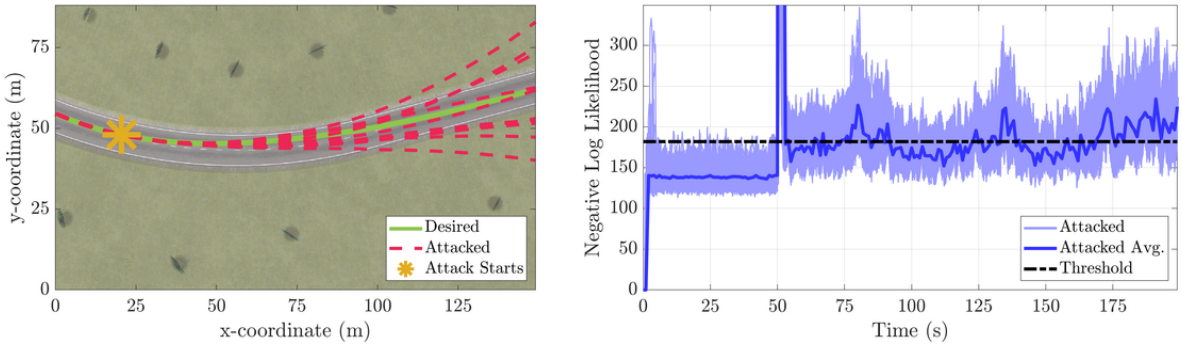


Figure 5.2: The simulated high fidelity car is attacked with a replay attack after 50 s of operation. The desired trajectory and 10 attacked realizations are plotted for the region that the attack is initiated (Left). Negative log likelihood for all 200 attacked realizations with average value are plotted (Right).

5.1.2 Results

This section illustrates the effectiveness of LTV dynamic watermarking using a high fidelity vehicle model in CarSim. For the simulation, the vehicle completes a 1,137 m long trajectory traveling at speeds up to 7 m/s in approximately 200 s. This is accomplished using a linear quadratic regulator (LQR) and a linearization of the car model (5.1)-(5.8). The simulated measurement signal at step n includes the ground plane coordinates $(x_{c,n}, y_{c,n})$ in meters, heading ψ_n in radians, longitudinal velocity $v_{1,n}$ in meters per second, lateral velocity $v_{2,n}$ in meters per second, yaw rate $\dot{\psi}_n$ in radians per second, and steering wheel angle δ_n in radians. Since the feedback from the simulation does not include noise, Gaussian measurement noise was added to the measurement such that when no attack is present

$$y_n = \begin{bmatrix} x_{c,n} & y_{c,n} & \psi_n & v_{1,n} & v_{2,n} & \dot{\psi}_n & \delta_n \end{bmatrix}^{\mathsf{T}} - \begin{bmatrix} \bar{x}_{c,n} & \bar{y}_{c,n} & \bar{\psi}_n & \bar{v}_{1,n} & \bar{v}_{2,n} & \dot{\bar{\psi}}_n & \bar{\delta}_n \end{bmatrix}^{\mathsf{T}} + z_n, \tag{5.12}$$

where $z_n \sim \mathcal{N}\left(0_{7\times 1}, 1 \times 10^{-8} I_7\right)$. The control signal sent to the simulation includes percent throttle u, steering wheel rate $\dot{\delta}$ in radians per second. A watermark with covariance

$$\Sigma_e = 0.015 I_2 \tag{5.13}$$

was added to the control input at each step. The watermark covariance was chosen to minimize degradation in system performance while also being sufficiently large to aid in detection. The matrix normalizing factor and the auto-correlation normalizing factor were generated from 200 realizations using (3.174) and (3.175). The window size ℓ of 21 steps was used for the statistical tests. For this window size, a threshold of 181.94 was used based on a false alarm rate of 0.002 for the un-attacked trials.

To generate a replay attack, the measurement signal from one run is recorded and then played

Figure 5.3: The $1/10^{\text{th}}$ scale autonomous rover used in real-world testing of LTV dynamic watermarking on a single autonomous vehicle.

back when the simulation is run for a separate realization. Since an attack need not start at the beginning, we chose to start the attack 50s after the start of the simulation. Furthermore, since the initial replayed measurement may be inconsistent with what is expected given the current observed state of the system, the attacked measurement instead was linearly interpolated between the true measurement and the replayed measurement over the course of 0.15s.

In practice, an autonomous vehicle would respond to the detection of an attack. We instead allowed the vehicle to continue normal operation up to a certain distance from the desired trajectory. This allows us to illustrate the results of a replay attack on an autonomous vehicle.

The results of these simulations can be seen in Figure 5.2. The left side of the figure shows the results of the replay attack on our high fidelity car model. The right side of the figure shows the ability of LTV Dynamic Watermarking to detect these attacks. Note, despite our attempt to smooth the transition to the replayed attack the negative log likelihood has a spike immediately following the start of the attack at 50 s. Moreover, the negative log likelihood continues to exceed the threshold as the attack continues and the transient effect of the transition diminishes.

5.2 Rover Example

In our real-world experiments on a single autonomous vehicle, we use the $1/10^{\text{th}}$ scale autonomous rover illustrated in Figure 5.3. The design of this rover was based on the MIT-racecar project [121] with some modifications. The rover is outfitted with a motor speed sensor, 3-axis accelerometer, 3-axis gyroscope, 3-axis magnetometer, planar lidar scanner, and stereo camera. Low level control is handled by a speed and servo controller while the high level control and communications are executed by a Nvidia Jetson TX2. Due to our testing location we chose to also make use of a motion capture system for ground plane coordinates.

5.2.1 Vehicle Model

We start with the empirically found non-linear model

$$\begin{bmatrix} \dot{x}_c \\ \dot{y}_c \\ \dot{\psi} \\ \dot{v} \end{bmatrix} = \begin{bmatrix} v\cos\psi - \dot{\psi}(c_8 + c_9 v^2)\sin\psi \\ v\sin\psi + \dot{\psi}(c_8 + c_9 v^2)\cos\psi \\ \frac{\tan(c_1\delta + c_2)v}{c_3 + c_4 v^2} \\ c_5 + c_6(v - v^d) + c_7(v - v^d)^2 \end{bmatrix}, \tag{5.14}$$

where (x_c, y_c) are the ground plane coordinates, ψ is the vehicle heading, v is the forward velocity, δ is the desired steering angle, v^d is the desired velocity, and $c_1, \ldots, c_9$ are fitted constants which can be found in Table 5.1. This model attempts to capture both the non-linear dynamics of the

Constant	Value
c_1	1.6615×10^{-5}
c_2	-1.9555×10^{-7}
c_3	3.619×10^{-6}
c_4	4.382×10^{-7}
c_5	-8.1112×10^{-2}
c_6	-1.4736×10^{0}
c_7	1.2569×10^{-1}
c_8	7.6459×10^{-2}
c_9	-1.3991×10^{-2}

Table 5.1: Fitted constants for the nonlinear dynamics in Eq. (5.14)

power-train and various dynamic effects such as tire slip. Namely, the constants c_1 and c_2 are used to calibrate the steering angle, c_4 and c_9 are used to model the effect of tire slip on the angular and lateral velocities, and c_5, c_6, and c_7 are used to approximate the drive train. Note that when $c_4 = 0$ and $c_9 = 0$ the equations for $\dot{x}_c$, $\dot{y}_c$, and $\dot{\psi}$ follow from [122, Table 2.1] for a proper selection of c_3

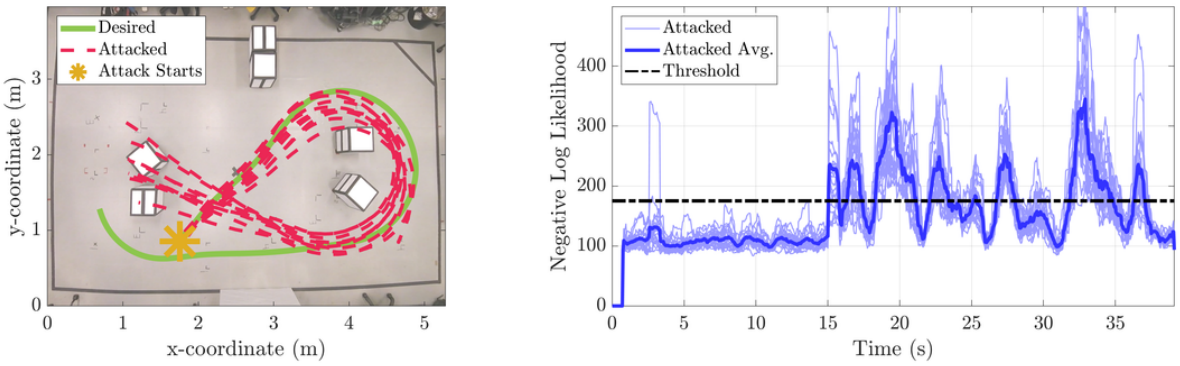


Figure 5.4: The 1/10 scale autonomous rover is attacked with a replay attack after 15 s of operation. The desired trajectory and 10 attacked realizations are plotted for the region that the attack is initiated (Left). Negative log likelihood for all 20 attacked realizations with average value are plotted (Right).

and c_4.

The dynamics in (5.14) are linearized and discretized for a time step of 0.05 s in much the same way as in (5.9)-(5.11) of the CarSim example.

5.2.2 Results

This section further illustrates the effectiveness of LTV Dynamic Watermarking on a 1/10 scale autonomous rover. For the experiment, the rover completes a lap around a track consisting of several turns and changes in velocity. The track has a length of 38.8 m and the rover travels at speeds up to 1.8 m/s. This is accomplished using a LQR and a linearized rover model. The measurement signal at step n includes the ground plane coordinates $(x_{c,n}, y_{c,n})$ in meters, heading ψ_n in radians, angular velocity $\dot{\psi}_n$ in radians per second, and longitudinal velocity v_n in meters per second. The ground plane coordinates and heading are measured using a motion capture system, the angular velocity is measured by an IMU, and the longitudinal velocity is measured by the motor controller. The control signal includes a desired speed in meters per second and a steering angle in radians. A watermark with covariance

$$\Sigma_e = \begin{bmatrix} 0.02 & 0 \\ 0 & 0.005 \end{bmatrix} \tag{5.15}$$

was added to the control input at each step. The matrix normalizing factor and the auto-correlation normalizing factor were generated from 100 experimental runs using (3.174) and (3.175). The window size of 15 steps ($\ell = 14$) was used for the statistical tests. For this window size, a threshold of 175.28 was used based on a false alarm rate of 0.002 for the un-attacked trials.

Implementation of the replay attack was done in the same fashion as was done in simulation except the attack was initiated at 15 s. For safety purposes, the rover is remotely stopped when the

attack causes it to leave the track area.

The results of these experiments can be seen in Figure 5.4. Similar to the simulated results, the left side of the figure shows the results of the replay attack on the 1/10 scale autonomous rover. Furthermore, the right of the figure shows the ability of LTV Dynamic Watermarking to detect these attacks. Note, the transition to the replayed measurements has a lesser effect on the negative log likelihood. Nonetheless, the negative log likelihood continues to exceed the threshold as the attack continues ensuring detection.

Chapter 6

Autonomous Platoon Applications

Vehicle platooning has been shown to decrease fuel consumption by reducing air drag [123]–[126], while also improving throughput on roads by reducing the occurrence of bottlenecks [127], [128]. To achieve these performance improvements without creating phantom traffic jams or crashes [129], [130], the longitudinal controller for these vehicle platoons must be *string stable* meaning that perturbations must be dampened by subsequent vehicles in the string [131]–[135].

For platoons that do not rely on V2V communication, string stability can be satisfied by maintaining a constant headway between vehicles in the platoon [136]–[138]. Unfortunately, since the headway is found by dividing the bumper-to-bumper following distance by the speed, this leads to conservatively sized gaps between vehicles that grow as the speed increases. Since the reduction in air drag becomes less prominent as the following distance increases, the methods that avoid utilizing V2V communication are unable to achieve all the potential energy efficiency benefits of vehicle platooning.

In contrast, for connected vehicles, the additional road user information can be used to adopt a constant spacing policy while still preserving string stability. Even under limited V2V communication with just one or two neighboring vehicles [139]–[144], energy efficiency and throughput can increase dramatically due to the reduction in following distance. However, these communication channels introduce vulnerabilities to cyber attacks.

In this chapter we illustrate the application of LTV distributed dynamic watermarking as described in Section 3.3 to a simulated platoon of autonomous vehicles. First, in Section 6.1, we define the particular model used to describe the dynamics of the platoon and explain how the assumptions in Subsection 3.3.1 are satisfied. Then, in Section 6.2, the details of the simulations and results are discussed. A video of the simulations is available at Porter *et al.* [145].

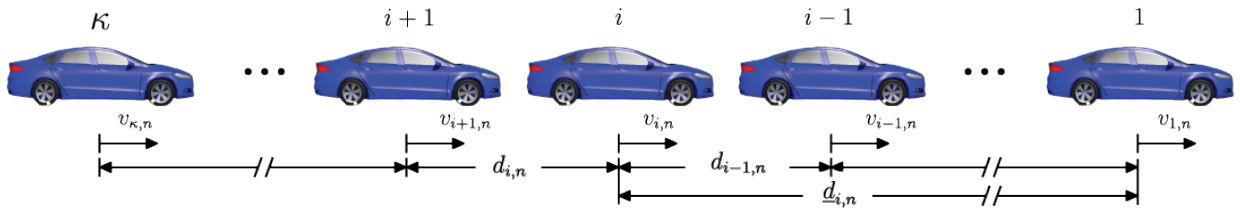


Figure 6.1: The platooning state at step n is described by the velocity of each vehicle $v_{1,n}, \ldots, v_{\kappa,n}$ and the distances between vehicles $d_{1,n}, \ldots, d_{\kappa-1,n}$. The distance from a vehicle in an arbitrary position i to the lead vehicle is $\underline{d}_{i,n}$.

6.1 Platoon Model

This section illustrates the ability of an LTV system model with distributed control satisfying ($\mathcal{D}3$) in Section 3.3 to describe a vehicle platoon. A thorough derivation is provided and the resulting system is presented. Whereas previous literature in platooning has often assumed a constant velocity, the use of an LTV system model can describe platooning with fewer assumptions.

The task of vehicle platooning is often broken into two components, lane keeping and vehicle following [146]. The objective of lane keeping is to minimize the *lateral error* i.e. the distance to the nearest point on the trajectory. The task of vehicle following attempts to maintain a constant distance or a constant headway to the preceding vehicle in the platoon. Note that the lead vehicle attempts to maintain a minimum headway when near other vehicles and a desired velocity otherwise. In this example, we focus on the task of vehicle following since it requires V2V communication to reduce following distances between vehicles. However, we also describe our lateral dynamics for the sake of completeness.

We start with the same empirically found non-linear model (5.14) from the single autonomous vehicle example in Section 5.2.

The state of the longitudinal dynamics for a platoon of κ vehicles, as illustrated in Figure 6.1, consists of each vehicle's velocity $v_{1,n}, \ldots, v_{\kappa,n}$, and the distances between subsequent vehicles $d_{1,n}, \ldots, d_{\kappa-1,n}$. For convenience we define

$$\underline{d}_{i,n} = \sum_{j=1}^{i-1} d_{i,n} \tag{6.1}$$

to denote the distance from the lead vehicle to vehicle i in the platoon. In this paper, the distance between vehicles is measured along the trajectory based on the center of each vehicle. This simplifies notation while still allowing for vehicle length to be accounted for in the platoons desired trajectory.

Lateral Dynamics

Consider the lateral error Δlat_i and heading error $\Delta\psi_i$ defined as

$$\Delta\text{lat}_i = (y_i - \bar{y}_i)\cos(\bar{\psi}_i) - (x_i - \bar{x}_i)\sin(\bar{\psi}_i) \tag{6.2}$$

and

$$\Delta\psi_i = \psi_i - \bar{\psi}_i \tag{6.3}$$

We approximate the continuous dynamics as follows.

$$\Delta\dot{\psi}_i = \dot{\psi} - \dot{\bar{\psi}} = \frac{\tan(c_1\delta_i + c_2)v_i}{c_1 + c_4 v_i^2} - \frac{\tan(c_1\bar{\delta}_i + c_2)\bar{v}_i}{c_3 + c_4\bar{v}_i^2} \approx \frac{(\tan(c_1\delta_i + c_2) - \tan(c_1\bar{\delta}_i + c_2))\bar{v}_i}{c_3 + c_4\bar{v}_i^2}, \tag{6.4}$$

where the second equality comes from (5.14) and the approximation from $v_i \approx \bar{v}_i$ and

$$\begin{aligned}
&\Delta\dot{\text{lat}}_i(\dot{y}_i - \dot{\bar{y}}_i)\cos(\bar{\psi}_i) - (\dot{x}_i - \dot{\bar{x}}_i)\sin(\bar{\psi}_i) - \dot{\bar{\psi}}_i\left((x_i\bar{x}_i)\cos(\bar{\psi}_i) + (y)_i\bar{y}_i)\sin(\bar{\psi}_i)\right) = \\
&\quad\approx v_i\sin(\Delta\psi_i) + \dot{\psi}_i(c_8 + c_9 v_i^2)\cos(\Delta\psi_i) - \dot{\bar{\psi}}_i(c_8 + c_9\bar{v}_i^2) = \\
&\quad\approx \bar{v}_i\sin(\Delta\psi_i) + \frac{(c_8 + c_9\bar{v}_i^2)\bar{v}_i}{c_3 + c_4\bar{v}_i^2}(\tan(c_1\delta_i + c_2)\cos(\Delta\psi_i) - \tan(c_1\bar{\delta}_i + c_2)),
\end{aligned} \tag{6.5}$$

where the first inequality comes from (5.14) and $(x_i - \bar{x}_i)\cos(\bar{\psi}_i) + (y_i - \bar{y}_i)\sin(\bar{\psi}_i) \approx 0$, and the second from (5.14) and $v_i \approx \bar{v}_i$. These approximations are reasonable since the longitudinal controller aims to maintain the desired velocity and we reduce longitudinal error defined as $(x_i - \bar{x}_i)\cos(\bar{\psi}_i) + (y_i - \bar{y}_i)\sin(\bar{\psi}_i)$ by accommoding drift along the trajectory. Linearizing (6.4)-(6.5) then gives us

$$\begin{bmatrix}\Delta\dot{\text{lat}}_i \\ \Delta\dot{\psi}_i\end{bmatrix} = \begin{bmatrix}0 & \bar{v}_i \\ 0 & 0\end{bmatrix}\begin{bmatrix}\Delta\text{lat}_i \\ \Delta\psi_i\end{bmatrix} + \frac{c_1\bar{v}_i}{\cos^2(c_1\bar{\delta}_i + c_2)(c_3 + c_4\bar{v}_i^2)}\begin{bmatrix}(c_8 + c_9\bar{v}_i^2) \\ 1\end{bmatrix}\Delta\delta_i, \tag{6.6}$$

where $\Delta\delta_i = \delta_i - \bar{\delta}_i$. Discretizing using a step size of 0.05 s and a zero-order hold on $\bar{v}_i$ and δ_i then results in

$$\begin{bmatrix}\Delta\text{lat}_{i,n+1} \\ \Delta\psi_{i,n+1}\end{bmatrix} = \begin{bmatrix}1 & \frac{\bar{v}_{i,n}}{20} \\ 0 & 1\end{bmatrix}\begin{bmatrix}\Delta\text{lat}_{i,n} \\ \Delta\psi_{i,n}\end{bmatrix} + \frac{c_1\bar{v}_{i,n}}{\cos^2(c_1\bar{\delta}_{i,n} + c_2)(c_3 + c_4\bar{v}_{i,n}^2)}\begin{bmatrix}\left(\frac{(c_8 + c_9\bar{v}_{i,n}^2)}{20} + \frac{\bar{v}_{i,n}}{800}\right) \\ \frac{1}{20}\end{bmatrix}\Delta\delta_{i,n}. \tag{6.7}$$

Longitudinal Dynamics

The state of the longitudinal dynamics for a platoon of κ vehicles, as illustrated in Figure 6.1, is made up of each vehicle's velocity $v_{1,n}, \ldots, v_{\kappa,n}$, and the distances between subsequent vehicles $d_{1,n}, \ldots, d_{\kappa-1,n}$. Next, under the assumption that the tracking error for the lane keeping controller is sufficiently small, we linearize the longitudinal dynamics from (5.14) as

$$\dot{\Delta d_i} = \Delta v_i - \Delta v_{i+1} \tag{6.8}$$

$$\dot{\Delta v_i} = \alpha_i \Delta v_i - \alpha_i \Delta v_i^d, \tag{6.9}$$

where $\Delta d_i = d_i - \bar{d}_i$, $\Delta v_i = v_i - \bar{v}_i$, $\Delta v_i^d = v_i^d - \bar{v}_i^d$, and α_i is the continuous equivalent to (6.13) defined as

$$\alpha_i = c_6 + 2c_7(\bar{v}_i - \bar{v}_i^d). \tag{6.10}$$

Selecting a time step of 0.05 s and assuming a zero-order hold on the input, these dynamics are then discretized as

$$\Delta d_{i,n+1} = \Delta d_{i,n} + \frac{\beta_i - 1}{\alpha_i}\Delta v_{i,n} - \frac{\beta_{i+1} - 1}{\alpha_{i+1}}\Delta v_{i+1,n} - (\frac{\beta_i - 1}{\alpha_i} - 0.05)\Delta v_i^d + (\frac{\beta_{i+1} - 1}{\alpha_{i+1}} - 0.05)\Delta v_{i+1}^d \tag{6.11}$$

$$\Delta v_{i,n+1} = \beta_i \Delta v_{i,n} - (\beta_i - 1)\Delta v_{i,n}^d, \tag{6.12}$$

where

$$\alpha_{i,n} = c_6 + 2c_7(\bar{v}_{i,n} - \bar{v}_{i,n}^d), \tag{6.13}$$

$$\beta_{i,n} = e^{0.05\alpha_{i,n}}. \tag{6.14}$$

Vectorizing these discrete dynamics for the state vector $x_n = [\Delta d_{1,n} \cdots \Delta d_{\kappa-1,n}\, \Delta v_{1,n} \cdots \Delta v_{\kappa,n}]^\intercal$ and inputs $u_{i,n} = \Delta v_{i,n}^d$ results in an LTV system satisfying

$$x_{n+1} = A_n x_n + \sum_{i=1}^{\kappa} B_{i,n}(K_{i,n}\hat{x}_{i,n} + e_{i,n}) + w_n, \tag{6.15}$$

where $K_{i,n}$, $\hat{x}_{i,n}$, and $e_{i,n}$ are vehicle i's state feedback, observed state, and watermark respectively

and

$$A_n = \left[\begin{array}{c:ccccc} & \frac{\beta_{1,n}-1}{\alpha_{1,n}} & -\frac{\beta_{2,n}-1}{\alpha_{2,n}} & & & \\ I_{\kappa-1} & & \ddots & \ddots & & \\ & & & & \frac{\beta_{\kappa-1,n}-1}{\alpha_{\kappa-1,n}} & -\frac{\beta_{\kappa,n}-1}{\alpha_{\kappa,n}} \\ \hdashline 0_{\kappa\times\kappa-1} & \multicolumn{5}{c}{\operatorname{diag}(\beta_{1,n},\ldots,\beta_{\kappa,n})} \end{array}\right], \tag{6.16}$$

$$B_{i,n} = \begin{cases} \left[\frac{1}{20} - \frac{\beta_{1,n}-1}{\alpha_{1,n}} \quad 0_{1\times\kappa-2} \quad 1-\beta_{1,n} \quad 0_{1\times\kappa-1}\right]^\intercal & i = 1 \\ \left[0_{1\times i-2} \quad \frac{\beta_{i,n}-1}{\alpha_{i,n}} - \frac{1}{20} \quad \frac{1}{20} - \frac{\beta_{i,n}-1}{\alpha_{i,n}} \quad 0_{1\times\kappa-2} \quad 1-\beta_{i,n} \quad 0_{1\times\kappa-i}\right]^\intercal & i \neq 1. \end{cases} \tag{6.17}$$

We assume each vehicle is able to measure its own location, velocity, and the distance to the previous vehicle in the platoon. Using the location measurements, the lateral error and heading error are calculated for the lane keeping task. For the vehicle following task the measurements for vehicle i at step n satisfy

$$y_{i,n} = C_i x_n + z_{i,n},\ i \in \{1,\ldots,\kappa\}, \tag{6.18}$$

where $z_{i,n}$ is the measurement noise and $C_{i,n}$ takes one of two forms depending on the choice of controller. In the first case, the leader measures its own velocity, while each of the other vehicles measures both their own velocity and the distance to the preceding vehicle

$$C_i = \begin{cases} \left[0_{1\times\kappa-1} \quad 1 \quad 0_{1\times\kappa-1}\right] & i = 1 \\ \left[\begin{array}{c:c:c:c:c} & 1 & & 0 & \\ 0_{2\times i-2} & & 0_{2\times\kappa-1} & & 0_{2\times\kappa-i} \\ & 0 & & 1 & \end{array}\right] & i \neq 1. \end{cases} \tag{6.19}$$

In the second case, we assume that the first vehicle also communicates its location along the trajectory which can be used by other vehicles to calculate the distance to the lead vehicle. However,

we model this as the first vehicle also measuring the distance to all following vehicles

$$C_i = \begin{cases} \left[\begin{array}{cccc:c:c} 1 & 0 & \dots & 0 & & \\ \vdots & \ddots & \ddots & \vdots & 0_{\kappa-1\times 1} & 0_{\kappa-1} \\ 1 & \dots & 1 & 0 & & \\ 1 & \dots & 1 & 1 & & \\ \hdashline & 0_{1\times\kappa-1} & & & 1 & 0_{1\times\kappa-1} \end{array}\right] & i = 1 \\ \left[\begin{array}{c:c:c:c:c} 0_{2\times i-2} & \begin{matrix}1\\0\end{matrix} & 0_{2\times\kappa-1} & \begin{matrix}0\\1\end{matrix} & 0_{2\times\kappa-i} \end{array}\right] & i \neq 1. \end{cases} \tag{6.20}$$

Since the vehicle following task only seeks to maintain a desired velocity and spacing policy, any longitudinal error with respect to a predefined trajectory will not be corrected by the controller. As a result, the position of the platoon can drift along the trajectory. Though this drift does not affect the derivations in this section, it does pose a practical problem which we resolve in Section 3.4.

6.1.1 Lateral Controller and Observer

For lane keeping, a lateral controller is introduced which operates independently of the longitudinal controller (i.e. each vehicle runs the same lateral controller and observer at all times). The feedback law follows

$$\delta_{i,n} = \begin{bmatrix} -0.25 & -1 \end{bmatrix} \begin{bmatrix} \Delta\hat{\text{lat}}_{i,n} \\ \Delta\hat{\psi}_{i,n} \end{bmatrix}. \tag{6.21}$$

Furthermore, the observer follows

$$\begin{aligned} \begin{bmatrix} \Delta\hat{\text{lat}}_{i,n+1} \\ \Delta\hat{\psi}_{i,n+1} \end{bmatrix} = \left(\begin{bmatrix} 1 & \frac{\bar{v}_{i,n}}{20} \\ 0 & 1 \end{bmatrix} + \begin{bmatrix} 0.3 & \frac{\hat{v}_{i,n}}{20} \\ 0 & 0.2 \end{bmatrix} \right) \begin{bmatrix} \Delta\hat{\text{lat}}_{i,n} \\ \Delta\hat{\psi}_{i,n} \end{bmatrix} + \frac{c_1 \bar{v}_{i,n}}{\cos^2(c_1\bar{\delta}_{i,n} + c_2)(c_3 + c_4\bar{v}_{i,n}^2)} \times \\ \times \begin{bmatrix} \left(\frac{(c_8 + c_9\bar{v}_{i,n}^2)}{20} + \frac{\bar{v}_{i,n}}{800} \right) \\ \frac{1 b_{i,n}}{20} \end{bmatrix} \Delta\delta_{i,n} - \begin{bmatrix} 0.3 & \frac{\hat{v}_{i,n}}{20} \\ 0 & 0.2 \end{bmatrix} \Delta\text{y-lat}_{i,n}, \end{aligned} \tag{6.22}$$

where $\Delta\text{y-lat}_{i,n}$ is the measurement of the lateral and heading error.

6.1.2 Longitudinal Controller Design

This section details the 3 different longitudinal controllers utilized in the platoon. In each case, we provide a general control scheme and the parameters chosen in this work based on their performance in the simulated platoon. The level 3 and level 2 controllers both attempt to maintain constant spacing between vehicles. However, they utilize different communication strategies that allow us to compare the benefits of full communication between vehicles in the platoon and a more limited communication strategy. The level 1 controller attempts to maintain constant headway without the aid of V2V communication. As a result, level 1 control strategy is used after an attack is detected, allowing the platoon to gracefully degrade. Note that the lateral controller has been included in Appendix 6.1.1 for completeness.

We assume that all vehicles in the platoon use the same communication level and switch to the mitigation strategy simultaneously when an attack is detected. For each level, the longitudinal controller and observer follows the LTV form ($\mathcal{D}3$) however the matrices $K_{i,n}$, $L_{(i,j),n}$, N_i, M_i, $U_{(i,j)}$, and $W_{(i,j)}$ are level dependent.

Level 3:

To take advantage of full V2V communication between the connected vehicles, a "fully-connected" controller and observer are devised. In this case

$$H = \{1, \dots, \kappa\} \times \{1, \dots, \kappa\} \tag{6.23}$$

and the measurements follow the model in (6.18),(6.19). The controller gains satisfy

$$K_{i,n} = \left[\begin{array}{c:c:c:c} \frac{0.5}{2^{(i-1)}} \cdots \frac{0.5}{2^{(1)}} & \frac{-0.5}{2^{(0)}} \cdots \frac{-0.5}{2^{(\kappa-i-1)}} & \frac{0.1}{2^{(i-1)}} \cdots \frac{0.1}{2^{(1)}} & \frac{-0.1}{2^{(0)}} \cdots \frac{-0.1}{2^{(\kappa-i)}} \end{array} \right] \tag{6.24}$$

for $i \in \{1, \dots, \kappa\}$. The idea here is to have each vehicle react to the velocity and following distance error of all other vehicles. In doing so, the watermark of each vehicle affects all others in the platoon. Furthermore, the magnitude of the control gains decays exponentially for vehicles further away in the platoon to reduce the combined effect of the watermarks especially in larger platoons.

As the platoon size increases, the number of communication channels and corresponding incoming messages for each vehicle increases. This problem along with other physical limiting factors such as latency between vehicles at the ends of the platoon can prove troublesome in larger, fully-connected platoons. Along with these limitations, the fact that the visibility of the watermark from vehicle i in vehicle j reduces as the platoon positions i and j are further apart means there are diminishing returns, from an attack detection standpoint, to having level 3 communication in

larger platoons.

$$N_i = I_{2\kappa-1} \tag{6.25}$$

$$L_{(i,j),n} = \begin{cases} \begin{bmatrix} -0.05 & 0_{1\times\kappa-2} & -0.1 & 0_{1\times\kappa-1} \end{bmatrix}^\intercal & j = 1 \\ \left[\begin{array}{c|c|c|c} 0_{2\times\kappa-2} & \begin{matrix} -0.5 \\ 0.05 \end{matrix} & 0_{2\times\kappa-1} & \begin{matrix} 0 \\ -0.1 \end{matrix} \end{array}\right]^\intercal & j = \kappa \\ \left[\begin{array}{c|c|c|c|c|c} 0_{2\times j-2} & \begin{matrix} -0.5 \\ 0.05 \end{matrix} & \begin{matrix} 0 \\ -0.05 \end{matrix} & 0_{2\times\kappa-2} & \begin{matrix} 0 \\ -0.1 \end{matrix} & 0_{2\times\kappa-j} \end{array}\right]^\intercal & \text{o/w.} \end{cases} \tag{6.26}$$

$$M_{i,n} = A_n + \sum_{j=1}^{\kappa} \left(B_{j,n} K_{j,n} + L_{(i,j),n} C_j \right) \tag{6.27}$$

$$W_{(i,j)} = C_j, \; U_{(i,j)} = I_{m_j}. \tag{6.28}$$

Level 2: In light of the limitations associated with level 3, a less connected strategy where

$$H = \{(i,j) \in \{1,\dots,\kappa\}^2 \mid j = 1 \text{ or } |i-j| \leq 1\} \tag{6.29}$$

and measurements that follow the measurement model defined in (6.18), (6.20) are considered for the level 2 communication. In this case, each vehicle observes a subset of the states. Namely, its distance to the lead vehicle, preceding vehicle, and following vehicle, and the velocities of each of these vehicles.

Next we derive the level 2 control strategy using inspiration from Swaroop and Hedrick [142, Section 3.4], in which the controller uses the state of the lead and preceding vehicles to calculate a desired acceleration as

$$\begin{aligned} \dot{v}_{i,n} = \frac{1}{1+\gamma_1} [& \dot{v}_{i-1,n} + \gamma_1 \dot{v}_{1,n} - (\gamma_2 + 0.6)(v_{i,n} - v_{i-1,n}) + \\ & - 0.6\gamma_2 (d_{i-1,n} - \bar{d}_{i-1,n}) - (\gamma_3 + 0.6\gamma_1)(v_{i,n} - v_{1,n}) + \\ & + 0.6\gamma_3 (\underline{d}_{i,n} - \bar{\underline{d}}_{i,n})], \end{aligned} \tag{6.30}$$

where γ_1 is used to shift the relative gain from the acceleration of the preceding vehicle to that of the leading vehicle, γ_2 adjusts the control gains corresponding to following distance and relative

velocity to the previous vehicle, and γ_3 adjusts the control gains corresponding to the following distance and relative velocity to the lead vehicle. To enact this control policy we start by setting $\gamma_1 = 0.2$, $\gamma_2 = 1$, and $\gamma_3 = 1.2$ to achieve a spacing error attenuation rate of less than 0.5 [142, Equation 3]. Then, since our controller will specify the desired velocity v^d instead of the desired acceleration, we relate the two using the partial derivative

$$\frac{\partial \dot{v}_{i,n}}{\partial v^d_{i,n}} = -1.2(c_6 + 2c_7(v_{i,n} - v^d_{i,n})). \tag{6.31}$$

Further, we assume that the deviation in the acceleration of the lead vehicle and preceding vehicle are negligible allowing us to ignore the corresponding terms. The resulting controller gain matrices satisfy

$$K_{i,n} = \frac{1}{-1.2(c_6 + 2c_7(\bar{v}_{i,n} - \bar{v}^d_{i,n}))} \begin{cases} \begin{bmatrix} 0 & -2.92 & 0 \end{bmatrix} & i = 1 \\ \begin{bmatrix} 1.32 & 0 & 2.92 & -2.92 & 0 \end{bmatrix} & i = 2 \\ \begin{bmatrix} 0.72 & 0.6 & 1.32 & 1.6 & -2.92 \end{bmatrix} & i = \kappa \\ \begin{bmatrix} 0.72 & 0.6 & 0 & 1.32 & 1.6 & -2.92 & 0 \end{bmatrix} & \text{o/w,} \end{cases} \tag{6.32}$$

with the corresponding observed state $\hat{x}_{i,n}$ approximating $N_i x_n$ defined such that

$$N_i x_n = \begin{cases} \begin{bmatrix} d_{1,n} & v_{1,n} & v_{2,n} \end{bmatrix}^\intercal & i = 1 \\ \begin{bmatrix} d_{1,n} & d_{2,n} & v_{1,n} & v_{2,n} & v_{3,n} \end{bmatrix}^\intercal & i = 2 \\ \begin{bmatrix} \underline{d}_{\kappa,n} & d_{\kappa-1,n} & v_{1,n} & v_{\kappa-1,n} & v_{\kappa,n} \end{bmatrix}^\intercal & i = \kappa \\ \begin{bmatrix} \underline{d}_{i,n} & d_{i-1,n} & d_{i,n} & v_{1,n} & v_{i-1,n} & v_{i,n} & v_{i+1,n} \end{bmatrix}^\intercal & \text{o/w} \end{cases} \tag{6.33}$$

$$M_{i,n} = N_i B_{i,n} K_{i,n} + \begin{cases} \left[\begin{array}{c:cc} 0.5 & \sigma_{1,n} & -\sigma_{2,n} \\ \hdashline 0_{2\times 1} & \multicolumn{2}{c}{\mathrm{diag}(\theta_{1,n}, \theta_{2,n})} \end{array}\right] & i = 1 \\ \left[\begin{array}{c:ccc} 0.5I_2 & \sigma_{1,n} & -\sigma_{2,n} & 0 \\ & 0 & \sigma_{2,n} & -\sigma_{3,n} \\ \hdashline 0_{3\times 2} & \multicolumn{3}{c}{\mathrm{diag}(\theta_{1,n}, \theta_{2,n}, \theta_{3,n})} \end{array}\right] & i = 2 \\ \left[\begin{array}{c:ccc} 0.5I_2 & \sigma_{1,n} & 0 & -\sigma_{\kappa,n} \\ & 0 & \sigma_{\kappa-1,n} & -\sigma_{\kappa,n} \\ \hdashline 0_{3\times 2} & \multicolumn{3}{c}{\mathrm{diag}(\theta_{1,n}, \theta_{\kappa-1,n}, \theta_{\kappa,n})} \end{array}\right] & i = \kappa \\ \left[\begin{array}{c:ccc} & \sigma_{1,n} & 0 & -\sigma_{i,n} & 0 \\ 0.5I_3 & 0 & \sigma_{i-1,n} & -\sigma_{i,n} & 0 \\ & 0 & 0 & \sigma_{i,n} & -\sigma_{i+1,n} \\ \hdashline 0_{4\times 3} & \multicolumn{4}{c}{\mathrm{diag}(\theta_{1,n}, \theta_{i-1,n}, \theta_{i,n}, \theta_{i+1,n})} \end{array}\right] & \text{o/w} \end{cases} \tag{6.34}$$

$$L_{(1,j),n} = \begin{cases} \left[\begin{array}{c:c} & -0.05 \\ 0_{3\times\kappa-1} & -0.1 \\ & 0 \end{array}\right] & j = 1 \\ \begin{bmatrix} -0.5 & 0 & 0 \\ 0.05 & 0 & -0.1 \end{bmatrix}^{\mathsf{T}} & j = 2 \end{cases} \tag{6.35}$$

for the first vehicle,

$$L_{(2,j),n} = \begin{cases} \left[\begin{array}{c:c:c} & & -0.05 \\ -0.25 & 0_{1\times\kappa-2} & 0 \\ 0_{4\times1} & & -0.1 \\ & & 0_{2\times1} \end{array}\right] & j = 1 \\ \begin{bmatrix} -0.25 & 0 & 0 & 0 & 0 \\ 0.05 & -0.05 & 0 & -0.1 & 0 \end{bmatrix}^\intercal & j = 2 \\ \begin{bmatrix} 0 & -0.5 & 0 & 0 & 0 \\ 0 & 0.05 & 0 & 0 & -0.1 \end{bmatrix}^\intercal & j = 3 \end{cases} \tag{6.36}$$

for the second vehicle,

$$L_{(\kappa,j),n} = \begin{cases} \left[\begin{array}{c:cc} & -0.5 & -0.05 \\ & 0 & 0 \\ 0_{5\times\kappa-2} & 0 & 0.1 \\ & 0_{2\times1} & 0_{2\times1} \end{array}\right] & j = 1 \\ \begin{bmatrix} 0 & -0.05 & 0 & -0.1 & 0 \end{bmatrix}^\intercal & j = \kappa - 1 \\ \begin{bmatrix} 0 & -0.5 & 0 & 0 & 0 \\ 0.05 & 0.05 & 0 & 0 & -0.1 \end{bmatrix}^\intercal & j = \kappa \end{cases} \tag{6.37}$$

for the last vehicle, and

$$L_{(i,j),n} = \begin{cases} \left[\begin{array}{c:c:c:c} & & & -0.05 \\ & & & 0 \\ 0_{7\times i-2} & \begin{array}{c}-0.5\\ 0_{6\times 1}\end{array} & 0_{7\times \kappa - i} & 0 \\ & & & -0.1 \\ & & & 0_{3\times 1} \end{array}\right] & j = 1 \\ \begin{bmatrix} 0 & -0.05 & 0 & 0 & -0.1 & 0 & 0 \end{bmatrix}^\intercal & j = i-1 \\ \begin{bmatrix} 0 & -0.5 & 0 & 0 & 0 & 0 & 0 \\ 0.05 & 0.05 & -0.05 & 0 & 0 & -0.1 & 0 \end{bmatrix}^\intercal & j = i \\ \begin{bmatrix} 0 & 0 & -0.5 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.05 & 0 & 0 & 0 & -0.1 \end{bmatrix}^\intercal & j = i+1 \end{cases} \tag{6.38}$$

for the remaining vehicles i.e. $i \notin \{1, 2, \kappa\}$.

$$U_{(i,j)}N_i x_n = W_{(i,j)}C_j x_n = \begin{cases} \begin{bmatrix} v_{1,n} \end{bmatrix} & j = 1,\ i = 1 \\ \begin{bmatrix} \sum_{k=1}^{i-1} d_{k,n} & v_{1,n} \end{bmatrix}^\intercal & j = 1,\ i \neq 1 \\ \begin{bmatrix} v_{j,n} \end{bmatrix} & j \neq 1,\ i = j+1 \\ \begin{bmatrix} d_{j-1,n} & v_j \end{bmatrix}^\intercal & j \neq 1,\ i \neq j+1 \end{cases}. \tag{6.39}$$

Level 1: In the event of an attack detection, communication between agents in the network should be severed to mitigate potential harm to the system. To maintain operation of the platoon, the vehicles are able to switch to a non-communicative platoon strategy such that

$$H = \{(i, i) \mid i \in \{1, \ldots, \kappa\}\}. \tag{6.40}$$

Here each vehicle still measures and observes its own velocity and the distance to the preceding vehicle.

The non-communicative level 1 controller is inspired by the University of Michigan Transportation Research Institute's (UMTRI)'s algorithm for adaptive cruise control (ACC) [147, Equation

1] which satisfies

$$v_{i,n}^{d} = v_{i-1,n} + \phi_1(f_{i-1,n} - T_h v_{i,n}) + \\ + \phi_2(v_{i-1,n} - v_{i,n}), \tag{6.41}$$

where ϕ_1 is the control gain for the error in the headway, $f_{i-1,n}$ is the bumper-to-bumper distance to the previous vehicle, and ϕ_2 is the control gain on the derivative of the following distance. In this work, we set $\phi_1 = 1$, $\phi_2 = 0.2$, and $T_h = 1$. Since (6.41) is already linear the control gain $K_{i,n}$ is written directly as

$$K_{i,n} = \begin{cases} \begin{bmatrix} -1 \end{bmatrix} & i = 1 \\ \begin{bmatrix} 1 & 1.2 & -1.2 \end{bmatrix} & i \neq 1, \end{cases} \tag{6.42}$$

with the corresponding observed state $\hat{x}_{i,n}$ approximating $N_i x_n$ such that

$$N_i x_{i,n} = \begin{cases} \begin{bmatrix} v_{1,n} \end{bmatrix} & i = 1 \\ \begin{bmatrix} d_{i-1,n} \\ v_{i-1,n} \\ v_{i,n} \end{bmatrix} & i \neq 1 \end{cases} \tag{6.43}$$

Note that a proportional gain is applied to the error in the lead vehicle's velocity since there is no preceding vehicle.

Since the level 1 control strategy is considered to be safe from cyber attacks across communication channels, we do not employ attack detection or a watermark after the switch is made. Though the level 1 controller is less susceptible to cyber attacks, it is unable to maintain constant following distances without sacrificing safety. In contrast, both the level 3 and level 2 control strategies enable the use of constant following distances. As a result, these strategies can lead to significant improvement in fuel economy and throughput on roads.

$$M_{i,n} = N_i B_{i,n} K_{i,n} + \begin{cases} \theta_{1,n} & i = 1 \\ \begin{bmatrix} 0.5 & \frac{\beta_{i-1,n}-1}{\alpha_{i-1,n}} & -\sigma_{i,n} \\ -1.2 & \beta_{i-1,n} & 0 \\ 0 & 0 & \theta_{i,n} \end{bmatrix} & i \neq 1 \end{cases} \tag{6.44}$$

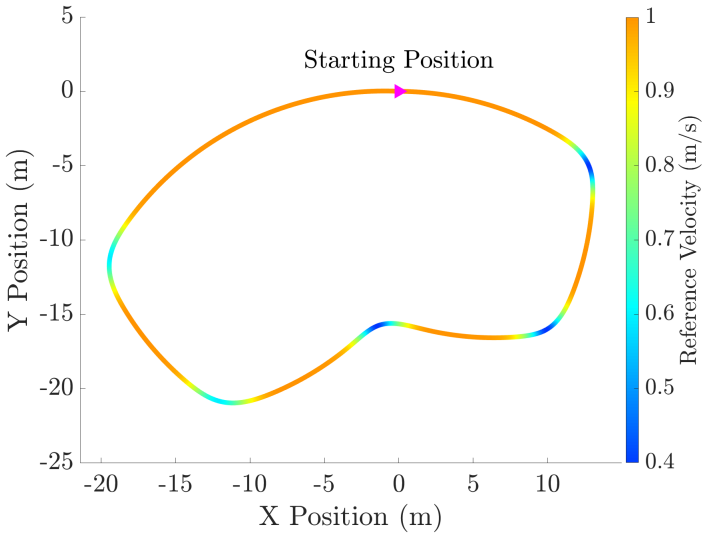


Figure 6.2: Reference trajectory of the lead vehicle in simulated platoon experiments. Each simulation consists of three laps.

$$L_{(i,i),n} = \begin{cases} -0.1 & i = 1 \\ \begin{bmatrix} -0.5 & 0.05 \\ -1.2 & 0 \\ 0 & -0.1 \end{bmatrix} & i \neq 1 \end{cases} \tag{6.45}$$

6.2 Results

This section illustrates the effectiveness of networked LTV dynamic watermarking on detecting attacks on V2V communication of platooning vehicles. The experiment is implemented on a simulated platoon of four autonomous vehicles traveling three times around the looped path illustrated in Figure 6.2. In the simulation, the vehicles have a vehicle length of 0.5m and drive according to the non-linear dynamics defined in (5.14).

For simulations of the level 3 and 2 controllers, the platoon was tasked with maintaining a 1m constant following distance which was chosen based on the ability of each controller to maintain the desired distance as shown in Table 6.1. Simulations of the level 1 controller had the platoon maintain a constant headway of 1 second.

Process and measurement noise as defined in (3.143) and (3.144) were also added at each step n, where $\Sigma_{w,n} = 1 \times 10^{-6} I_p$, and $\Sigma_{z_i,n} = \text{diag}(1 \times 10^{-4}, 1 \times 10^{-3})$ for $i \in \{2, \ldots, \kappa\}$. For the lead vehicle, the measurement noise covariance is $\Sigma_{z_1,n} = 1 \times 10^{-3}$. However, for the special case

where the distance from each vehicle to the lead vehicle $\underline{d}_{i,n}$ is treated as a measurement the value $\Sigma_{z_1,n} = \text{diag}(1\times10^{-4}I_{\kappa-1}, 1\times10^{-3})$ is used instead. While the noise added to the system is Gaussian, the state update between time steps is done using the nonlinear dynamics in equation (5.14). This results in a non-Gaussian distribution of the platoon state, which is meant to better approximate the noise of a real-world system.

6.2.1 Dynamic Watermarking Setup

At each time step a watermark $e_{i,n}$ satisfying

$$\Sigma_{e_i} = 0.25, \forall i \in \{1, \ldots, \kappa\} \tag{6.46}$$

was added to each vehicles input. While the watermark enables the detection of a wider range of attacks, it also increases the noise in the system resulting in reduced performance of the controller. This leads to a trade off between performance and making sure the watermark is sufficiently visible in the face of other noise sources. As discussed at the beginning of the chapter, the benefit of V2V communications in vehicle platooning stem from the reduction in following distance under a constant spacing policy. Therefore, to observe the reduction in performance due to the watermark, the mean and standard deviation of the bumper-to-bumper distances between vehicles were computed over 20 simulations with and without the added watermark. Table 6.1 shows the results for each level controller. Since the level 1 controller is used only after an attack is detected, we do not add a watermark to this controller. From this comparison, we note that there is indeed a reduction in performance resulting from the watermark as illustrated by the increased standard deviation for both the level 3 and level 2 controller. However, even with this reduced performance, the level 3 and level 2 controller still maintain a smaller following distance than the level 1 controller.

	Without watermark		With watermark	
Level	Mean	Std	Mean	Std
3	0.50	0.01	0.50	0.11
2	0.50	0.02	0.49	0.08
1	1.35	0.21	-	-

Table 6.1: Aggregate statistics for bumper-to-bumper distance (m) using 20 un-attacked simulations for each controller/watermark combination. Each simulation consists of a platoon of four vehicles following the trajectory in Figure 6.2.

For each controller, the matrix normalization factor and the auto-correlation normalizing factor were generated from 50 simulations using (3.178)-(3.181) and (3.182)-(3.183) respectively. To illustrate the benefit of using the matrix normalizing factor and auto-correlation normalizing factor of the proposed method, we provide a comparison to the LTI equivalent described in Section 3.1.

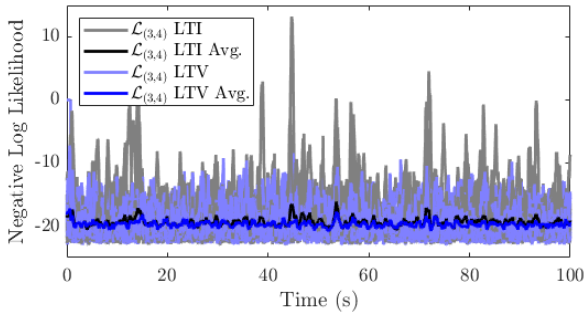


Figure 6.3: Comparing LTV to LTI Dynamic Watermarking

We compute the matrix normalization factor and auto-correlation normalizing factor for the LTI case as follows

$$V_{(i,j)} = \frac{1}{6001} \sum_{n=0}^{6000} V_{(i,j),n}, \tag{6.47}$$

$$G_{(i,j)} = I_{p_{(i,j)}+q_i}, \tag{6.48}$$

where 6001 is the number of steps in the simulation.

For measurement $s_{(3,4)}$ we calculate the negative log likelihood using both the LTI and LTV normalizing factors for 20 un-attacked simulations as illustrated in Figure 6.3. While the average negative log likelihood signal, taken over 20 simulations, is similar for both the LTI and LTV case, the actual signal for the LTI case shows far more spiking. As a result, the threshold in the LTI case would likely need to be set higher to avoid false alarms. However, a higher threshold reduces the ability of the detector to identify attacks. Hence, the LTV Dynamic Watermarking is superior for this system.

To select a robust threshold, the attack thresholds are computed to achieve a desired false alarm rate based on a set of un-attacked trials. The false alarm rate is defined as the number of time steps above the attack threshold divided by the total number of time steps. In this paper, 20 trials were used to calculate the thresholds which achieved a false alarm rate of 0.5%.

To decide when to switch to the level 1 controller, we count the number of times each negative log likelihood has exceeded its threshold in the last 40 steps. If this value exceeds 24 (60%) for any given communication channel, the platoon switches to the level 1 controller. The values of 60% and 40 steps were chosen based on their ability to reduce the number of unnecessary switches from false alarms while still avoiding collisions in our simulated platoon.

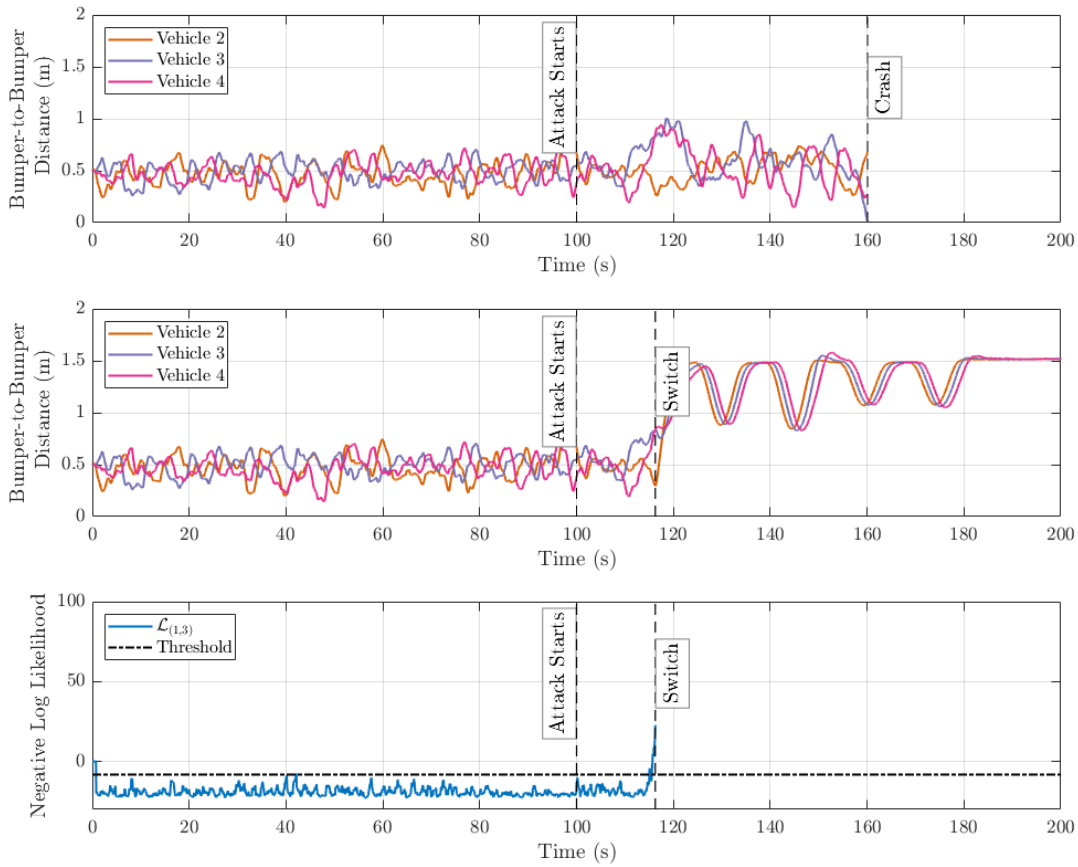


Figure 6.4: (Top) Performance of the level 3 controller after a replay attack without switching to the level 1 controller and crashing soon thereafter. (Middle) Platoon switching to level 1 controller after detecting the attack and safely completing the entire trajectory. (Bottom) Negative log likelihood of channel which detected the replay attack first.

6.2.2 Attack Schemes

In this paper, we considered two different types of attacks, the stealthy replay attack and an aggressive attack. While these attacks are not necessarily optimal, they are meant to represent two possible approaches of an attacker: 1) to go unnoticed while the effect of the attack slowly builds and 2) to make no attempt at remaining stealthy while trying to affect the system before a mitigation strategy can be implemented.

For a replay attack, the measurement signals (i.e. $s_{(i,j),n}$) from all or a subset of H communication channels from an un-attacked realization are recorded and then played back when the simulation is run for a separate realization. Since an attack need not start at the beginning, we chose to start the attack 100s after the start of the simulation. The replayed measurements included the distance between vehicles and the velocity of each vehicle in a platoon.

For the aggressive attack, we aimed to cause a collision as quickly as possible. To generate the attack, the communication channel from vehicle 1 to vehicle 2 (i.e. $s_{(2,1),n}$) is hacked and the velocity measurement is set to zero. The attack leads vehicle 2 to believe the lead vehicle is braking

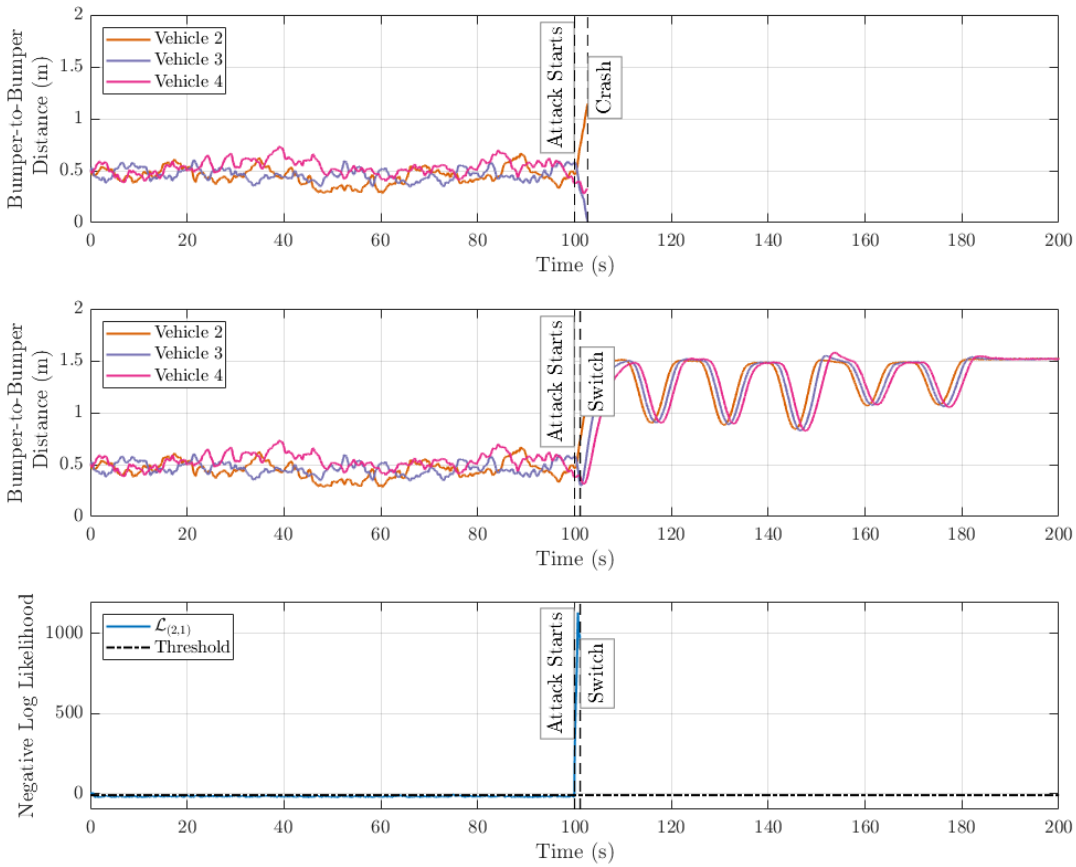


Figure 6.5: (Top) Performance of the level 2 controller after an aggressive attack without switching to the level 1 controller and crashing soon thereafter. (Middle) Platoon switching to level 1 controller after detecting the attack and safely completing the entire trajectory. (Bottom) Negative log likelihood of channel which detected the replay attack first.

and so brakes as well. This results in vehicle 3, which receives the unaltered measurement from the lead vehicle, to crash into vehicle 2. While this attack scheme is overt, it only requires intercepting a single communication channel and, if not mitigated quickly, results in a crash. As with the replay attack, we start this attack 100s after the start of the simulation.

6.2.3 Simulation Results

To demonstrate the proposed detection algorithm, simulations were run for the level 2 and 3 control methods following Algorithm 1 with an attack scheme as described in Section 6.2.2. After an attack was detected as described in Section 6.2.1, the simulation was split into two concurrent simulations of the platoon, one in which the platoon degrades to the level 1 controller and one in which the platoon does not. For the simulation, a crash was defined as the bumper-to-bumper distance between any two vehicles reaching 0 m.

For the simulations presented here, the replay attack involved attacking all communication

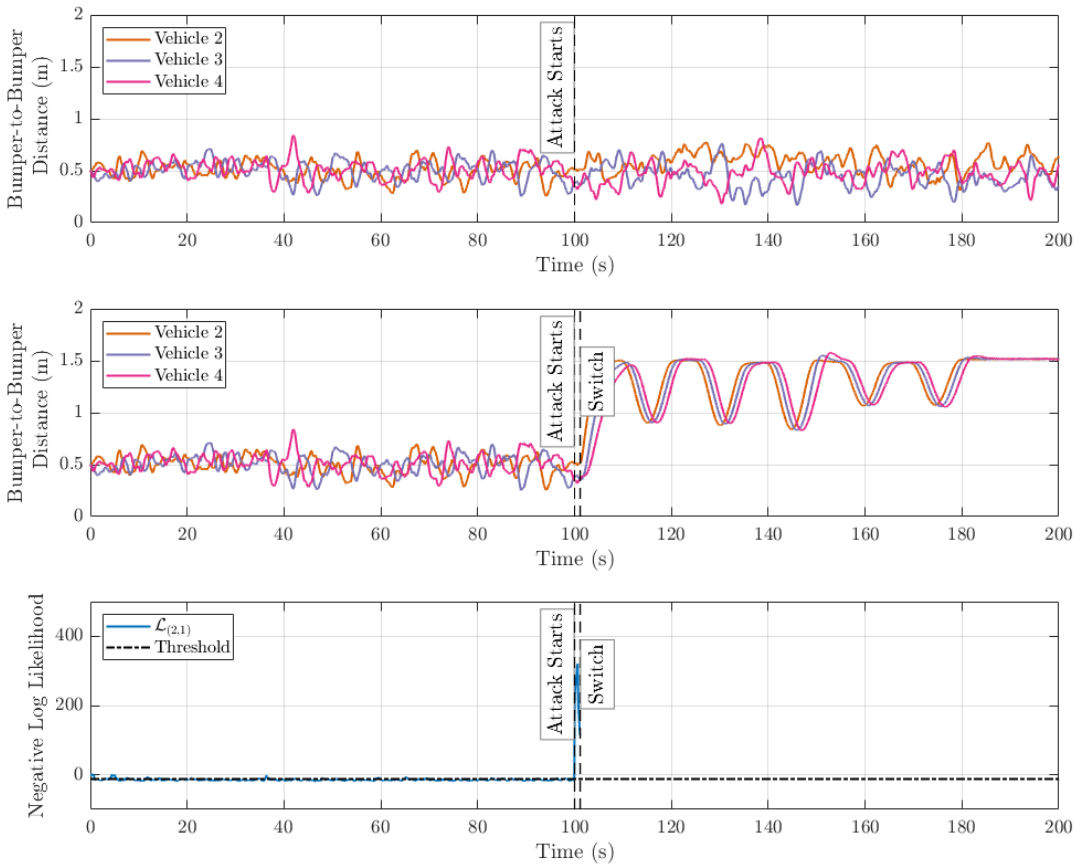


Figure 6.6: (Top) Performance of the level 3 controller after an aggressive attack without switching to the level 1 controller. However, it completes the trajectory without crashing. (Middle) Platoon switching to level 1 controller after detecting the attack and completing the entire trajectory. (Bottom) Negative log likelihood of channel which detected the replay attack first.

channels between vehicles. However, even for simulations where a subset of channels were replay attacked, the attack was detected in the corresponding channels and the controller was successfully able to degrade. This is important because even attacking a subset of communication channels can result in a crash and so being able to detect this in a timely manner is crucial. In all cases, the attack was detected before any crash allowing the platoon to gracefully degrade to the level 1 controller.

For the level 3 controller, we see the effects of the replay attack in Figure 6.4. Even though the replay attack is subtle in operation, it is still able to cause a crash if the platoon does not degrade control schemes. However, the level 3 controller appears resilient to the aggressive attack as illustrated in Figure 6.6. This is likely because the level 3 controller has a higher weighting on maintaining constant distance than velocity according to the controller gain in (6.24).

In contrast, the level 2 controller appears to be more susceptible to the aggressive attack. In Figure 6.5, the performance of the level 2 controller worsens drastically under the aggressive attack as vehicle 2 brakes and almost immediately collides with vehicle 3 as a result. However, the level 2 controller is more resilient to the replay attack, as seen in Figure 6.7. This is likely because the

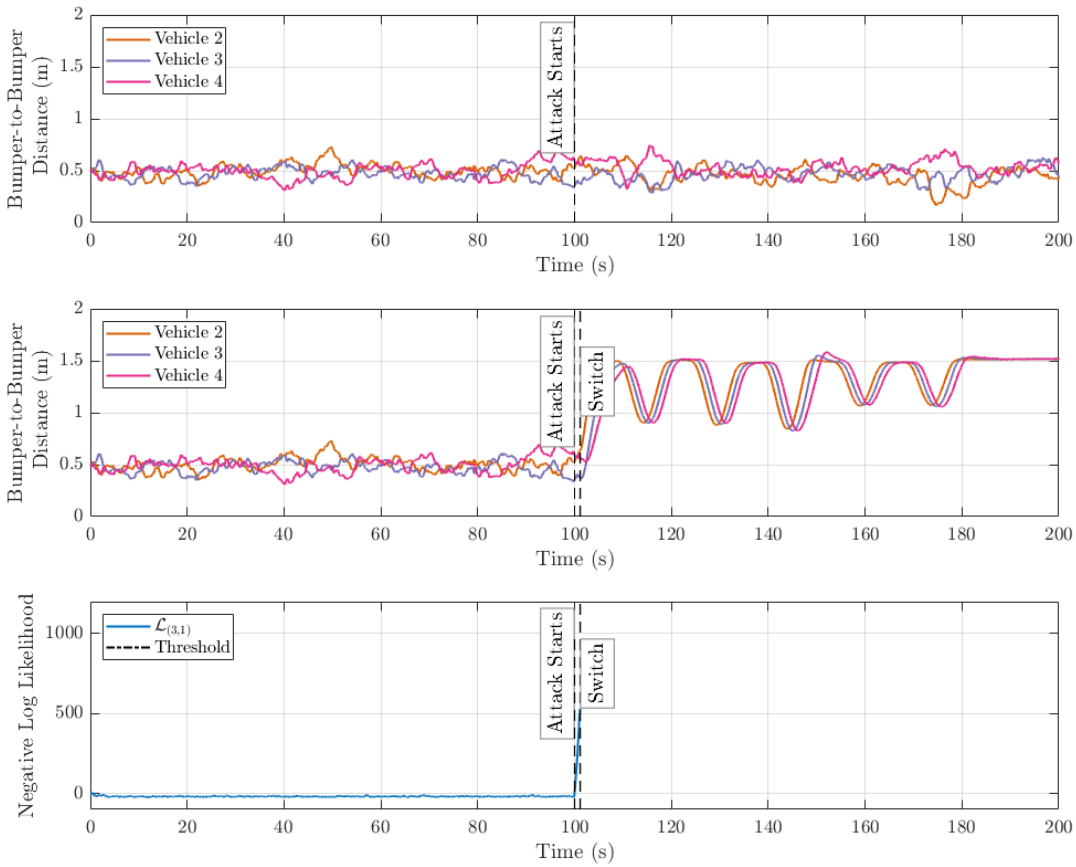


Figure 6.7: (Top) Performance of the level 2 controller after a replay attack without switching to the level 1 controller. However, it completes the trajectory without crashing. (Middle) Platoon switching to level 1 controller after detecting the attack and completing the entire trajectory. (Bottom) Negative log likelihood of channel which detected the replay attack first.

level 2 controller relies mainly on states that are measured directly.

6.2.4 Scaling the Platoon Size

To highlight the effect of scaling up the platoon size, we ran the same simulated experiments on a platoon of ten autonomous vehicles. In each realization, the platoon was subject to one of three possible attacks for both level 3 and 2 controllers: a replay attack on 30% of communication channels, a replay attack on 100% of channels, and an aggressive attack (as described in Section 6.2.2). For each realization, the time of attack was randomly sampled from a range of $[100, 200]$s and the subset of communication channels attacked are randomly selected to avoid any bias from the trajectory or specific communication channels.

For each controller, the matrix normalization factor and the auto-correlation normalizing factor were generated from 100 un-attacked realizations. Attack thresholds are computed on 500 un-attacked realizations for a rate of false alarm of 0.1%. The platoon switches to the level 1 control

strategy when the negative log likelihood exceeds the attack threshold for 18 steps in the last 40 steps (45%) for any communication channel.

The results of comparing LTI vs LTV Dynamic Watermarking for the level 3 and level 2 controllers are shown in Tables 6.2 and 6.3 respectively. The results are shown for replay attack 1/ replay attack 2/ aggressive attack.

From Tables 6.2 and 6.3, potential crashes represent the number of realizations which would crash when not running the dynamic watermarking attack detection. Looking at the potential crashes, we see that the level 3 controller is more robust to aggressive attacks whereas the level 2 controller is more robust to replay attacks.

Actual crashes represent the number of realizations which crashed while running the LTI or LTV dynamic watermarking algorithm. These crashes result from the effect of the attack on the platoon remaining below the user-defined threshold for detection. Selecting different user-defined parameters can reduce the number of actual crashes with the trade off of potentially higher number of false alarms.

It is worth noting that for the level 3 controller, one replay attack 1 realization using the LTI attack detection scheme and one realization using the LTV scheme were not determined successful detections, false alarms, or crashes. Upon further investigation, we concluded that the platoon performance was not affected by the attack in those realizations and so the dynamic watermark algorithm was not able to successfully detect the attack. A similar conclusion was made, looking at replay attack 1 for the level 2 controller, for 17 and 35 realizations using the LTI and LTV attack detection schemes respectively, where the replay attack did not deteriorate the performance of the platoon significantly and so there were no crashes and no successful detection. These statistics are recorded in Tables 6.2 and 6.3 as no attack detected and no crash caused. Selecting different user-defined parameters may improve the detection rate for these realizations.

When comparing the LTV dynamic watermarking algorithm to the LTI version, we see that the LTV attack detection scheme has a greater number of successful attack detections and lesser number of false alarms while maintaining a similar number of crashes. This difference in performance is highlighted in Table 6.2, where successful attack detection using LTI dynamic watermarking is approximately 75% whereas using LTV dynamic watermarking, we achieve approximately 95%. Increasing the user-defined thresholds for the LTI attack detection scheme led to a marginal decrease in number of false alarms at the cost of a substantial increase in number of crashes. Hence, the LTV dynamic watermarking is superior for this system.

Algorithm 1: Longitudinal Control for Vehicle i

set cntrl_lvl=3 (resp. 2);
set $\hat{x}_{i,0} = 0_{p_i}$;
set $n = 0$;
set H_i using (6.23) (resp. (6.29)) and (3.140);
Loop
  for $j \in H_i$ **do**
    get $s_{(i,j),n}$;
    set $\text{detect}_{(i,j),n} = 0$;
    if $n > \rho_{(i,j)}$ and cntrl_lvl$\neq$1 **then**
      set $V_{(i,j),n} = \bar{V}_{(i,j),h_i(n)}$;
      set $\bar{r}_{(i,j),n}$ using (3.165);
    end
    if $n > \rho_{(i,j)} + \ell_{(i,j)}$ **then**
      set $G_{(i,j),n} = \bar{G}_{(i,j),h_i(n)}$;
      set $\mathcal{L}_{(i,j),n}$ using (3.166), (3.169), and (3.171);
      if $\mathcal{L}_{(i,j),n} > \text{Threshold}_{(i,j)}$ **then**
        set $\text{detect}_{(i,j),n}$=1;
      end
      if $(\sum_{k=n-39}^{n} \text{detect}_{(i,j),n}) > 24$ **then**
        set cntrl_lvl=1;
        reformat $\hat{x}_{i,n}$
      end
    end
  end
  find $h_i(n)$ from high res. trajectory;
  set $\{\bar{v}_{j,n}, \bar{v}^d_{j,n}\} = \{\underline{v}_{j,h_i(n)}, \underline{v}^d_{j,h_i(n)}\}$;
  if cntrl_lvl=3 (resp. 2) **then**
    set $M_{i,n}, K_{i,n}$ using (6.27),(6.24) (resp. (6.34),(6.32));
    for $j \in H_i$ **do**
      set $L_{(i,j),n}$ using (6.26) (resp. (6.35)-(6.38));
    end
    sample $e_{i,n}$;
    set $u_{i,n} = K_{i,n}\hat{x}_{i,n} + e_{i,n} + \bar{u}_{i,n}$;
  else
    set $M_{i,n},\ L_{(i,i),n},\ K_{i,n}$ using (6.44),(6.45),(6.42);
    set $u_{i,n} = K_{i,n}\hat{x}_{i,n} + \bar{u}_{i,n}$;
  end
  set $\hat{x}_{i,n+1}$ using ($\mathcal{D}$3);
  send $u_{i,n}$;
  set $n = n + 1$;
EndLoop

	LTI	LTV
Successful detections	1513/1494/1530	1903/1876/1913
False alarms	482/470/470	88/78/87
Potential crashes	1099/1997/24	
Actual crashes	4/36/0	8/46/0
No attack detected and no crash caused	1/0/0	1/0/0
Mean detection time (s)	2.771/2.823/0.905	2.701/2.587/0.957
Standard deviation detection time (s)	3.118/3.125/0.041	3.812/3.576/0.363

Table 6.2: Attack detection statistics for 2000 trials of level 3 controller.

	LTI	LTV
Successful detections	1880/1902/1968	1944/1987/1982
False alarms	103/98/32	21/13/18
Potential crashes	3/4/2000	
Actual crashes	0/0/0	0/0/0
No attack detected and no crash caused	17/0/0	35/0/0
Mean detection time (s)	2.193/0.907/1.099	1.925/0.901/0.899
Standard deviation detection time (s)	10.221/0.075/0.031	9.003/0.077/0.022

Table 6.3: Attack detection statistics for 2000 trials of level 2 controller.

Chapter 7

Conclusions and Future Directions

In this final chapter we start by revisiting the contributions of this dissertation in Section 7.1. Then we discuss potential areas of future research in Section 7.2. Finally, we provide our final concluding remarks in Section 7.3.

7.1 Discussion of Contributions

Dynamic watermarking, as defined in this dissertation, is shown to detect cyber-attacks that alter measurements in a variety of ITS applications. Moreover, dynamic watermarking is proven to detect attack models that previous detection algorithms cannot. In doing so, dynamic watermarking can enable safe and equitable ITSs by alerting the system to the presence of an attack and allowing mitigation strategies to be implemented in a timely manner. The particular contributions of this dissertation that facilitate the application of dynamic watermarking are organized by chapter as follows.

Chapter 3: Dynamic Watermarking This chapter lays the ground work for the remainder of the dissertation by deriving multiple forms of dynamic watermarking each with their own particular use cases. Namely, dynamic watermarking is derived for both LTI and LTV systems [22], [95], [96] and a method for extending dynamic watermarking to distributed control systems is described [85], [100]. In each case, we outline a generic model with necessary assumptions. Then, we provide limit-based tests that are guaranteed to detect a wide range of attack models. The limit based tests are then used to develop implementable statistical tests. Furthermore, we demonstrate the importance of using a sufficiently accurate model and corresponding form of dynamic watermarking by illustrating the potential adverse effects.

Chapter 4: Tools for Selecting User-Defined Parameter In this chapter, we discuss the process of selecting user-defined parameters and provide tools to enable informed decision making. When selecting parameters such as the threshold τ, window size ℓ, and watermark covariance Σ_e, one may wish to minimize the rate at which false alarms occur, constrain the effect an attacker can have while going unnoticed, and ensure that particular forms of attack will be detected. We address each of these goals by discussing methods for estimating the rate of false alarms and attackers capability. Moreover we evaluate the real-world ability of LTI dynamic watermarking and several other detection algorithms to detect several attack models.

Chapter 5: Single Autonomous Vehicle Applications In Chapter 5, we focus on detecting cyber-attacks on a single autonomous vehicle [95]. We provide a proof of concept for LTV dynamic watermarking using both a high-fidelity vehicle model in CarSimand a 1/10 scale autonomous rover. In each case, a replay attack is implemented in the middle of a trajectory following task, and LTV Dynamic Watermarking is shown to quickly detect the attack in a repeatable fashion. This contribution is particularly important since all previous dynamic watermarking approaches were limited to LTI systems which, as noted in Chapter 3, are insufficient for enabling reliable detection of attacks.

Chapter 6: Autonomous Platoon Applications In Chapter 5, we focus on detecting cyber-attacks on a platoon of autonomous vehicles that utilize V2V communications [100]. In particular, we introduced 3 levels of V2V communication and defined corresponding longitudinal controllers two of which leveraged the extra information for feedback control while the third does not rely on any V2V communication and was used as a mitigation strategy in the event of an attack. Compared to LTI distributed dynamic watermarking, we showed that LTV distributed dynamic watermarking is superior in that it provides a more consistent test metric. We described two different attack schemes, one stealthy and one aggressive, and showed that our algorithm could detect both types of attack while utilizing each controller and successfully degrading to a safe control strategy before a crash can occur.

7.2 Future Research Directions

Despite the advances described in this dissertation, several gaps remain. We discuss some of the gaps here.

7.2.1 Linearization Gap

Although in Section 3.1 and Section 3.2 we provide guarantees of detection for LTI and LTV systems respectively, the true dynamics of many systems of interest are in fact non-linear. As a result, these guarantees of detection may not hold for the real-world system. Some extensions to simple non-linear systems have been proposed [81], [87], [88]. However, it is unclear if the existing approaches can be extended to more complex models such as CAVs under real-world conditions. Ideally, we could apply dynamic watermarking to any non-linear model or an arbitrarily close approximation. Then provide a similar guarantee of detection or, in the case of an approximate model, a bound on the attackers ability given the approximation gap.

7.2.2 Attack Identification

CAVs and other CPS rely on a variety of measurement sources. As such, an attacker may only be able to alter a subset of the measurements corresponding to a particular source. The ability of a detection algorithm to identify which source or sources have been compromised can allow for the attack to be mitigated. For example, CAVs have many measurements are redundant or can be approximated using other measurements. This could allow a CAV to operate even when under attack by disregarding measurements that are deemed untrustworthy. Likely, the solution to identifying attacked measurement sources will be to isolate each source's effect and test them individually similarly to what is done in the distributed control case.

7.2.3 Guarantees for Distributed Control

The extension of dynamic watermarking to distributed control systems described in Section 3.3 provides a means of applying the statistical tests of both LTI and LTV dynamic watermarking in applications with distributed control. However, the guarantees of detection can not be trivially extended. To the best of our knowledge, the only guarantees provided for dynamic watermarking applied to distributed control systems are provided by Ko *et al.* [87] in which they apply non-linear dynamic watermarking to a platoon of vehicles. Nevertheless, they make several simplifying assumptions that make it unclear if their method can be extended beyond simplistic vehicle models. Ideally, similar guarantee's of detection could be applied to any distributed control system regardless of the underlying model.

7.2.4 Confidence in Detection

Instead of simply providing a binary classification of whether or not an attack is occurring, a measure of confidence in the classification could enable more informed decisions. Namely, one

might consider a mitigation strategy that enacts increasingly impactful actions corresponding to the confidence of a given detection. Such a strategy would likely reduce the adverse effect of false alarms while also reacting quickly to aggressive attacks. Due to the statistical nature of the tests described in this dissertation, providing a confidence measure may be relatively straightforward. However, to the best of our knowledge, no existing literature examines the effectiveness of using a confidence measure with dynamic watermarking.

7.3 Concluding Remarks

In the last several years, cyber attacks have become increasingly prevalent in all facets of technology. ITS are no exception. Moreover, future ITS aim to increase the reliance on networked communications in transportation thus exacerbating the potential effects of a cyber attack. The attack detection algorithm, dynamic watermarking, outlined in this dissertation can be used to detect cyber-attacks that alter the content of measurement channels. Namely, this dissertation develops dynamic watermarking for LTI and LTV system models both in classical and distributed control applications. As a result, dynamic watermarking can be applied to a wide range of ITS and, when paired with an appropriate mitigation strategy, ensures safe and equitable operation.

While this work focuses on detecting attacks on ITS, the applications of dynamic watermarking are vast. For example, other large CPS, such as power and water infrastructure, share many commonalities with ITS and would also benefit from this work. Furthermore, dynamic watermarking has the potential to collaborate with the related field of anomaly detection. In particular, dynamic watermarking uses sophisticated statistical tests that can improve existing anomaly detection methods while also enabling the detection of cyber attacks. In return, literature in anomaly detection provides a general framework for identifying the source of a given anomaly. This framework can potentially enable dynamic watermarking to identify the source of an attack or to discern the difference between a faulty sensor and an attack. With the addition of such functionality, and those outlined in the future research directions, dynamic watermarking is capable of ensuring safe and equitable operation of both ITS and other future CPS.

Bibliography

[1] T. Reed, "Inrix global traffic scorecard," 2019.

[2] D. Ellis and B. Glover, "2019 urban mobility report," 2019.

[3] M. Barth and K. Boriboonsomsin, "Real-World Carbon Dioxide Impacts of Traffic Congestion," *Transportation Research Record*, vol. 2058, no. 1, pp. 163–171, 2008.

[4] NHTSA, *Traffic safety facts annual report*, updated: June 30, 2020, 2020.

[5] X. Huang, D. Zhao, and H. Peng, "Empirical Study of DSRC Performance Based on Safety Pilot Model Deployment Data," *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 10, pp. 2619–2628, 2017.

[6] K. N. Balke, M. Lukuc, B. T. Kuhn, M. W. Burris, J. Zmud, A. Morgan, R. G. Dowling, G. Morrison, R. Marsters, and T. Szymkowski, "Connected vehicle pilot deployment program independent evaluation: Comprehensive evaluation plan—new york city," United States. Department of Transportation. Intelligent Transportation Systems Joint Program Office, Tech. Rep., 2019.

[7] ——, "Connected vehicle pilot deployment program independent evaluation: Comprehensive evaluation plan—tampa," United States. Department of Transportation. Intelligent Transportation Systems Joint Program Office, Tech. Rep., 2019.

[8] ——, "Connected vehicle pilot deployment program independent evaluation: Comprehensive evaluation plan—wyoming," United States. Department of Transportation. Intelligent Transportation Systems Joint Program Office, Tech. Rep., 2019.

[9] R. Langner, "Stuxnet: Dissecting a Cyberwarfare Weapon," *IEEE Security & Privacy*, vol. 9, no. 3, pp. 49–51, 2011.

[10] M. Abrams and J. Weiss, "Malicious control system cyber security attack case study - Maroochy water services, australia," *MITRE*, 2008.

[11] R. M. Lee, M. J. Assante, and T. Conway, "German steel mill cyber attack," *Industrial Control Systems*, vol. 30, p. 62, 2014.

[12] R. M. Lee, M. J. Assante, and T. Conway, "Analysis of the cyber attack on the ukrainian power grid," *Electricity Information Sharing and Analysis Center (E-ISAC)*, 2016.

[13] H. Sandberg, S. Amin, and K. H. Johansson, "Cyberphysical security in networked control systems: An introduction to the issue," *IEEE Control Systems Magazine*, vol. 35, no. 1, pp. 20–23, 2015.

[14] A. A. Cárdenas, S. Amin, and S. Sastry, "Research challenges for the security of control systems," in *Proceedings of the 3rd Conference on Hot Topics in Security*, ser. HOTSEC'08, USENIX Association, 2008, pp. 1–6.

[15] M. Gerla and P. Reiher, "Securing the Future Autonomous Vehicle: A Cyber-Physical Systems Approach," in *Securing Cyber-Physical Systems*, CRC Press, 2015, ch. 7, pp. 197–220.

[16] D. Dominic, S. Chhawri, R. M. Eustice, D. Ma, and A. Weimerskirch, "Risk Assessment for Cooperative Automated Driving," in *Proceedings of the 2nd ACM Workshop on Cyber-Physical Systems Security and Privacy - CPS-SPC '16*, 2016, pp. 47–58.

[17] M. Amoozadeh, A. Raghuramu, C. N. Chuah, D. Ghosal, H. Michael Zhang, J. Rowe, and K. Levitt, "Security Vulnerabilities of Connected Vehicle Streams and Their Impact on Cooperative Driving," *IEEE Communications Magazine*, vol. 53, no. 6, pp. 126–132, 2015.

[18] 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, Jul. 2014.

[19] M. A. Rahman, E. Al-Shaer, and R. B. Bobba, "Moving Target Defense for Hardening the Security of the Power System State Estimation," in *Proceedings of the First ACM Workshop on Moving Target Defense*, ser. MTD '14, New York, NY, USA: ACM, 2014, pp. 59–68.

[20] J. Tian, R. Tan, X. Guan, and T. Liu, "Hidden Moving Target Defense in Smart Grids," in *Proceedings of the 2nd Workshop on Cyber-Physical Security and Resilience in Smart Grids*, ser. CPSR-SG'17, New York, NY, USA: ACM, 2017, pp. 21–26.

[21] J. Zhao, G. Zhang, Z. Y. Dong, and K. P. Wong, "Forecasting-aided imperfect false data injection attacks against power system nonlinear state estimation," *IEEE Transactions on Smart Grid*, vol. 7, no. 1, pp. 6–8, 2016.

[22] P. Hespanhol, M. Porter, R. Vasudevan, and A. Aswani, "Dynamic watermarking for general LTI systems," in *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*, IEEE, 2017, pp. 1834–1839.

[23] A. Greenberg, "Hackers Remotely Kill a Jeep on the Highway—With Me in It," *Wired*, 2015.

[24] ——, "Hackers Cut a Corvette's Brakes Via a Common Car Gadget," *Wired*, 2015.

[25] T. E. Humphreys, B. M. Ledvina, V. Tech, M. L. Psiaki, B. W. O. Hanlon, and P. M. Kintner, "Assessing the Spoofing Threat: Development of a Portable GPS Civilian Spoofer," in *Proceedings of the 21st International Technical Meeting of the Satellite Division of The Institute of Navigation*, 2008, pp. 2314–2325.

[26] A. Rawnsley, "Iran's Alleged Drone Hack: Tough, but Possible," *Wired*, 2011.

[27] D. Hambling, "Ships fooled in GPS spoofing attack suggest Russian cyberweapon," *Wired*, 2017.

[28] D. Goward, "Chinese GPS spoofing circles could hide Iran oil shipments," *GPS World*, 2019.

[29] M. Burgess, "To protect Putin, Russia is spoofing GPS signals on a Massive scale," *Wired*, 2019.

[30] M. L. Psiaki and T. E. Humphreys, "GNSS Spoofing and Detection," *Proceedings of the IEEE*, vol. 104, no. 6, pp. 1258–1270, 2016.

[31] P. Y. Montgomery, T. E. Humphreys, and B. M. Ledvina, "Receiver-autonomous spoofing detection: Experimental results of a multi-antenna receiver defense against a portable civil GPS spoofer," in *Proceedings of the 2009 International Technical Meeting of The Institute of Navigation*, 2009, pp. 124–130.

[32] A. J. Kerns, D. P. Shepard, J. A. Bhatti, and T. E. Humphreys, "Unmanned Aircraft Capture and Control Via GPS Spoofing," *Journal of Field Robotics*, vol. 31, no. 4, pp. 617–636, 2014.

[33] J. Bhatti and T. E. Humphreys, "Hostile Control of Ships via False GPS Signals: Demonstration and Detection," *Navigation*, vol. 64, no. 1, pp. 51–66, 2017.

[34] A. Konovaltsev, M. Cuntz, C. Hättich, and M. Meurer, "Autonomous Spoofing Detection and Mitigation in a GNSS Receiver with an Adaptive Antenna Array," The Institute of Navigation, 2013.

[35] M. Psiaki and T. Humphreys, "Civilian GNSS Spoofing, Detection, and Recovery," in *Position, Navigation, and Timing Technologies in the 21st Century*. John Wiley & Sons, Ltd, 2020, ch. 25, pp. 655–680.

[36] J. Petit, B. Stottelaar, M. Feiri, and F. Kargl, "Remote Attacks on Automated Vehicles Sensors: Experiments on Camera and Lidar," *Black Hat Europe*, 2015.

[37] K. Eykholt, I. Evtimov, E. Fernandes, B. Li, A. Rahmati, C. Xiao, A. Prakash, T. Kohno, and D. Song, "Robust Physical-World Attacks on Deep Learning Visual Classification," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[38] Y. Cao, C. Xiao, B. Cyr, Y. Zhou, W. Park, S. Rampazzi, Q. A. Chen, K. Fu, and Z. M. Mao, "Adversarial Sensor Attack on LiDAR-based Perception in Autonomous Driving," in *Proceedings of the 26th ACM Conference on Computer and Communications Security (CCS'19)*, London, UK, 2019.

[39] "IEEE Standard for Information technology– Local and metropolitan area networks– Specific requirements– Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications Amendment 6: Wireless Access in Vehicular Environments," *IEEE Std 802.11p-2010*, pp. 1–51, 2010.

[40] "Evolved universal terrestrial radio access (e-UTRA) and evolved universal terrestrial radio access network (e-UTRAN); overall description; stage 2 (release 14)," *3GPP TS 36.300*, 2016.

[41] "IEEE Guide for Wireless Access in Vehicular Environments (WAVE) Architecture," *IEEE Std 1609.0-2019*, pp. 1–106, 2019.

[42] "IEEE Standard for Wireless Access in Vehicular Environments–Security Services for Applications and Management Messages," *IEEE Std 1609.2-2016*, pp. 1–240, 2016.

[43] T. Fei and W. Wang, "LTE Is Vulnerable: Implementing Identity Spoofing and Denial-of-Service Attacks in LTE Networks," in *2019 IEEE Global Communications Conference (GLOBECOM)*, 2019, pp. 1–6.

[44] M. Muhammad and G. A. Safdar, "Survey on existing authentication issues for cellular-assisted V2X communication," *Vehicular Communications*, vol. 12, pp. 50–65, 2018.

[45] A. A. Cárdenas, S. Amin, and S. Sastry, "Secure Control: Towards Survivable Cyber-Physical Systems," in *The 28th International Conference on Distributed Computing Systems Workshops*, Jun. 2008, pp. 495–500.

[46] S. Amin, A. A. Cárdenas, and S. Sastry, "Safe and Secure Networked Control Systems under Denial-of-Service Attacks," in *International Workshop on Hybrid Systems: Computation and Control*, Berlin, Heidelberg: Springer, 2009, pp. 31–45.

[47] Y. Liu, P. Ning, and M. K. Reiter, "False Data Injection Attacks Against State Estimation in Electric Power Grids," *ACM Transactions on Information and System Security (TISSEC)*, vol. 14, no. 1, 13:1–13:33, 2011.

[48] 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*, ser. HiCoNS '12, New York, NY, USA: ACM, 2012, pp. 55–64.

[49] R. M. Ferrari and A. M. Teixeira, "Detection and isolation of routing attacks through sensor watermarking," in *2017 Annual American Control Conference (ACC)*, May 2017, pp. 5436–5442.

[50] R. S. Smith, "A Decoupled Feedback Structure for Covertly Appropriating Networked Control Systems," *IFAC Proceedings Volumes*, vol. 44, no. 1, pp. 90–95, 2011.

[51] Y. Yuan and Y. Mo, "Security in cyber-physical systems: Controller design against Known-Plaintext Attack," in *54th IEEE Conference on Decision and Control (CDC)*, Dec. 2015, pp. 5814–5819.

[52] D. Umsonst, E. Nekouei, A. M. Teixeira, and H. Sandberg, "On the Confidentiality of Linear Anomaly Detector States," in *2019 Annual American Control Conference (ACC)*, Jul. 2019, pp. 397–403.

[53] X.-S. Si, W. Wang, C.-H. Hu, and D.-H. Zhou, "Remaining useful life estimation–a review on the statistical data driven approaches," *European journal of operational research*, vol. 213, no. 1, pp. 1–14, 2011.

[54] H. Lee, "Framework and development of fault detection classification using IoT device and cloud environment," *Journal of Manufacturing Systems*, vol. 43, pp. 257–270, 2017.

[55] Y. Lei, F. Jia, J. Lin, S. Xing, and S. X. Ding, "An intelligent fault diagnosis method using unsupervised feature learning towards mechanical big data," *IEEE Transactions on Industrial Electronics*, vol. 63, no. 5, pp. 3137–3147, 2016.

[56] W. Yu, T. Dillon, F. Mostafa, W. Rahayu, and Y. Liu, "A Global Manufacturing Big Data Ecosystem for Fault Detection in Predictive Maintenance," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 1, pp. 183–192, 2020.

[57] M. Toothman, B. Braun, S. J. Bury, M. Dessauer, K. Henderson, R. Wright, D. M. Tilbury, J. Moyne, and K. Barton, "Trend-Based Repair Quality Assessment for Industrial Rotating Equipment," *IEEE Control Systems Letters*, vol. 5, no. 5, pp. 1675–1680, 2021.

[58] M. Saez, F. Maturana, K. Barton, and D. Tilbury, "Anomaly detection and productivity analysis for cyber-physical systems in manufacturing," in *2017 13th IEEE Conference on Automation Science and Engineering (CASE)*, 2017, pp. 23–29.

[59] M. A. Saez, F. P. Maturana, K. Barton, and D. M. Tilbury, "Context-Sensitive Modeling and Analysis of Cyber-Physical Manufacturing Systems for Anomaly Detection and Diagnosis," *IEEE Transactions on Automation Science and Engineering*, vol. 17, no. 1, pp. 29–40, 2020.

[60] R. Isermann, *Fault-diagnosis applications: model-based condition monitoring: actuators, drives, machinery, plants, sensors, and fault-tolerant systems*. Springer Science & Business Media, 2011.

[61] M. Porter, A. Joshi, P. Hespanhol, A. Aswani, M. Johnson-Roberson, and R. Vasudevan, "Simulation and real-world evaluation of attack detection schemes," in *2019 Annual American Control Conference (ACC)*, IEEE, 2019, pp. 551–558.

[62] S. Weerakkody, O. Ozel, P. Griffioen, and B. Sinopoli, "Active detection for exposing intelligent attacks in control systems," in *2017 IEEE Conference on Control Technology and Applications (CCTA)*, Aug. 2017, pp. 1306–1312.

[63] S. Weerakkody, B. Sinopoli, S. Kar, and A. Datta, "Information flow for security in control systems," in *55th IEEE Conference on Decision and Control (CDC)*, Dec. 2016, pp. 5065–5072.

[64] Y. Mo, E. Garone, A. Casavola, and B. Sinopoli, "False data injection attacks against state estimation in wireless sensor networks," in *49th IEEE Conference on Decision and Control (CDC)*, Dec. 2010, pp. 5967–5972.

[65] Y. Mo and B. Sinopoli, "Integrity Attacks on Cyber-physical Systems," in *Proceedings of the 1st International Conference on High Confidence Networked Systems*, ser. HiCoNS '12, New York, NY, USA: ACM, 2012, pp. 47–54.

[66] C. Kwon, W. Liu, and I. Hwang, "Security analysis for Cyber-Physical Systems against stealthy deception attacks," in *2013 Annual American Control Conference (ACC)*, Jun. 2013, pp. 3344–3349.

[67] N. Hashemi and J. Ruths, "Generalized chi-squared detector for LTI systems with non-Gaussian noise," in *2019 Annual American Control Conference (ACC)*, Jul. 2019, pp. 404–410.

[68] C. Murguia and J. Ruths, "CUSUM and Chi-squared Attack Detection of Compromised Sensors," in *2016 IEEE Conference on Control Applications (CCA)*, Sep. 2016, pp. 474–480.

[69] D. Umsonst and H. Sandberg, "Anomaly Detector Metrics for Sensor Data Attacks in Control Systems," in *2018 Annual American Control Conference (ACC)*, Jun. 2018, pp. 153–158.

[70] S. Jajodia, A. K. Ghosh, V. Swarup, C. Wang, and X. S. Wang, *Moving target defense: creating asymmetric uncertainty for cyber threats*. Springer Science & Business Media, 2011, vol. 54.

[71] A. M. Teixeira, I. Shames, H. Sandberg, and K. H. Johansson, "Revealing stealthy attacks in control systems," in *2012 50th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, Oct. 2012, pp. 1806–1813.

[72] J. Giraldo, A. A. Cárdenas, and R. G. Sanfelice, "A Moving Target Defense to Detect Stealthy Attacks in Cyber-Physical Systems," in *2019 Annual American Control Conference (ACC)*, Jul. 2019, pp. 391–396.

[73] A. Kanellopoulos and K. Vamvoudakis, "Switching for Unpredictability: A Proactive Defense Control Approach," in *2019 Annual American Control Conference (ACC)*, Jul. 2019, pp. 4338–4343.

[74] S. Weerakkody and B. Sinopoli, "Detecting integrity attacks on control systems using a moving target approach," in *54th IEEE Conference on Decision and Control (CDC)*, Dec. 2015, pp. 5820–5826.

[75] C. Schellenberger and P. Zhang, "Detection of covert attacks on cyber-physical systems by extending the system dynamics with an auxiliary system," in *56th IEEE Conference on Decision and Control (CDC)*, Dec. 2017, pp. 1374–1379.

[76] M. Ghaderi, K. Gheitasi, and W. Lucia, "A Novel Control Architecture for the Detection of False Data Injection Attacks in Networked Control Systems," in *2019 Annual American Control Conference (ACC)*, Jul. 2019, pp. 139–144.

[77] P. Griffioen, S. Weerakkody, and B. Sinopoli, "An Optimal Design of a Moving Target Defense for Attack Detection in Control Systems," in *2019 Annual American Control Conference (ACC)*, Jul. 2019, pp. 4527–4534.

[78] Y. Mo and B. Sinopoli, "Secure Control Against Replay Attacks," in *2009 47th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, Sep. 2009, pp. 911–918.

[79] 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 (CDC)*, Dec. 2014, pp. 3757–3764.

[80] B. Satchidanandan and P. R. Kumar, "Dynamic Watermarking: Active Defense of Networked Cyber-Physical Systems," *Proceedings of the IEEE*, vol. 105, no. 2, pp. 219–240, 2017.

[81] ——, "Defending Cyber-Physical Systems from Sensor Attacks," in *International Conference on Communication Systems and Networks*, Springer, 2017, pp. 150–176.

[82] M. Hosseini, T. Tanaka, and V. Gupta, "Designing optimal watermark signal for a stealthy attacker," in *European Control Conference (ECC)*, Jun. 2016, pp. 2258–2262.

[83] Y. Mo, S. Weerakkody, and B. Sinopoli, "Physical Authentication of Control Systems: Designing Watermarked Control Inputs to Detect Counterfeit Sensor Outputs," *IEEE Control Systems Magazine*, vol. 35, no. 1, pp. 93–109, Feb. 2015.

[84] B. Satchidanandan and P. R. Kumar, "On the Design of Security-Guaranteeing Dynamic Watermarks," *IEEE Control Systems Letters*, vol. 4, no. 2, pp. 307–312, Apr. 2020.

[85] P. Hespanhol, M. Porter, R. Vasudevan, and A. Aswani, "Statistical watermarking for networked control systems," in *2018 Annual American Control Conference (ACC)*, IEEE, 2018, pp. 5467–5472.

[86] J. Rubio-Hernan, L. De Cicco, and J. Garcia-Alfaro, "Event-triggered watermarking control to handle cyber-physical integrity attacks," in *Nordic Conference on Secure IT Systems*, Springer, 2016, pp. 3–19.

[87] W.-H. Ko, B. Satchidanandan, and P. Kumar, "Dynamic Watermarking-based Defense of Transportation Cyber-physical Systems," *ACM Transactions on Cyber-Physical Systems*, vol. 4, no. 1, pp. 1–21, 2019.

[88] W. H. Ko, B. Satchidanandan, and P. R. Kumar, "Theory and implementation of dynamic watermarking for cybersecurity of advanced transportation systems," in *2016 IEEE Conference on Communications and Network Security (CNS)*, 2016, pp. 416–420.

[89] 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 (SmartGridComm)*, Oct. 2017, pp. 271–276.

[90] S. Weerakkody, O. Ozel, and B. Sinopoli, "A Bernoulli-Gaussian physical watermark for detecting integrity attacks in control systems," in *2017 55th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, Oct. 2017, pp. 966–973.

[91] R. M. Ferrari and A. M. Teixeira, "Detection and isolation of replay attacks through sensor watermarking," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 7363–7368, 2017.

[92] A. M. Teixeira and R. M. Ferrari, "Detection of Sensor Data Injection Attacks with Multiplicative Watermarking," in *European Control Conference (ECC)*, Jun. 2018, pp. 338–343.

[93] R. Romagnoli, S. Weerakkody, and B. Sinopoli, "A Model Inversion Based Watermark for Replay Attack Detection with Output Tracking," in *2019 Annual American Control Conference (ACC)*, Jul. 2019, pp. 384–390.

[94] P. Hespanhol, M. Porter, R. Vasudevan, and A. Aswani, "Sensor Switching Control Under Attacks Detectable by Finite Sample Dynamic Watermarking Tests," *IEEE Transactions on Automatic Control (TAC)*, 2020, To Appear.

[95] M. Porter, S. Dey, A. Joshi, P. Hespanhol, A. Aswani, M. Johnson-Roberson, and R. Vasudevan, "Detecting Deception Attacks on Autonomous Vehicles via Linear Time-Varying Dynamic Watermarking," in *2020 IEEE Conference on Control Technology and Applications (CCTA)*, IEEE, 2020, pp. 969–976.

[96] M. Porter, P. Hespanhol, A. Aswani, M. Johnson-Roberson, and R. Vasudevan, "Detecting Generalized Replay Attacks via Time-Varying Dynamic Watermarking," *IEEE Transactions on Automatic Control (TAC)*, 2020, To Appear.

[97] M. Olfat, S. Sloan, P. Hespanhol, M. Porter, R. Vasudevan, and A. Aswani, "Covariance-Robust Dynamic Watermarking," in *2020 IEEE 59th Annual Conference on Decision and Control (CDC)*, To Appear, 2020.

[98] W. Hess, D. Kohler, H. Rapp, and D. Andor, "Real-Time Loop Closure in 2D LIDAR SLAM," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, 2016, pp. 1271–1278.

[99] M. Porter, A. Joshi, P. Hespanhol, A. Aswani, M. Johnson-Roberson, and R. Vasudevan, "Simulation and real-world evaluation of attack detection schemes: Video," 2018. [Online]. Available: `www.roahmlab.com/acc2019_dynwatermark_video`.

[100] M. Porter, A. Joshi, S. Dey, Q. Wu, P. Hespanhol, A. Aswani, M. Johnson-Roberson, and R. Vasudevan, "Resilient Control of Platooning Networked Robotic Systems via Dynamic Watermarking," *arXiv preprint arXiv:2106.07541*, 2021.

[101] T. Anderson, *An Introduction to Multivariate Statistical Analysis*, 3rd ed., ser. Wiley Series in Probability and Statistics. Wiley, 2003.

[102] G. Grimmett and D. Stirzaker, *Probability and random processes*, 3rd ed. Oxford university press, 2001.

[103] D. R. Brillinger, *Time series: data analysis and theory*. Siam, 1981, vol. 36.

[104] P. Billingsley, *Probability and Measure*, 3rd ed. Wiley, 1995.

[105] E. Hendricks, O. Jannerup, and P. H. Sørensen, *Linear systems control: deterministic and stochastic methods*. Springer, 2008.

[106] D. Luenberger, "Observers for multivariable systems," *IEEE Transactions on Automatic Control*, vol. 11, no. 2, pp. 190–197, 1966.

[107] W. Henning, "User Guidelines for Single Base Real Time GNSS Positioning," *National Geodetic Survey*, Version 3.1, April 2014.

[108] Septentrio, *AsteRx SB ProDirect datasheet*, BBr: 06/2020, 2020.

[109] S. Ye, Y. Yan, and D. Chen, "Performance analysis of velocity estimation with BDS," *The Journal of Navigation*, vol. 70, no. 3, p. 580, 2017.

[110] VectorNav Technologies, *Industrial Series datasheet*, Version 12-0009-R3, 2017.

[111] C. Murguia and J. Ruths, "On Reachable Sets of Hidden CPS Sensor Attacks," in *2018 Annual American Control Conference (ACC)*, 2018, pp. 178–184.

[112] Y. Mo and B. Sinopoli, "On the Performance Degradation of Cyber-Physical Systems under Stealthy Integrity Attacks," *IEEE Transactions on Automatic Control*, vol. 61, no. 9, pp. 2618–2624, 2016.

[113] H. Hotelling, "Multivariate Quality Control," *Techniques of Statistical Analysis*, 1947.

[114] Y. Mo and B. Sinopoli, "False Data Injection Attacks in Control Systems," in *First Workshop on Secure Control Systems*, 2010, pp. 226–231.

[115] J. B. Lasserre, *Moments, Positive Polynomials and Their Applications*. 2010.

[116] M. M. Tobenkin, F. Permenter, and A. Megretski, *Spotless Library*, 2018. [Online]. Available: `http://github.com/spot-toolbox/spotless`.

[117] M. ApS, *The mosek optimization toolbox for matlab manual. version 8.1.* 2017. [Online]. Available: `http://docs.mosek.com/8.1/toolbox/index.html`.

[118] *Introduction to carsim.* 2021. [Online]. Available: `https://www.carsim.com/downloads/pdf/CarSim_Introduction.pdf`.

[119] A. Liniger, A. Domahidi, and M. Morari, "Optimization-based autonomous racing of 1: 43 scale RC cars," *Optimal Control Applications and Methods*, vol. 36, no. 5, pp. 628–647, 2015.

[120] E. Bakker, L. Nyborg, and H. B. Pacejka, "Tyre modelling for use in vehicle dynamics studies," *SAE Transactions*, pp. 190–204, 1987.

[121] A. Agarwal, A. Anders, A. Fishberg, C. Walsh, S. Karaman, and T. Henderson, *Mit-racecar*, `https://github.com/mit-racecar`, 2019.

[122] R. Rajamani, *Vehicle dynamics and control*. Springer Science & Business Media, 2011.

[123] C. Bonnet and H. Fritz, "Fuel consumption reduction in a platoon: Experimental results with two electronically coupled trucks at close spacing," SAE Technical Paper, Tech. Rep., 2000.

[124] A. A. Alam, A. Gattami, and K. H. Johansson, "An experimental study on the fuel reduction potential of heavy duty vehicle platooning," in *13th International IEEE Conference on Intelligent Transportation Systems*, 2010, pp. 306–311.

[125] K.-Y. Liang, J. Mårtensson, and K. H. Johansson, "Fuel-saving potentials of platooning evaluated through sparse heavy-duty vehicle position data," in *2014 IEEE Intelligent Vehicles Symposium Proceedings*, IEEE, 2014, pp. 1061–1068.

[126] B. McAuliffe, M. Croken, M. Ahmadi-Baloutaki, and A. Raeesi, "Fuel-economy testing of a three-vehicle truck platooning system," National Research Council Canada, Aerospace Aerodynamics Laboratory, Tech. Rep. LTR-AL-2017-000, 2017.

[127] C. C. Chien and P. Ioannou, "Automatic Vehicle-Following," in *1992 American Control Conference*, 1992, pp. 1748–1752.

[128] R. Hall and C. Chin, "Vehicle sorting for platoon formation: Impacts on highway entry and throughput," *Transportation Research Part C: Emerging Technologies*, vol. 13, no. 5-6, pp. 405–420, 2005.

[129] Y. Sugiyama, M. Fukui, M. Kikuchi, K. Hasebe, A. Nakayama, K. Nishinari, S.-i. Tadaki, and S. Yukawa, "Traffic jams without bottlenecks—experimental evidence for the physical mechanism of the formation of a jam," *New Journal of Physics*, vol. 10, no. 3, p. 033 001, 2008.

[130] M. R. Flynn, A. R. Kasimov, J.-C. Nave, R. R. Rosales, and B. Seibold, "Self-sustained nonlinear waves in traffic flow," *Phys. Rev. E*, vol. 79, p. 056 113, 5 2009.

[131] R. Herman, E. W. Montroll, R. B. Potts, and R. W. Rothery, "Traffic dynamics: analysis of stability in car following," *Operations research*, vol. 7, no. 1, pp. 86–106, 1959.

[132] K.-c. Chu, "Decentralized control of high-speed vehicular strings," *Transportation science*, vol. 8, no. 4, pp. 361–384, 1974.

[133] D. Swaroop and J. K. Hedrick, "String stability of interconnected systems," *IEEE transactions on automatic control*, vol. 41, no. 3, pp. 349–357, 1996.

[134] J. Ploeg, N. Van De Wouw, and H. Nijmeijer, "Lp string stability of cascaded systems: Application to vehicle platooning," *IEEE Transactions on Control Systems Technology*, vol. 22, no. 2, pp. 786–793, 2013.

[135] S. Feng, Y. Zhang, S. E. Li, Z. Cao, H. X. Liu, and L. Li, "String stability for vehicular platoon control: Definitions and analysis methods," *Annual Reviews in Control*, 2019.

[136] P. A. Ioannou and C.-C. Chien, "Autonomous intelligent cruise control," *IEEE Transactions on Vehicular technology*, vol. 42, no. 4, pp. 657–672, 1993.

[137] D. Swaroop, J. K. Hedrick, C. Chien, and P. Ioannou, "A comparision of spacing and headway control laws for automatically controlled vehicles," *Vehicle system dynamics*, vol. 23, no. 1, pp. 597–625, 1994.

[138] J. Zhou and H. Peng, "String stability conditions of adaptive cruise control algorithms," *IFAC Proceedings Volumes*, vol. 37, no. 22, pp. 649–654, 2004.

[139] S. S. Stankovic, M. J. Stanojevic, and D. D. Siljak, "Decentralized overlapping control of a platoon of vehicles," *IEEE Transactions on Control Systems Technology*, vol. 8, no. 5, pp. 816–832, 2000.

[140] G. J. Naus, R. P. Vugts, J. Ploeg, M. J. van De Molengraft, and M. Steinbuch, "String-stable CACC design and experimental validation: A frequency-domain approach," *IEEE Transactions on vehicular technology*, vol. 59, no. 9, pp. 4268–4279, 2010.

[141] J. Ploeg, D. P. Shukla, N. van de Wouw, and H. Nijmeijer, "Controller synthesis for string stability of vehicle platoons," *IEEE Transactions on Intelligent Transportation Systems*, vol. 15, no. 2, pp. 854–865, 2013.

[142] D. Swaroop and J. K. Hedrick, "Constant spacing strategies for platooning in automated highway systems," 1999.

[143] P. Seiler, A. Pant, and K. Hedrick, "Disturbance propagation in vehicle strings," *IEEE Transactions on Automatic Control*, vol. 49, no. 10, pp. 1835–1842, 2004.

[144] L. Xiao, F. Gao, and Jiangfeng Wang, "On scalability of platoon of automated vehicles for leader-predecessor information framework," in *2009 IEEE Intelligent Vehicles Symposium*, 2009, pp. 1103–1108.

[145] M. Porter, A. Joshi, S. Dey, Q. Wu, P. Hespanhol, A. Aswani, M. Johnson-Roberson, and R. Vasudevan, "Resilient control of platooning networked robotic systems via dynamic watermarking: Video," 2020. [Online]. Available: `www.roahmlab.com/tro2020_platoon_video`.

[146] X. Xu, J. W. Grizzle, P. Tabuada, and A. D. Ames, "Correctness Guarantees for the Composition of Lane Keeping and Adaptive Cruise Control," *IEEE Transactions on Automation Science and Engineering*, vol. 15, no. 3, pp. 1216–1229, 2018.

[147] P. Fancher, H. Peng, Z. Bareket, C. Assaf, and R. Ervin, "Evaluating the influences of adaptive cruise control systems on the longitudinal dynamics of strings of highway vehicles," *Vehicle System Dynamics*, vol. 37, no. sup1, pp. 125–136, 2002.