Physics-based early warnings for forensic readiness in industrial control systems

Physics-based Early Warnings for Forensic Readiness in Industrial Control Systems

Mazen Azzam

Lero - The Irish Software Research Centre

Department of Computer Science & Information Systems

Faculty of Science & Engineering

University *of* Limerick

Submitted to the University of Limerick for the degree of

PhilosophiæDoctor (PhD)

September 2022

Abstract

Attacks on safety-critical Industrial Control Systems (ICS) can cause significant physical damage to equipment. Forensic investigations are often used to understand how such attacks occurred. However, such investigations may be hindered by the loss of potential digital evidence due to physical damage. Therefore, ICS need to be *Forensic-Ready*, i.e. capable of preserving, in advance, data that could constitute evidence of a potential attack. A possible solution would be to collect and preserve all data generated by an ICS at all times. However, such a solution may not be feasible as it can result in a large amount of collected data that would require a large amount of time and storage resources to be analysed. Furthermore, not all data generated by an ICS may be relevant to forensic investigations, and its real-time collection could incur a control performance overhead. Thus, proactive data collection in ICS must (i) be triggered only at specific points in time and before a potential attack could cause damage; and (ii) must be restricted to data that could be relevant to forensic investigations, while ensuring it incurs little control performance overhead. To the best of our knowledge, most of the existing work on ICS forensics only considers post-incident investigations. Exceptions to this line of work propose to collect data in real-time whenever an attack is detected. However, the reliability of a detection-based trigger may be compromised if an attack on ICS can evade detection before causing damage. Such attacks are generally referred to as *stealthy attacks*. Additionally, no approach to identify data that could be relevant to the investigation of such attacks exists. In this thesis, we propose an approach for the engineering of *Forensic Readiness* in ICS faced with the threat stealthy attacks. Our contribution is twofold:

1. First, we propose an approach that can specify *when* to proactively collect potential evidence of a stealthy attack. Instead of relying on attack detection, we propose a framework for *physics-based Early Warning Systems (EWS)* that relies on predictive safety checks to warn about *potential damage* from a stealthy attack. We trigger data collection activities whenever a potential stealthy attack could cause damage; but the likelihood of such damage, as computed by the EWS, is too low to warrant a potentially expensive fail-safe. To the best of our knowledge, there is limited work on warning about stealthy attacks well before they are able to cause damage.
2. Second, we propose an approach to identify *which* data is relevant to forensic investigations of stealthy attacks on ICS. We consider data present in control devices at risk of being damaged to be relevant in our context. Such data may explain how the attacks that we are concerned with may have occurred. We identify this data using the safety checks performed by the EWS. Then, we propose a framework that decides on a subset of relevant data to collect by achieving a trade-off between the expected impact of an attack and the potential data collection overhead.

To evaluate our approach, we employ a virtual testbed based on the widely-used benchmark Tennessee-Eastman Process (TEP). We use extensive simulations of the testbed under attacks and with randomised parameters to demonstrate the accuracy and real-time performance of our EWS. In 90.8% of simulations, our technique correctly predicts damage from a stealthy attack with an execution time smaller than the system's sampling time. We also show that our EWS predicts such damage in situations where existing monitoring techniques fail to do so. Furthermore, simulations of the TEP under attacks demonstrate that our approach rarely (only in 5.8% of cases) misses any data that may be lost to physical damage. Moreover, we showcase a 35% reduction in control performance overhead as a result of the reduced

amount of collected data when compared with an “all-data” collection approach. Finally, we show a use case of our approach whereby it improved the efficiency of an existing live ICS forensic log analysis tool by an order of magnitude.

Declaration

I herewith declare that I have produced this thesis without the prohibited assistance of third parties and without making use of aids other than those specified; notions taken over directly or indirectly from other sources have been identified as such. This thesis has not previously been presented in identical or similar form to any other Irish or foreign examination board.

The thesis work was conducted from 2018 to 2022 at the University of Limerick under the supervision of Prof. Bashar Nuseibeh (University of Limerick), Dr. Liliana Pasquale (University College Dublin), and Prof. Gregory Provan (University College Cork).

Mazen Azzam

Limerick, 2022

Acknowledgements

First and foremost, I would like to thank my supervisors, *Prof. Bashar Nuseibeh*, *Dr. Liliana Pasquale*, and *Prof. Gregory Provan*. I am immensely grateful for their generous support, guidance, kindness, and inspiration that made this work possible, especially during the troubling times of lockdowns and the COVID pandemic.

I am also grateful to be part of two brilliant research groups, the *SPARE* (Security, Privacy, Adaptation and Requirements Engineering) and *SEAD* (Software Engineering and Design). I would like to thank each member of these groups for the insightful discussions and support over the years of my PhD studies, especially through our weekly meetings and yearly "Away Days".

I was also fortunate to be part of Lero - the Irish Software Research Centre. The administrative support, the coffee chats in the halls of the Tierney building, the summer outings, and the monthly virtual meetings during the pandemic have made my PhD journey a smooth and enjoyable one. I would like to thank *Dara O'Connor*, *Louise Connolly*, *Edel Collins*, *Susan Mitchell*, *Gerard Mulligan*, and *Denis Hogan*.

I was also fortunate to be part of the CONFIRM manufacturing research centre. Through this centre, I had the pleasure of engaging in insightful multidisciplinary discussions on cyber-physical systems and smart manufacturing which added value to my work.

I would like to thank *Prof. Ernesto Damiani* and *Dr. Jim Buckley*, my external and internal examiners, respectively. I am grateful for the time they took to read this dissertation and for their valuable corrections and suggestions. I would also like to thank *Prof. Brian Fitzgerald* for taking the time to chair my PhD viva voce.

Thanks to *Dr. Goetz Botterweck* for reviewing my research progress over the course of my studies, and for providing feedback that guided my work.

My research work was funded by the Science Foundation Ireland (SFI), under grant numbers 13/RC/2094_P2 and 16/RC/3918.

Last but not least, I would like to thank my mother *Lody*, my father *Wajih*, my sister *Mira*, my brother *Maher*, and my best friends — *Nael*, *Maria*, *Aya*, and *Chiara*. Their moral support and encouragement have kept my spirits high and helped me navigate the PhD journey successfully.

To my parents, Lody and Wajih

Contents

List of Tables

LIST OF TABLES

List of Figures

Glossary

ADS Anomaly Detection System

CPS Cyber-Physical System

CUSUM CUmulative SUm

DF Digital Forensics

DoS Denial-of-Service

EWMA Exponential Weighted Moving Average

EWS Early Warning System

FDIA False Data Injection Attack

FR Forensic Readiness

HVAC Heating Ventilation and Air Conditioning

ICS Industrial Control System

IDS Intrusion Detection System

IoT Internet of Things

IT Information Technology

LTI Linear Time-Invariant

MitM Man-in-the-Middle

MPC Model Predictive Control

NCS Networked Control System

PLC Programmable Logic Controller

RTU Remote Terminal Unit

SCADA Supervisory Control And Data Acquisition

SPRT Sequential Probability Ratio Test

TEP Tennessee-Eastman Process

UAV Unmanned Aerial Vehicle

Chapter 1

Introduction

Cyber-Physical Systems (CPS) augment physical systems with enhanced capabilities, such as real-time monitoring and dynamic control (Alur (2015)). In the past few decades, advances in computational capabilities have allowed these systems to become increasingly pervasive. Modern applications of CPS can be found in diverse fields, such as healthcare, transportation, electricity generation and distribution. Industrial Control Systems (ICS) are considered a sub-class of CPS, and denote several types of control systems used in industrial applications. In ICS, software controls and monitors safety-critical processes that constitute part of the critical infrastructure (Conti et al. (2021)).

While ICS were originally isolated from the Internet, their connection to enterprise systems in recent years has exposed several security vulnerabilities, and has led to an increase in attacks against such systems (Loukas (2015)). In 2017, a malware named *Triton* was discovered at a petrochemical facility in Saudi Arabia. This malware was able to manipulate safety systems into causing physical damage (Di Pinto et al. (2018)). In 2016, an attack on the Ukrainian power grid led to a blackout for hundreds of thousands of users, and in 2014, a spear-phishing attack on a steel mill furnace in Germany caused severe physical damage (Lee et al. (2014)). Earlier notable instances of attacks against ICS include the *Stuxnet* worm (Albright et al. (2011)) that infected Iranian nuclear plants and the Maroochy wastewater treatment plant breach in Australia (Slay and Miller (2007)).

Forensic investigations in ICS can be helpful to understand how attacks targeting ICS have taken place (van der Knijff (2014)). To perform such investigations, the collection of data generated by different components and that may explain how an attack took place is required. However, such data tends to be volatile, mainly due to the low computational and storage resources in control devices. Furthermore, attacks on ICS can cause physical damage to low-level equipment (i.e. those in vicinity to physical processes, like sensors and actuators) which may cause further data losses. ICS are normally equipped with process historians — servers that typically store diagnostics data about physical processes and used to inform process-related decisions (Altschaffel et al. (2019)). Diagnostics data, while it may guide the detection of potential faults or physical anomalies, may not necessarily be useful in explaining how a potential attack occurred. Therefore, such data may not be suitable for forensic investigations (Ahmed et al. (2017); Janicke et al. (2015)). For example, given a Programmable Logic Controller (PLC) — a proprietary computer solely used for the control of physical processes (Yau and Chow (2015)) — a process historian may store values of control variables received or computed by the PLC (e.g. temperature, valve percentage opening) and analyse trends over time. However, a historian may not store the code or configuration implemented on the PLC, a piece of data that could be used as evidence of attacks that involve PLC code modification (Ahmed et al. (2017); Janicke et al. (2015)).

Consequently, post-incident forensic investigations may be limited by the unavailability of *relevant* data, i.e. data that could constitute potential evidence (van Vliet et al. (2015)). Therefore, ICS need to be *Forensic-Ready* (Rowlingson et al. (2004)), i.e. capable of identifying, collecting, and preserving, *in advance*, data that may be used as evidence to investigate potential known incidents — should they occur (Alrajeh et al. (2017)). To prevent data and evidence loss by physical damage due to an attack, proactive data collection should be performed before such harm can take place. The collection of all data generated by an ICS at all times, a possible solution to engineering forensic-ready ICS, can negatively affect control performance and the tractability of forensic investigations due to the typically large volume of such data (Kiss et al. (2014)). Hence, to prevent data loss by damage, there is a need for defining a timely *trigger* for data collection, which should be fired well before an attack is able to cause damage. Second, to ensure the tractability of potential investigations, proactive data collection must only consider *relevant* data (Pasquale et al. (2018)). Third, real-time data collection must account for the control performance overhead that would result from potential logging operations.

The majority of work on ICS forensics (Mohamed et al. (2020)) is geared towards post-incident investigations and generic organisation-level guidelines for forensic readiness (Eden et al. (2016); van der Knijff (2014)). A few exceptions to this line of work can be found within ICS *live forensics*, where methods to investigate attacks as the system is running (Taveras (2013)) are proposed. These techniques rely on the detection of an attack in order to trigger data collection and live investigations. However, the reliability of such a trigger for data collection depends on the detectability of attacks. Attackers that are capable of evading existing Intrusion/Anomaly Detection Systems (IDS/ADS) may still be able to cause incidents that are hard to investigate, especially if such incidents cause physical damage.

In recent years, a large body of work on ICS security (Conti et al. (2021); Giraldo et al. (2018); Kaouk et al. (2019); Lun et al. (2019); Sánchez et al. (2019)) has focused on attacks that can cause physical damage before getting detected. These attacks are generally referred to as *stealthy attacks*. To avoid detection, it has been shown that resourceful and knowledgeable attackers can take advantage of certain control-theoretical properties (Bai et al. (2015); Pasqualetti et al. (2015)). For example, such attackers can conceal their manipulations of sensor measurements by exploiting the noise native to these measurements (Murguia et al. (2018)). Attackers can then manipulate sensors over a long period of time such that controllers slowly drive a physical process to an unsafe state. Existing approaches for stealthy attack detection face several practical limitations (Giraldo et al. (2018)). For example, setting anomaly detection thresholds arbitrarily low to detect stealthy attacks masked by measurement noise can cause a high number of costly false alarms (Urbina et al. (2016)). Furthermore, there is limited work on ensuring that an attack on ICS is detected well before it causes damage (Giraldo et al. (2018)). Hence, attack detection cannot be currently used as a reliable timely trigger for proactive data collection.

A possible alternative approach to provide timely triggers for proactive data collection lies within *Early Warning Systems* (EWS). EWS have been mainly proposed for traditional IT systems to assist IDS/ADS in detecting "low-and-slow" network attacks. EWS monitor the occurrence of suspicious and seemingly benign network events from a variety of heterogeneous sources, often called weak evidence. Differently from IDS/ADS, instead of waiting for clear evidence of an intrusion or an anomaly, EWS constantly generate predictions and advice on unfamiliar situations before any suspected breach can cause harm (Kalutarage, Shaikh, Lee, Lee and Kiat (2015)). EWS may not reveal attacks on their own, but can guide the selection of appropriate measures to detect potential intrusions, thus complementing existing IDS as a security solution (Ramaki and Atani (2016)). Although stealthy attacks affecting physical processes in an ICS resemble to a certain extent low-and-slow network attacks in IT, to the best of our knowledge, *no approach for designing EWS for ICS under such attacks exists.*

Predictive monitoring of the safety of ICS under attacks can be used as a way to warn about potential attacks and trigger data collection. A limited body of work exists in this area. In some works, the proposed monitoring is resource-intensive (Kwon and Hwang (2018)) and its real-time performance with more complex systems is not clarified. In other instances, safety monitoring does not consider stealthy attacks and assumes that raw measurements are not subverted (Coletta (2018)). Therefore, there is a need for an approach to design EWS in an ICS that can provide real-time predictions about potential harm and timely triggers for proactive evidence collection given the threat of stealthy attacks.

1.1 Problem Statement

The investigation of attacks on ICS faces several challenges; most notably the large amount of data that can constitute potential evidence and the fact that such attacks can cause physical damage and loss of potential evidence. To tackle these challenges, the proactive collection of data relevant to potential investigations is necessary. A solution consisting of collecting *all data* generated by an ICS at *all times* (including data that can explain how an attack occurred) would be infeasible in practice. Thus, our problem is composed of two main parts.

Due to the fact that attacks on ICS can cause physical damage to equipment and subsequent loss of potential evidence, the first part of the problem calls for a *timely trigger* for proactive data collection, i.e. well before a potential attack manages to cause damage. Existing techniques (e.g. Janicke et al. (2015); Taveras (2013)) for the investigation of potential attacks as the ICS is operating (live forensics) rely on the detection of a certain attack to trigger data collection. However, such a trigger may not be always reliable if an ICS is faced with the threat of a stealthy attack. The detection of such attacks may not happen within a sufficient time period before they manage to cause damage to components that may store potential evidence (Giraldo et al. (2018)). Existing techniques (Coletta (2018); Kwon and Hwang (2018)) that attempt to predict potential safety violations caused by attacks are either resource intensive or do not consider stealthy attacks.

Since not all data generated by an ICS may necessarily explain how an attack occurred, the second part of the problem is then concerned with the need to identify such *relevant* data. In the context of our work, and since we are concerned with attacks that cause physical damage, relevant data can be defined as data at risk of being lost to physical damage. Additionally, since ICS operate under strict safety requirements, the real-time collection of relevant data must not compromise a control system's performance. To the best of our knowledge, existing approaches for live forensics in ICS (Janicke et al. (2015); Taveras (2013)) do not consider these challenges.

1.2 Aim and Objectives

In this thesis, we aim to provide an approach for the engineering of *Forensic-Readiness* in safety-critical ICS presented with the threat of stealthy attacks that can cause physical damage. To address the challenges described previously, this thesis sets two main objectives:

1. Provide an alternative *trigger* to proactive data collection, in the form of *early warnings* about potential stealthy attacks being able to cause damage before being detected.

2. Identify ICS data that is *relevant* to potential forensic investigations, i.e. data that is at most risk of being lost due to damage by an attack, and decide on a subset of relevant data to collect, by achieving a trade-off between potential attack impact and the cost of data collection.

1.3 Overview

To meet the objectives mentioned previously, we adopt a model-based approach. Our choice for such an approach stems from the fact that we are concerned with attacks that affect the behaviour of physical processes in an ICS. Existing work (Hadžiosmanović et al. (2012)) has shown that adequately addressing such attacks requires the incorporation of knowledge about physical processes. While such knowledge can be obtained by analysing process data and building statistical models, data about attacks that can cause damage can be difficult to obtain from real-life systems due to the potential safety issues associated with simulating such attacks. Furthermore, empirical evidence (Erba and Tippenhauer (2020)) has shown that data-driven solutions to stealthy attacks may not be as effective as methods that rely on first principles. Consequently, our approach considers a widely-used (Lun et al. (2019)) modelling framework for physical processes in an ICS — *Linear Time-Invariant* (LTI) systems. In this section, we summarise the approach that we propose in this thesis and the manner by which we evaluate it.

1.3.1 Approach

Objective 1. To provide early warnings about potential stealthy attacks that can cause damage to the system, we propose a framework for physics-based *Early Warning Systems* (EWS), which can complement the existing anomaly detector in a given system. The proposed framework performs predictions of potential (attack-induced) damage that could cause data loss based on the physical behaviour of the system. Differently to anomaly detection, where typically a certain deviation from normal behaviour is sought to potentially reveal an intrusion, our physics-based EWS assumes that the system is going to be subjected to a stealthy attack starting at the current moment in time. Then, given a model of such attacks, the physics-based EWS computes the likelihood that such an attack may cause damage in the future before the existing anomaly detector is able to reveal the resulting anomalies. We refer to this likelihood as the *Suspicion Metric*. The computation of this metric relies on attempting to predict potential harm *if* the system is subjected to a stealthy attack. Consequently, warnings issued by such a metric do not necessarily imply that the system is indeed under attack. This is in contrast to traditional anomaly detection, where an alarm may mean that a system is currently subjected to an attack or a fault, with a probability equal to the true alarm rate. A warning issued by our physics-based EWS however only means that *if* the system is subjected to a stealthy attack, then physical damage may be possible, with a probability equal to the computed "suspicion metric". Such a warning can be used as a trigger for further data collection, alternative to alarms from the existing anomaly detector.

Our framework computes the suspicion metric by defining two preliminary indicators, or what we call *Grounds for Suspicion*: (i) *Feasibility* of a stealthy attack, defined as its ability to cause damage to the system before it gets detected given the current physical state and the physics governing the system's behaviour; and (ii) the *Proximity* of the system to a predefined unsafe operating region (e.g. exceeding temperature limits). We then use reachability analysis to aggregate these indicators into the suspicion metric. Based on this metric, warnings of (at least) two degrees of criticality — high and low — can be raised. A low criticality warning suggests that *if* the system is subjected to a stealthy attack, then there is a low likelihood of damage happening before the attack gets detected, and that operators have sufficient time to collect more data. We then consider such a warning to be a suitable trigger for proactive data collection. A high criticality warning suggests that *if* the system is subjected to a stealthy attack, then the likelihood of damage happening within a relatively short period of time is high. This implies the need for more immediate measures, such as switching to backup controllers or activating an existing fail-safe, which we do not tackle in this work.

To compute the suspicion metric, providing measures for feasibility and proximity as defined previously requires the use of reachability analysis tools that are specific to the type of system at hand. Since different types of control systems and associated modelling formalisms (e.g. continuous-time, hybrid, discrete) exist, we described our framework for physics-based EWS independently from the specific modelling formalism. Instead, we provided *Soundness Principles* to guide the instantiation of the framework given a certain system and stealthy attack model. These principles seek to ensure that the suspicion metric is a sound measure of the proposed grounds of suspicion, and provides meaningful warning thresholds that may guide proactive data collection.

To demonstrate the applicability of the proposed framework, we applied it to a popular class of systems — Linear Time-Invariant (LTI) systems. We adapted existing results in reachability analysis under a practical class of stealthy attacks (Murguia and Ruths (2018)) to design a sound suspicion metric. We proposed a predictive online safety monitoring algorithm for LTI systems under stealthy attacks. The algorithm can provide real-time safety checks of whether the current state of the system can be driven into an unsafe set of states through an attack that can evade the existing anomaly detector. We used the size of the intersection (if non-empty) of the reachable set as a measure of feasibility, while the time taken to reach this intersection as a measure of proximity. We then combined these measures to provide a sound suspicion metric with clearly defined thresholds. We applied existing results in ellipsoid calculus (Dai and Pelckmans (2012); Kurzhanski and Varaiya (2000)) to enable efficient real-time computation of these measures.

Objective 2. To identify *relevant* ICS data, which we define in our context as data that is at risk of being lost, we propose a technique based on the safe Process Plant Layout (PPL) principles (Quiroz-Pérez et al. (2021)). PPL is a well established method to design the geographical layout of large-scale ICS where equipment are divided into different areas such that damage in one area has a low likelihood of spreading to others. Since such layout plans with associated safety constraints are available in a large-scale ICS by design, applying PPL to identify relevant data is a sound approach that requires little effort to set up. By mapping the safety constraints that we would predict to be violated to the specific concerned areas, we determine data sources (control devices and network connections) and specific data that may be lost to damage. To reduce the control performance overhead that would result from real-time data collection, we propose a framework to select which subset of the relevant data is "worth" collecting. In this framework, we use the suspicion metric to compute the expected reduction in an ICS' revenue should damage from a stealthy attack take place. We then only collect data whose collection cost is below this expected reduction in revenue. The result is a subset of relevant data to collect leading to a reduced control performance overhead that may result from real-time collection.

1.3.2 Evaluation

To evaluate our approach, we built a virtual testbed based on the benchmark Tennessee-Eastman Process (TEP). The TEP is a benchmark process proposed by Downs and Vogel (1993) and is based on a real-life process system. The TEP has been used extensively to study problems in the process control field, and more recently in the context of ICS security (Cárdenas et al. (2011); Geng et al. (2019); Genge and Siaterlis (2014); Krotofil and Cárdenas (2013); McEvoy and Wolthusen (2011*b*)). We implemented in MATLAB/Simulink a modified version of the simulation proposed by Bathelt et al. (2015) which is based on the control architecture proposed by Ricker Ricker (1996). The modifications included the addition of an anomaly detector, a high-level simulation of network behaviour, and nodes to simulate attacks, as well as the implementation of the proposed approach itself.

First, to evaluate the accuracy of the proposed physics-based EWS for LTI systems, we performed a large number of simulations under attacks conforming to the threat model that we consider. In each of these simulations, we checked whether the attack managed to cause damage to the system before being detected by the anomaly detector. Then, we checked whether the proposed algorithm raised a warning before the damage happened. Depending on whether the algorithm's predictions matched the result of the simulated attack, we computed the rates of false/true positives/negatives. We also used numerical simulations to compare our approach with similar existing work, specifically work that relies on pure proximity measures that use raw sensor measurements (Carcano et al. (2011); Castellanos and Zhou (2019); Coletta (2018); Etigowni et al. (2018)). We showed through different scenarios that the system can be made to appear as though it is moving away from unsafe operating regions, thus tricking existing proximity-based techniques. However, the proposed approach can still raise warning in these scenarios as it relies on reachable sets under stealthy attacks. Additionally, we used attack scenarios to evaluate our suspicion metric. In particular, we showed that warnings are issued by the metric well before any harm can take place. Furthermore, the scenarios illustrated how the criticality of the warnings (i.e. low and high) are consistent with how "fast" an attacker can cause damage. This result also showed that our choice of a low criticality warning as a trigger for proactive data collection is appropriate.

Second, we evaluated the real-time performance of the proposed algorithm by considering two aspects: the scalability of its execution time with respect to different parameters and its effect on communication delays between important control components. Using extensive simulations, we particularly assessed how the execution time scales with respect to the complexity of the set of unsafe states by introducing additional hypothetical safety constraints. We also assessed scalability of the execution time with respect to the predictive component of the suggested algorithm. To evaluate real-time performance, we augmented the TEP with a simulation of networked behaviour of different components using the MATLAB/Simulink Truetime library (Cervin et al. (2003)). This implementation has allowed us to measure the effect of introducing our scheme on the time delay in communication between critical parts of the TEP.

Finally, we considered in our evaluation the components associated with the identification and collection of relevant data. Namely, we evaluated the relevance of the collected data, the reduction in the control performance overhead caused by real-time data collection, and the applicability of our data collection approach in terms of supporting live forensic investigations. Our evaluation again relied on simulations of attacks conforming to the threat model that we considered for our physics-based EWS. In these simulations, we particularly inspected whether our approach missed collecting any relevant data. We also compared the resulting overhead on the control performance to the case where an "agnostic" collection of all data at all times is enabled. Furthermore, we demonstrated a use case of this approach where we showed its potential in improving the efficiency of log-based live Programmable Logic Controller (PLC) forensics analysis approaches (Yau and Chow (2015)).

1.4 Research Contributions

The primary research contribution of the present work is a general approach to engineering forensic readiness in ICS faced with the threat of stealthy attacks seeking to cause physical damage. The novelty of the work lies in the combination of the following:

1. A framework for physics-based Early Warning Systems (EWS) for ICS under potential stealthy attacks affecting the physical behaviour of the system. The framework is described in a fashion independent of the modelling formalism employed to represent the concerned system. A set of soundness principles is proposed to guide the instantiation of the framework to a particular system.

   The following contributions are related to this first component of our approach:

   (a) An efficient, polynomial-time online monitoring algorithm for systems that can be approximated using an LTI model. This algorithm particularly assesses whether the current state of the system can be taken to an unsafe state via an attack that satisfies a practical and widely studied model of stealthy attacks. The efficiency and real-time performance of the algorithm is guaranteed by incorporating results from ellipsoidal calculus. Extensive simulations have shown that the algorithm can correctly predict, in 90.8% of the time, that if the system is subjected to a stealthy attack, then the latter will be able to cause damage. Furthermore, the algorithm can perform these safety checks with an acceptable real-time performance, i.e. within a time period well below the system's sampling time. This execution time scales linearly with the number of safety constraints considered and the length of the prediction horizon. In contrast to existing work, the proposed algorithm can warn about damage from a potential stealthy attack even if the latter forces the system to appear as moving away from unsafe operating modes.

(b) An online security metric for LTI systems that reflects the probability that a potential stealthy attack may be able to cause damage, which we call the *suspicion metric*. The metric combines two components: a measure of *feasibility* of a stealthy attack defined as its ability to cause physical damage before getting detected and based on the size of the reachable set's intersection with the set of unsafe states; and a *time-to-unsafe-states* (proximity) measure that approximates the shortest time a potential stealthy attack would need to cause damage. Warnings based on this metric are raised well before a potential attack is able to cause damage, and in a manner that can soundly guide the selection of post-warning measures, including proactive data collection.

2. An approach to proactively identify and collect a subset of relevant data that can constitute potential evidence of a stealthy attack on an ICS. This component of our overall approach to FR can be broken down into two separate contributions:

   (a) An automated safety-driven technique that leverages the safe Process Plant Layout (PPL) to identify data that is likely to be lost (i.e. relevant data) to damage by a stealthy attack on a geographically-distributed ICS. Using the PPL available by design, we map the safety constraints predicted to be violated by a stealthy attack to specific areas of the plant that may be damaged as a result. We then identify data originating from devices and network connections in these areas as relevant. Our empirical evaluation has shown that the proposed technique misses such data in only 5.8% of cases.

   (b) A framework to select a subset of relevant data to be proactively collected based on a trade-off between the potential impact of a stealthy attack and the cost of performing data collection. Using the computed suspicion metric, we compute the expected loss in revenue resulting from a potential stealthy attack. We then use this loss to determine the value of collecting more data that may reveal a potential intrusion, and we only collect relevant data with a collection cost below this value. Empirical evaluation of this component on the TEP has demonstrated a 35% improvement in controller performance when compared to the case where an "all-data" collection approach is used. Furthermore, we assess the applicability of our approach in the case where one employs a class of live forensics methods consisting of checking whether a PLC's code is maliciously modified. Our approach to data collection can reduce the code checking execution time by an order of magnitude.

1.5 Thesis Outline

The remainder of thesis is organised along the following chapters:

- *Chapter 2 - Background and Literature Review*: we provide a narrative review of existing literature, highlighting the identified gaps and the resulting research questions which guided the work in this thesis.
- *Chapter 3 - Forensic Readiness of Industrial Control Systems under Stealthy Attacks*: we present an overview of the work performed in this thesis, and we motivate it with an example. We also describe our general framework for physics-based EWS in ICS faced with stealthy attacks.
- *Chapter 4 - Online Safety Monitoring and Early Warnings for Stealthy Attacks*: we propose an online safety monitoring algorithm for ICS under stealthy attacks; particularly for systems that can be represented using an LTI model. We design this algorithm according to the framework proposed in the previous chapter, and we propose a suspicion metric reflecting the probability of damage taking place if the system is subjected to a potential stealthy attack.
- *Chapter 5 - Safety-Driven Collection of Potential Evidence of Stealthy Attacks*: we propose our approach for proactive collection of data that can constitute evidence in ICS presented with the threat of stealthy attacks.

We extend our EWS framework with an automated technique to identify data relevant to forensic investigations in our context (i.e. data at risk of being lost to damage). We also present a framework to select a subset of data in a manner that reduces the overhead on control performance by trading-off collection cost and expected attack impact.

- *Chapter 6 - Evaluation*: we discuss the evaluation of the research contributions presented in this thesis. We employ a virtual testbed — the Tennessee-Eastman Process (TEP) — to evaluate different aspects of our work, such as accuracy and real-time performance of the online safety monitoring algorithm, and the relevance of the proactively collected data.
- *Chapter 7 - Conclusions and Future Directions*: we summarise and conclude the thesis and suggest ideas for future work based on the present contributions.

1.6 Publications

Most of the work described in this thesis was published or accepted with minor revisions, and we are working towards publishing the final part: the approach to proactively identify relevant data and decide on a subset of such data to collect through a trade-off between attack impact and data collection cost. The publications are as follows:

- M. Azzam, L. Pasquale, G. Provan and B. Nuseibeh, "Grounds for Suspicion: Physics-based Early Warnings for Stealthy Attacks on Industrial Control Systems", in *IEEE Transactions on Dependable and Secure Computing*, doi: 10.1109/TDSC.2021.3113989.

  This paper describes our framework for physics-based EWS in ICS faced with the threat of stealthy attacks along with an instantiation of the framework for LTI systems. In addition, we design a suspicion metric for LTI systems that measures the probability that a stealthy attack could cause damage.

- M. Azzam, L. Pasquale, G. Provan and B. Nuseibeh, "Efficient Predictive Monitoring of Linear Time-Invariant Systems Under Stealthy Attacks", accepted with minor revision in *IEEE Transactions on Control Systems Technology*.

  This paper describes in detail our design for the predictive online safety monitoring algorithm for ICS faced with stealthy attacks.

- M. Azzam, L. Pasquale, G. Provan and B. Nuseibeh, "Forensic Readiness of Industrial Control Systems Under Stealthy Attacks", under submission.

  This paper extends our physics-based EWS with components that can automatically identify relevant data, and select a subset of such data to collect by trading-off attack impact and collection cost.

The author of this thesis dissertation is the main author of all three publications. In all cases, this thesis should be regarded as the definitive version of the work.

Chapter 2

Background and Literature Review

In this chapter, we perform a narrative review of the research areas where we identified gaps that have shaped the main research questions and objectives of the current work. We begin the chapter with an overview of the main related areas and their intersections, along with a description of the positioning of the present work, before moving to a more detailed review of related work in each area.

2.1 Introduction

The work described in this thesis lies at the intersection of three main research areas: (1) Digital Forensics (DF), (2) Cyber-Physical /Industrial Control Systems (CPS/ICS), and (3) IT security. The DF area is concerned with the primary aim of this thesis, which is the engineering of Forensic Readiness (FR) in ICS. The CPS/ICS area is mainly concerned with our ICS application domain and the tools that we use to achieve our objectives. The IT security area is related to our first main objective, particularly in what concerns providing early warnings against attacks on ICS. Figure 2.1 shows a diagram of the main related areas which are the focus of the present review, along with their different intersections and the research questions that we formulated within.

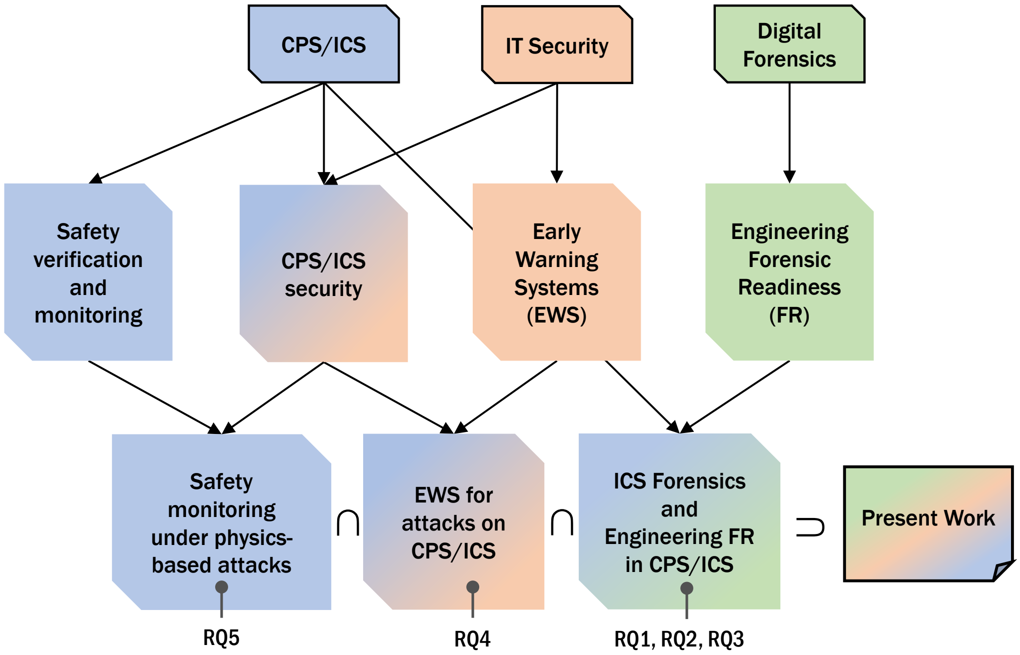


Figure 2.1: Research areas and sub-areas reviewed in this chapter, along with their different intersections of interest. An arrow from area/sub-area A $\rightarrow$ to area/sub-area B means that B is a sub-area of A. If A $\rightarrow$ B and C $\rightarrow$ B then B $\subseteq$ A $\cap$ C.

Within each of these main areas, we identified the main sub-areas and intersections that formed the focus of our review:

- Within DF, the main sub-area of concern is the Engineering of FR, as it relates directly to our main aim.
- Within CPS/ICS, two sub-areas are of concern: the first is CPS/ICS security, which shares some work related to the main area of IT security, and is related to the main motivation behind this work: stealthy attacks affecting physical processes in CPS/ICS. The second sub-area of concern is safety verification and monitoring, which relates to the technique that we employ in achieving the first main objective of this thesis.
- As for the area of IT security, in addition to works shared with CPS/ICS security, our review was mainly concerned with the sub-area of Early Warn-

ing Systems (EWS). We explored existing techniques used to provide early warnings against attacks on IT infrastructure that are mainly characterised with a resourceful adversary, and a "low-and-slow" approach to achieve adversarial goals without getting detected. As the stealthy attacks that we are concerned with in ICS feature similar characteristics, a review of existing approaches to EWS in the context of the more mature field of IT security is important in order to approach the relatively new application domain of CPS/ICS.

Our review has identified gaps and formulated research questions within the intersections of these areas.

(DF ∩ CPS/ICS). At the intersection of DF and CPS/ICS, we reviewed existing work on ICS forensics and on engineering forensic-ready ICS. Namely, we investigated the features that set ICS forensics apart from the more traditional IT applications of DF. We also discussed the small body of work on supporting FR in ICS. Through this review, we highlighted the main gap in the literature that motivated the work in this thesis, which is the limited work on engineering FR in ICS, particularly the lack of stealthy attacks considerations. We formulated three research questions based on our review of this area:

- [**RQ1**] *When should we trigger proactive data collection activities in ICS given that anomaly detectors may not raise an alarm about a stealthy attack in a sufficient period of time (i.e. well before the attack is able to cause damage)?*
- [**RQ2**] *How can we identify data that is relevant to a potential investigation of a stealthy attack in ICS?*
- [**RQ3**] *How can we select a subset of relevant data such that the control performance overhead resulting from real-time data collection is reduced?*

(CPS/ICS ∩ IT security). To better understand how stealthy attacks affect the FR of ICS, and to explore techniques that may help us in answering [**RQ1**], we performed a review of the literature on CPS/ICS security. This area also includes work related to IT security, and therefore can be considered to lie at the intersection of the two main areas of CPS/ICS and IT security. We reviewed existing IT security and physics-based techniques to secure CPS/ICS. This review has clarified the limitations to stealthy attack detection in ICS, and particularly highlighted the lack of techniques that can detect such attacks well before they can cause damage. Second, we reviewed the limited number of existing approaches to EWS in CPS/ICS **(CPS/ICS security ∩ EWS)** to understand current progress in this area and the limitations of EWS applications for CPS/ICS security. Based on the review of this area, we formulated the following research question:

- [**RQ4**] *How can we warn about stealthy attacks on ICS well before they are able to cause any harm?*

(CPS/ICS security ∩ safety verification). At the intersection of the sub areas of safety verification/monitoring and CPS/ICS security, we discussed the limited body of work on monitoring the safety of ICS under attacks affecting physical behaviour. This body of work is related to the technique that we employ to achieve our first main objective. We highlight a lack of efficient and scalable online safety monitoring approaches for ICS faced with stealthy attacks. This review resulted in the following research question:

- [**RQ5**] *How can we perform efficient and scalable online safety monitoring under the threat of stealthy attacks in ICS?*

The rest of this chapter is organised as follows: Section 2.2 provides a brief background on CPS/ICS, specifically the features that sets them apart from traditional IT systems, and the implications of these features on security in ICS. In Section 2.3, we review work within the area of DF and FR. Section 2.4 reviews work on CPS/ICS security, in addition to the area of EWS, where we highlight the features that distinguish them from traditional Intrusion Detection Systems (IDS). We review existing techniques to monitor the safety of ICS under attacks in Section 2.5. We finally conclude and summarise the discussions of this chapter in Section 2.6.

2.2 Cyber-Physical and Industrial Control Systems (CPS/ICS) Preliminaries

Before reviewing the areas of CPS/ICS forensics and security, we provide in this section a brief background on CPS/ICS. To understand the need for security and forensics approaches tailored for CPS/ICS, it is necessary to understand the differences between these systems and traditional IT systems. We describe these differences in this section, and we highlight the implications of these differences on how problems related to security and forensics are treated in CPS/ICS.

CPS can be generally defined as systems where software controls and monitors a certain physical process (Alur (2015)). With the expansion of the computational capabilities of computers in the past two decades, CPS has become a new computing paradigm with a variety of applications: smart manufacturing, medical devices, wearables, autonomous vehicles, smart grids, etc. (Alguliyev et al. (2018)). Industrial Control Systems (ICS) can be considered a subclass of CPS, with applications mainly in the industry, and with physical processes that are often safety-critical and that form part of a nation's critical infrastructure (Kaouk et al. (2019); Zhu et al. (2011)). Examples include chemical processing systems, manufacturing, power generation and distribution, transportation systems — among others.

Figure 2.2 shows a diagram of the most common network architecture of a typical ICS, the Supervisory Control and Data Acquisition (SCADA) standard. In this typical architecture, networked components can be grouped along different levels (Cárdenas et al. (2008); Kaouk et al. (2019)). At the lowest level (Level 0) lie the physical processes being controlled and the devices used to interface with these processes, such as sensors and actuators. These devices enable the exchange of data between the physical processes and regulatory controllers. Control algorithms are typically installed on proprietary Programmable Logic Controllers (PLC's), with Remote Terminal Units (RTU's) acting as an interface with these devices (Level 1). Operator workstations reside at the next level (Level 2) where supervisory control is performed, which consists mainly of varying the set-points for physical quantities of interest. At Level 3, high-level management of different manufacturing operation is performed, such as scheduling different activities, logging physical process data in historians, and optimising different aspects of production. The ICS is then connected to the corporate enterprise system (Level 4) where, similarly to traditional IT systems, high-level decisions about business-related activities are undertaken.

ICS feature several characteristics that differentiate them from traditional IT systems (Table 2.1). The most important of which lies in the main design ob-

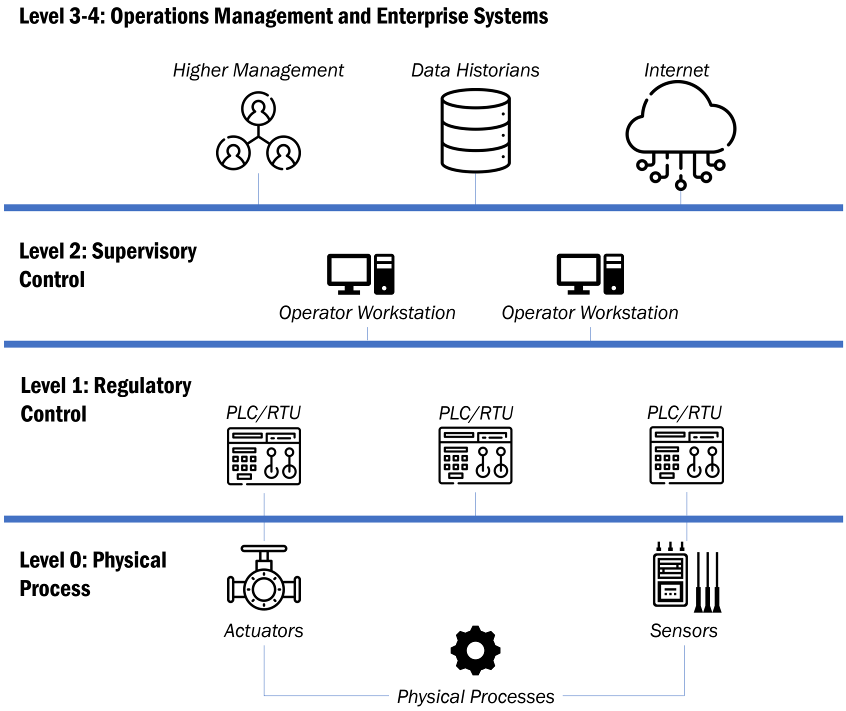


Figure 2.2: Standard Supervisory Control And Data Acquisition (SCADA) architecture for an Industrial Control System (ICS) (PLC = Programmable Logic Controller, RTU = Remote Terminal Unit).

jective. Due to ICS operations potentially affecting the physical world, these systems primarily focus on the safety of individuals and equipment while satisfying some liveness requirements (Zhu et al. (2011)). However, IT systems are mainly designed for information storage and protection from misuse. Due to the safety criticality of ICS, hard real-time requirements are generally imposed on different control operations. In contrast, IT systems can generally tolerate latency (Kaouk et al. (2019); Zhu and Başar (2011)). ICS are also designed to operate continuously for long periods of time, hence, they are less amenable to periodic software updates and system reboots (Cárdenas et al. (2008)). Additionally, due to hard real-time requirements, software patching in ICS could involve a lengthy and costly operation, and consequently it may not be economically viable. Tra-

Table 2.1: Main differences between Industrial Control Systems (ICS) and traditional IT systems.

	ICS	**Traditional IT Systems**
Main Design Objective	Safe operation of physical processes	Security; information storage and protection
Real-time Performance	Hard real-time requirements	Latency generally tolerated
Lifetime	$>$ 10 years	$<$ 5 years
Software Updates	Infrequent, requires significant planning	Periodic updates and patches

ditional IT systems on the other hand have significantly shorter lifespans where updates and software maintenance activities are conducted regularly.

The peculiarities of ICS have significant consequences on the approaches used to secure these systems. At a fundamental level, their distinctive main design objective imposes a reordering in terms of priority of the commonly desirable security properties (Conti et al. (2021); Zhu et al. (2011)). Traditionally, security in IT systems has revolved around the "*C-I-A*" triad: *Confidentiality* referring to the restriction of access for unauthorised individuals; *Integrity* referring to the prevention of unauthorised modifications of data; and *Availability* referring to the requirement that data should be accessible to authorised users when needed (Anderson (2020)). In ICS, the ordering of these properties is reversed, with Availability and Integrity being prioritised over Confidentiality due to the safety-criticality of the physical processes. Controllers in an ICS need to access sensor measurements within hard-real-time requirements, and computational resources need to be constantly available to execute control algorithms. Controllers also need to access accurate and unmodified measurements in order to execute safe and correct feedback control to maintain a certain set-point (Cárdenas et al.

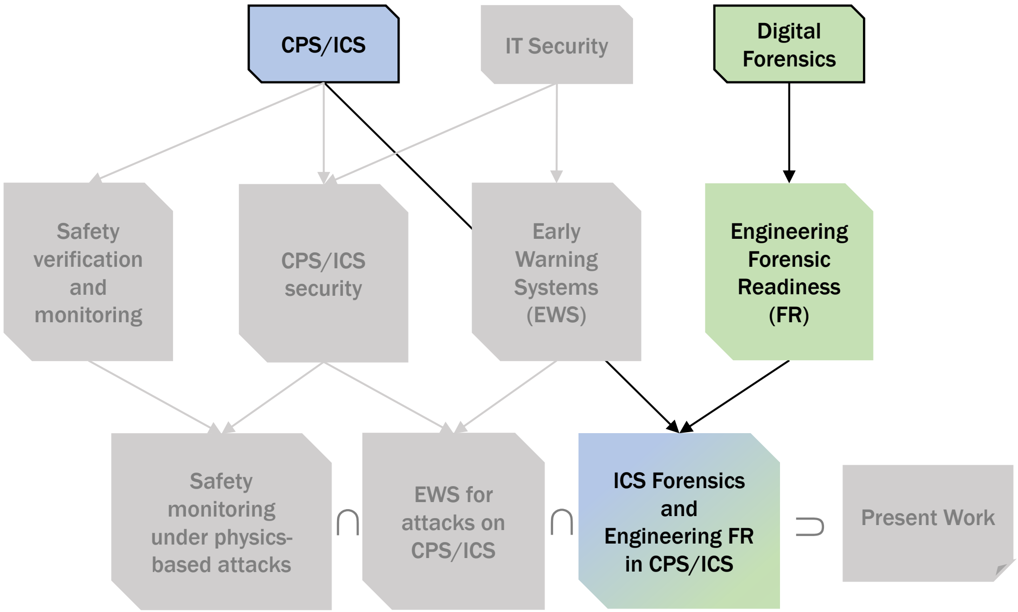


Figure 2.3: Research areas reviewed in Section 2.3.

(2008); Kaouk et al. (2019); Zhu and Başar (2011)).

In addition to the reversal of security priorities, security approaches to ICS systems must accommodate for the difficulty of performing software patching to ICS components. Moreover, these approaches must also account for the hard-real-time requirements implying the need to minimise interference with different regulatory control operations. This reduces the feasibility of solutions such as encryption as they may cause intolerable latency in control communications. Finally, within the control network (Levels 0-3, Figure 2.2), proprietary industrial communication protocols such as Modbus, Profibus, OPC, DNP3 — among others, are designed without any meaningful security consideration (Kaouk et al. (2019)).

2.3 Digital Forensics and Forensic Readiness

In the previous section, we provided some background on CPS/ICS, particularly regarding the features that distinguish them from traditional IT systems, and we highlighted the implications of these differences on how security is approached in these systems. In this section, we discuss how the field of ICS forensics is also affected by these differences. We begin with a brief overview of the general field of Digital Forensics (DF) and approaches to engineer Forensic Readiness (FR) in traditional IT systems. We then review the relatively small body of existing work applied to CPS/ICS (Figure 2.3).

2.3.1 Traditional IT/Software Systems

DF refers to the processes employed to collect, preserve and interpret digital evidence to reconstruct how an incident occurred (Rowlingson et al. (2004)). These processes have gained attention over the past few decades with the increase in the number and complexity of cases that investigators struggle to handle. This is due first to the increase in the number and variety of digital devices connected to the internet. Second, the complexity of the systems that need to be investigated has increased as well. The advent of CPS and Internet of Things (IoT) devices for example has lead investigators to consider cyber-physical interactions that haven't been observed before in traditional IT/software systems (Alrimawi et al. (2017)).

In the majority of existing work on digital forensic processes, little attention is given to the factors that influence the object of investigation and the associated evidence prior to the digital investigation process (Rowlingson et al. (2004)). A large portion of important evidence stored in digital devices however tends to be volatile, e.g. data stored on RAM, and may be vulnerable to tampering by an offender (Pasquale et al. (2018)). The ability of investigators to reconstruct how a certain incident happened is greatly dependent on the quality of the evidence and the ability of digital devices to preserve data related to the incident. Consequently, there is a need for software systems to be *forensic-ready*, i.e. capable of identifying, collecting, and preserving, *in advance*, data that may be used as evidence to investigate potential, known incidents — should they occur (Alrajeh et al. (2017)).

Pasquale et al. (2018, 2013) have argued for a software engineering approach to support forensic readiness by eliciting requirements for the *proactive* collection of data that may constitute evidence of a potential, known incident. For example, ensuring the *minimality* and *relevance* of the proactively collected data represents two of the main goals of Forensic Readiness (FR). The two goals imply a tradeoff in the engineering of forensic-ready systems, as minimality, on the one hand, aims to minimise the amount of preserved (potential) evidence in order to account for performance issues and reduce the cognitive load on investigators. On the other hand, the preserved evidence must be relevant; i.e. able to support or refute potential incident hypotheses (Alrajeh et al. (2017); Pasquale et al. (2018)).

However, work on supporting FR is limited. The majority of work in this area proposes high-level, organisational guidelines and frameworks that aim to facilitate potential forensic investigations and ensure the integrity of potential evidence. For example, Rowlingson et al. (2004) proposed a list of activities that organisations can engage in to support FR; such as defining relevant business scenarios, identifying potential sources of evidence and increasing the awareness of staff using training programmes. Alenezi et al. (2017) proposed a framework that combines technical, organisational, and legal factors to support cloud-based FR; while Elyas et al. (2015) employed focus groups of experts to discuss challenges faced by practitioners to achieve FR.

A few exceptions to this line of work on FR exist. For example, Shields et al. (2011) proposed a continuous logging system of all potential evidence. However, it is not clear how this proposal scales with the size of the environment as the logging of all potential evidence may not be practical and may severely complicate later analysis and investigations. A more selective approach was proposed by Pasquale et al. (2016) where evidence collection activities are configured to target potential attack scenarios. Alrajeh et al. (2017) proposed a formal and explicit representation of proactive evidence preservation requirements, and designed an approach to synthesise specifications for systems to meet these requirements. The approach was designed specifically to ensure the relevance and minimality of collected evidence in the context of potential incident hypotheses.

2.3.2 Industrial Control Systems

The differences between traditional IT/Software systems and CPS/ICS (Table 2.1) have had implications on how DF are tackled in an ICS or a CPS in general. Al-rimawi et al. (2017) have highlighted a fundamental challenge in investigating general CPS: the interplay between the cyber and physical aspects of a CPS leads to novel attack surfaces that have not been explored before in the practice of DF. In the authors' example, an offender takes advantage of the topology of a smart building and the connectivity of different devices to launch an attack on the Heating Ventilation and Air Conditioning (HVAC) system, causing the server room to overheat and damaging the stored data. This extended attack surface in a CPS/ICS requires digital investigations to be aware of potential cyber-physical interactions. Existing software tools for digital forensics do not support the collection of evidence related to attacks that exploit these interactions.

For ICS in particular, several other practical challenges to DF in these systems are encountered (Awad et al. (2018); Eden et al. (2016); Mohamed et al. (2020); van der Knijff (2014); van Vliet et al. (2015)):

- The safety criticality of industrial processes and the difficulty of shutting them down for investigations limits the ability of forensic tools to interfere in their operation;
- The proprietary nature of most ICS devices and the resulting lack of documentation can complicate data acquisition tasks and may lead to a loss of data;
- Low-level control devices, i.e. those that exist in the vicinity of physical processes (levels 0-1 in Figure 2.2), typically feature low computational and storage resources, which limits the applicability of software forensic tools.

In a large-scale ICS, shutting down a certain safety-critical physical process is often expensive, complex, and time-consuming. It is an operation that requires a significant amount of planning and is usually performed at a frequency in the scale of several years, as illustrated by Wang et al. (2014) with the case study of an ethylene chemical plant. As existing forensic tools may not preserve the state of different hardware and software as they are running (Taveras (2013)), there has been some attention given to tools that can perform *live forensics* in an ICS, i.e. while the system is running (Chan et al. (2010)). In an ICS however, an important challenge facing live forensics is the preservation of the real-time performance requirements essential for the safe operation of physical processes (Ahmed et al. (2012); Taveras (2013)). Janicke et al. (2015) proposed a real-time forensic monitoring framework that provides assurances of behavioural ICS requirements expressed in interval temporal logic, while Al-Sharif et al. (2018) developed a technique to collect digital evidence from the RAM of a computer in an ICS network possibly targeted by an attack. This work however focuses on devices that exist at a high-level in an ICS network (Figure 2.2). Hence, the proposed approach may not directly interfere in the safety-critical operation of devices present at a lower level in the network.

The challenges to ICS forensics caused by the proprietary nature of low-level control devices and their low computational and storage resources are highlighted in a case study performed by van Vliet et al. (2015). The study describes the process of extracting data from a PLC that was damaged following a fire in a wind turbine in the Netherlands in October 2013. With a lack of specialised forensic tools, the data extraction process required a significant effort and close cooperation between the investigative body and the manufacturer to ensure that potential evidence is not lost. Another major difficulty in performing this data acquisition lied in the fact that powering the device would overwrite existing log entries as the PLC is configured to clear space due to low storage capacity. More recently, there has been a growing body of work on developing specialised tools to analyse and acquire evidence from specific ICS devices. For example, Chan et al. (2018); Qasim et al. (2020); Yau et al. (2018) have proposed different logging mechanisms to extract memory variables and configurations of PLC's; while Hadžiosmanović et al. (2012); Myers et al. (2017) have focused on acquiring and analysing logs from engineering workstations.

Other work in ICS forensics has proposed frameworks inspired by work on traditional IT/Software systems forensics (Altschaffel et al. (2019); Elhoseny et al. (2017); Kilpatrick et al. (2006); Mishra (2019); Wu et al. (2013)). For example, Mishra (2019) has proposed an investigation framework that relies on models of the epidemics of malware propagation in an ICS network, while Elhoseny et al. (2017) and Kilpatrick et al. (2006) proposed forensic investigation frameworks based on a multi-agent distributed architecture to acquire data from different parts of an ICS.

Most of the aforementioned existing work on ICS forensics has focused on forensic investigations undertaken after an incident has occurred. Exceptions to this line of work, i.e. live forensics methods, rely on the detection of some anomalous behaviour to trigger data collection activities, while architectures and framework for digital forensics in an ICS are designed for post-mortem investigations. Work on supporting FR in ICS is limited to design-time considerations and generic high-level guidelines. For example, van der Knijff (2014) and Eden et al. (2016) provided a list of forensically relevant sources of data in an ICS and proposed some guidelines to plan incident response measures. Grispos et al. (2017) and Ab Rahman et al. (2016) proposed general design considerations for medical and cloud-based CPS, respectively, in order to incorporate FR requirements. Alrimawi et al. (2019) proposed a technique to extract patterns of incidents that have previously occurred in a given CPS and instantiate these patterns to other CPS in order to understand how the same incident may reoccur. While this technique is promising in terms of supporting the FR of CPS facing reoccurring threats, the work does not provide details on how to support the proactive collection of data that may be relevant to a certain incident.

With attacks that target the physical process in an ICS, the resulting physical damage will lead to a loss of potential evidence stored on low-level control devices. This problem was also highlighted in the case study by van Vliet et al. (2015) where the authors that noted that all electronic devices in the vicinity of the turbine that caught fire were damaged beyond the possibility of extracting data from them. Most existing work on ICS forensics relies on the occurrence of a certain incident in order to trigger data collection or investigation activities (Eden et al. (2016); Mohamed et al. (2020); van der Knijff (2014)). In the case of work on live forensics (Al-Sharif et al. (2018); Taveras (2013)), data collection and investigation activities are assumed to be triggered when an anomaly is detected. However, the reliability of such a trigger depends highly on the ability to detect anomalies resulting from attacks. For example, a class of attacks targeting the physical process, called *Stealthy Attacks*, can evade existing anomaly detectors and still cause physical damage (Bai et al. (2015); Pasqualetti et al. (2013)).

Existing anomaly detectors designed for attacks targeting physical processes may not be able to raise an alarm about such attacks within a sufficient period of time, i.e. before they are able to cause physical damage and subsequently the loss of important data (Giraldo et al. (2017)). Therefore, proactive collection of data in ICS cannot rely solely on alerts from existing Intrusion/Anomaly Detection Systems (IDS/ADS), and must take into consideration the physical effects of attacks. To the best of our knowledge, no work has yet considered these factors in the design of techniques to engineer FR in ICS. As a result, we formulate the following research question:

RQ1. *When should we trigger proactive data collection activities in ICS given that anomaly detectors may not raise an alarm about a stealthy attack in a sufficient period of time (i.e. well before the attack is able to cause damage)?*

Another aspect to consider in proactive data collection is the large amount of data typically generated by an ICS (Kiss et al. (2014)). Collecting all of this data at all times may not be feasible, and may render forensic investigations intractable. In addition, this "agnostic" collection of data may negatively affect the performance of controllers, which is essential to guarantee safe operation and maintenance of real-time requirements. Thus, to support the FR of a given ICS, it is important to identify data that is *relevant* to a forensic investigation, and to systematically restrict data collection to a subset of this data to reduce potential control performance overhead (Pasquale et al. (2018)). To the best of our knowledge, there is no work that considers these aspects for the engineering of forensic-ready ICS. Consequently, we formulate the following two research questions:

RQ2. *How can we identify data that is relevant to a potential investigation of a stealthy attack in ICS?*

RQ3. *How can we select a subset of relevant data such that the control performance overhead resulting from real-time data collection is reduced?*

2.4 CPS and ICS Security

In the previous section, we reviewed work on CPS/ICS forensics, and we highlighted a lack of methods to support the FR of ICS. In particular, we discussed the volatility of evidence in ICS as a result of the ability of attacks to cause physical damage in these systems. An important challenge to the FR of ICS is the fact that *Stealthy Attacks* can evade anomaly detectors and cause damage without getting detected. As the vast majority of live forensics methods in ICS rely on the detection of an anomaly to trigger data collection, there is a need for an alternative trigger given the threat of stealthy attacks. In this section, we provide some background on the security issues of CPS/ICS, particularly attacks targeting physical processes, and we explain the difficulty of detecting stealthy attacks in particular (Figure 2.4). We then review existing work on providing *Early Warnings* about cyberattacks, a topic that is primarily tackled within the traditional IT literature, with scarce applications in CPS/ICS security.

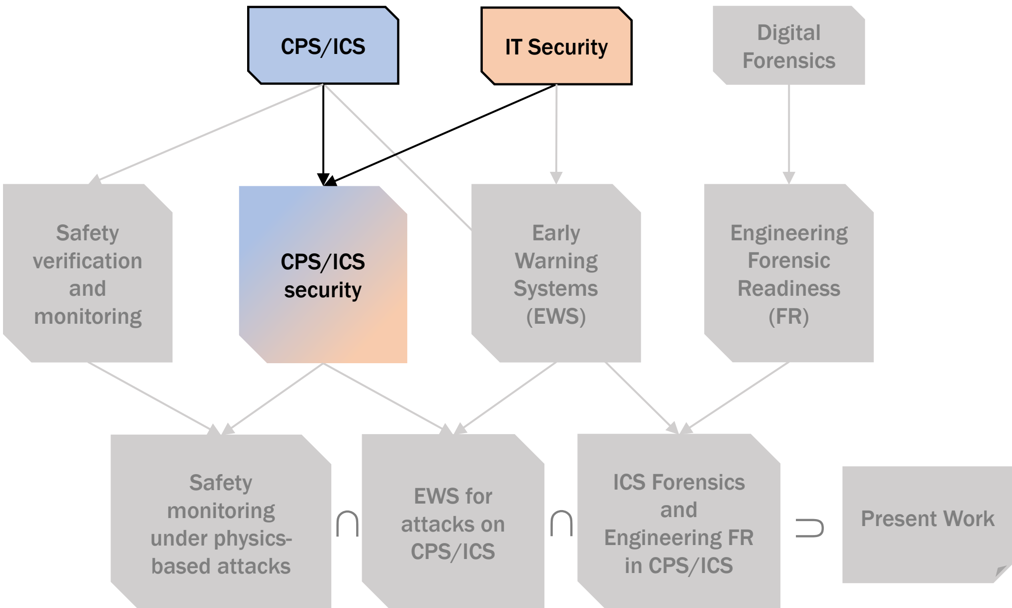


Figure 2.4: Research areas reviewed in Sections 2.4.1 and 2.4.2.

2.4.1 IT-based Approaches

In the past decade, the opening of the traditionally isolated control network to the enterprise network (Level 4, Figure 2.2) has exposed several vulnerabilities within ICS. As a result, there has been a significant research effort to secure ICS. Typically, control networks exhibit a simpler behaviour than the one found in traditional IT systems, with a mostly fixed topology, relatively regular communication patterns, rare changes in users, and a reduced number of different protocols (Cárdenas et al. (2008)). As a result, the implementation of network-based Intrusion Detection Systems (IDS') tends to be easier than in traditional IT systems (Cheung et al. (2007)).

Several network-based IDS inspired from techniques found in traditional IT systems have been proposed. Mitchell and Chen (2014) have grouped these techniques into two main categories: (1) *knowledge-based* and (2) *behaviour-based* intrusion detection. Knowledge-based techniques rely on a known threat model and look for features in the network traffic that are consistent with the predefined model. Examples include the works by Genge et al. (2014) and Cheminod et al. (2017). Behaviour-based intrusion detection relies in contrast on a model of the network itself, and attempts to capture deviations from this model due to potential intrusions. Models of ICS networks can be constructed using formal specifications, such as in the work of Mercaldo et al. (2019), Bernieri et al. (2019), McEvoy and Wolthusen (2010) — among others. Learning-based techniques have also been trained on ICS network datasets and used as models of normal network operations (Liu et al. (2018); Niu et al. (2019); Sayegh et al. (2014)). Other IT-based approaches to ICS security have considered network design for security (Genge and Siaterlis (2014); McEvoy and Wolthusen (2011*a*)); for example, McEvoy and Wolthusen (2011*a*) have suggested an efficient network protocol that seeks to verify the authenticity of network packets circulating in an ICS network.

IDS techniques relying purely on network behaviour are however insufficient to properly detect attacks and secure ICS networks. Hadžiosmanović et al. (2012) has shown that both categories of network-based IDS face several limitations in ICS. First, knowledge-based techniques are ineffective against attacks that exploit zero-day vulnerabilities, for which no known threat model is available, with Stuxnet (Albright et al. (2011)) being the most relevant example of this limitation. Second, the authors have empirically demonstrated that behaviour-based techniques that lack knowledge of the specific semantics of the process are ineffective against attacks that cause little variability in network behaviour. Attackers may use valid commands within normal communication channels to disrupt the operation of an ICS. For example, a PLC can be made to forcefully close a valve mounted on a pipe transporting hot steam, thus potentially causing the pipe to burst. Normal network behaviour could be maintained during this attack, thus evading anomaly-based IDS, and without any knowledge of the physical behaviour of the system (i.e. the semantics), attackers would still manage to cause damage.

2.4.2 Physics-based Approaches

In light of the limitations of traditional IT-based techniques to secure ICS, physics-based approaches, drawing particularly on results within the control theory community, have received increased attention in the past few years (Conti et al. (2021); Giraldo et al. (2018); Lun et al. (2019); Mahmoud et al. (2019); Rashidinejad et al. (2019); Sánchez et al. (2019)). We begin this section with a brief background on attacks that target physical processes in a CPS/ICS before reviewing physics-based approaches to such attacks.

2.4.2.1 Background

The works reviewed in this section consider mainly data obtained from low-level control devices and the physical process itself (Levels 0-1 in Figure 2.2). At this level, control systems feature a feedback loop as shown in Figure 2.5-a. A feedback control system consists mainly of four components: (1) the physical process itself; (2) sensors that report some physical quantities of interest, often grouped in a vector $y(k)$ with k as the sampling time instant; (3) a controller that regulates the process based on some control law; and (4) actuators that influence the process upon receiving a signal $u(k)$ from the controller.

The attacks considered within works adopting a physics-based approach to ICS security consist of disrupting the flow of data between the devices that form the control loop, and are often considered to be Man-in-the-Middle (MitM) attacks. When discussing such attacks, a fundamental assumption is often made that the offenders have managed to gain access to communication channels and

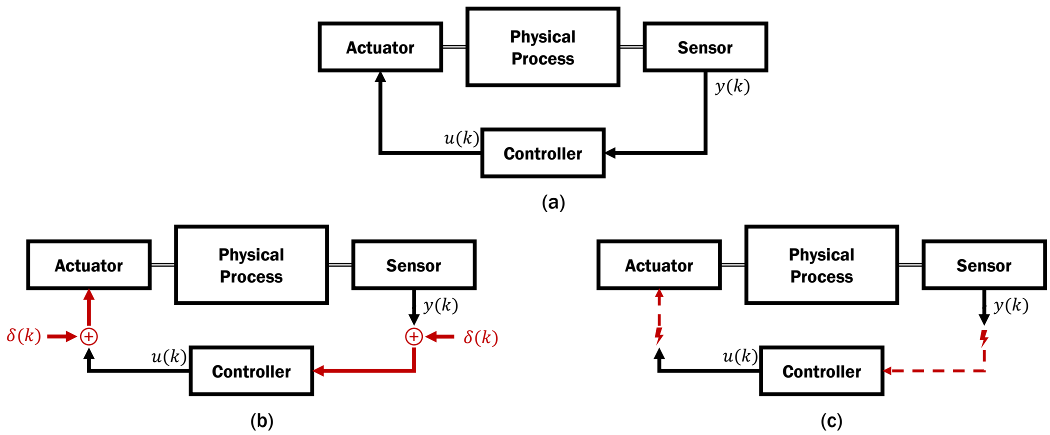


Figure 2.5: (a) General feedback control architecture, with $y(k)$ denoting sensor measurements and $u(k)$ denoting actuator signals (control inputs) at a time k, respectively. (b) General *deception* attack consisting of modifying sensor measurements and/or control inputs with a signal $\delta(k)$. (c) General *DoS* attack consisting of preventing sensor measurements and/or control inputs from respectively reaching the controller and the physical process.

control devices and have bypassed existing cyber firewalls or network-based IDS. Access to control devices can also be assumed to have taken via physical means, such as an attacker physically accessing the relevant devices and modifying certain parameters. In both cases, the study of attacks on ICS within physics-based approaches is often focused on the manner by which control data is modified/intercepted, and is decoupled from the manner by which such devices or communication channels have been accessed. Thus, "attacks" in the context of physics-based methods in ICS often refer to the disruption of the flow of data between control devices.

Attacks inside the control loop can fall within two main categories (Mousavinejad et al. (2019)): (1) *Deception* attacks that compromise its integrity (Figure 2.5-b); and (2) *Denial-of-Service (DoS)* attacks which compromise the availability of the control data (Figure 2.5-c). DoS attacks are similar to the ones found within traditional IT systems, and consist of either preventing sensor measurements from reaching the controller or preventing the controller from sending control inputs. This could easily disrupt the stability of the system as real-time response to potential disturbances is severely hindered (Teixeira, Shames, Sandberg and Johansson (2015)).

Deception attacks consist mainly of corrupting control data with false measurements that don't reflect the true value of the physical quantities of interest, or false control signals that don't reflect the controller's control law. Deception attacks can be further broken down into two main types depending on the kind of malicious modification performed to the signals exchanged between control devices. (i) *Replay* attacks are ones where the attacker records sensor measurements for a certain period of time, then replays these measurements while they perform some other illicit actions to the system (Teixeira, Shames, Sandberg and Johansson (2015)). This is similar to when a security camera in a building is made to replay a video of normal activity while offenders execute their own illicit actions (Mo and Sinopoli (2009)). (ii) *False Data Injection Attacks (FDIA's)* consist of directly modifying sensor measurements or control signals in real-time. A MitM offender could for example add a bias to certain measurements received by the controller so that the latter is tricked into executing an inadequate control response.

2.4.2.2 Attack Detection

Physics-based attack detection in ICS refers to a variety of detection methods where knowledge of the physical dynamics of the system is at the centre of the detection procedure, and indicators of attacks and/or anomalies are based on real-time measurements of the physical process (Giraldo et al. (2018)). The fundamental idea behind this approach, which is derived from the traditional Fault Detection and Identification (FDI) field (Hwang et al. (2009)), is that physical processes in ICS behave according to immutable natural laws, and any deviation from this behaviour could indicate the presence of an anomaly or an attack (Giraldo et al. (2018); Urbina et al. (2016)).

Several works in the past decade have considered this approach to attack detection in CPS/ICS. These works only consider the actions of the attacker at the control loop (Figure 2.5), with an underlying assumption that they have already managed to gain access to different communication channels between low-level control devices, and have been able to evade existing firewalls and/or network-based IDS; or even have physical access to the victim components. The vast majority of works falling into this line of research consider deception attacks as a main concern. DoS attacks are easier to detect at the physical level, as the anomaly detector would immediately sense the absence of sensor measurements at a given time instant, and digital controllers are inherently designed to use the last received sensor measurement in the absence of further real-time inputs. The non-trivial part of securing control systems against DoS attacks lies in either making controllers more resilient to the absence of sensor measurements, or in distinguishing between an actual DoS attack and a fault in the network (Teixeira, Shames, Sandberg and Johansson (2015)). The first aspect is a concern for the Networked Control Systems (NCS) research field, while the second is mostly a concern of network-based IDS. We therefore focus in this section on works dealing with the detection of deception attacks.

The most popular category of physics-based detection methods employs a statistical change detection technique to keep track of significant deviations from normal operations. This class of detectors consists of a *residual* metric that computes the difference between measurements received by the controller and measurements estimated based on a model of the system. A statistical change detection test is then applied to check whether the residual conforms to a nominal statistical distribution, where this nominal distribution is dependent on the noise inherent to the system. In particular, the *Chi-squared* and *Cumulative-Sum* (CUSUM) tests (Mo et al. (2014); Murguia and Ruths (2016)) for statistical change detection have been extensively used and studied in this area of the literature. Other such tests include the Sequential Probability Ratio Test (SPRT) (Cárdenas et al. (2011)), the Exponential Weighted Moving Average (EWMA) (Umsonst and Sandberg (2018)), the Kullback-Leibler divergence (Bai et al. (2015)), and the Wassertein metric (Li and Martínez (2020)).

One of the earliest works to apply these ideas to attack detection is found in the papers by Amin et al. (2012*a*,*b*) which constructed a detailed physical model of a SCADA water distribution system and designed an attack detection strategy based on an ensemble of observers that compute residuals for different sensors in the system. Palleti et al. (2018) and Qadeer et al. (2017) used a CUSUM-based anomaly detector to detect a variety of attacks on a water treatment system. A large body of work considered this approach to attack detection in power systems (Lakshminarayana et al. (2019); Pan et al. (2019); Sargolzaei et al. (2019); Urbina et al. (2016); Zhu and Başar (2011)).

The main limitation behind this class of detectors is the possibility for a resourceful attacker, i.e. an offender with sufficient knowledge of system parameters, to bypass these detectors by exploiting noise and/or control theoretical properties. The term *Stealthy Attack* encompasses a variety of attacks in the feedback control loop that can achieve their goal, which is typically assumed to be the damaging of the system, without getting detected by the existing physics-based anomaly detector.

A large body of work over the past decade has demonstrated both empirically and theoretically that stealthy attacks can be designed to evade existing anomaly detectors. Cárdenas et al. (2011) constructed models of false data injection attacks that can bypass CUSUM anomaly detectors with knowledge of these detectors' tuning parameters by making sure that the false data injected by the attacker does not cause the residual to exceed the detection threshold. The attacker was assumed to possess knowledge of the detection threshold of the CUSUM test. While in general, detection thresholds can be made arbitrarily small to limit the impact of a stealthy attack, this would come at the cost of a high false alarm rate. This relationship was investigated further by Urbina et al. (2016) where an adversary model capable of adapting to changing detection thresholds to remain stealthy was studied.

Several works in the literature have also leveraged results from control theory to prove the possibility of more sophisticated attacks that can bypass existing anomaly detectors (Ayas and Djouadi (2016); Li and Yang (2019); Liu et al. (2011); Park et al. (2019); Smith (2015); Teixeira, Sandberg and Johansson (2015)). Pasqualetti et al. (2015, 2013) provided theoretical proofs that the existence of invariant zeroes in a given control system guarantees the existence of undetectable FDIA on sensors and actuators, regardless of the existing residual-based anomaly detector. Bai et al. (2015) provided theoretical guarantees on the stealthiness of an attacker seeking to deteriorate state estimation in a Kalman filter with a stochastic control system. Murguia and Ruths (2018); Murguia et al. (2018, 2017) have described attack models where stealthiness is achieved by mimicking or only slightly changing the false alarm rate of the existing residual-based anomaly detector.

In light of the limitations of residual-based anomaly detectors, some work has considered *active monitoring* as an effective way of detecting sophisticated stealthy attacks. Active monitoring refers to a class of attack detection methods that rely on modifying the inputs to the system in such a way that the modification would have some known effect on the system's outputs. These effects are then searched for within received sensor measurements where their absence may the suggest the presence of some kind of anomaly. Keller and Sauter (2013) has suggested inducing purposeful communication data losses to reveal potential stealthy MitM attacks. Marquis et al. (2018); Mo et al. (2014, 2015); Satchidanandan and Kumar (2017); Weerakkody et al. (2014) have proposed watermarking of sensor outputs using a known component of the input (actuator) signal. Moving target defence (Fang et al. (2019); Griffioen, Weerakkody and Sinopoli (2019); Tian et al. (2019); Weerakkody et al. (2017); Weerakkody and Sinopoli (2015)), which introduces time-varying parameters that modify the mathematical model of the system, has also been suggested as a way to detect stealthy attacks in the feedback control loop as it hinders the attacker's knowledge of the system. Teixeira et al. (2012) and Hoehn and Zhang (2016) proposed the modification of the structure of the system itself, by introducing additional measurements or modulating input signals in order to remove the possibility of the undetectable attacks described by Pasqualetti et al. (2015, 2013).

The main limitation with active monitoring techniques lies in their potential to compromise the safety and the performance of a control system in practice (Griffioen, Weerakkody, Sinopoli, Ozel and Mo (2019)). Unlike traditional IT systems, ICS operate under hard real-time requirements to ensure safety of operators and equipment. Introducing communication losses in the control loop may have a negative effect on the satisfaction of these requirements. Modifying feedback control inputs may compromise the performance of the system in terms of remaining at a given set-point and may introduce economic losses. While some theoretical guarantees on the performance degradation in the presence of watermarking and signal modulation were presented in the respective works, it is not clear whether these guarantees would hold in practice. Finally, modifications of a control system's structure could be very expensive if at all feasible in realistic situations.

Within the majority of work on attack detection in CPS/ICS, the issue of detection delay is rarely addressed (Bonakdarpour et al. (2018); Giraldo et al. (2018)). This delay refers specifically to the time elapsed between the start of a given attack and the time instant an alarm is raised. Detection delay is especially important in safety-critical systems where a sufficient period of time is required to implement defensive measures that can prevent potential damage from taking place. With physics-based detectors, there is a trade-off between detection delay and false alarm rate (Giraldo et al. (2018); Urbina et al. (2016)). However, to the best of our knowledge, work addressing this issue is scarce. Akowuah and Kong (2021) have proposed an adaptive approach whereby the parameters of a CUSUM detector are tuned in real-time to either favour a low false alarm rate or a low detection delay depending on the current state of the system. While the approach has been shown to perform well with simple deception attacks (fixed biases on sensor values), it is not clear how it performs under stealthy attacks where knowledge of the existing anomaly detector is used to evade it. The detection delay is also considered as a performance metric in the work by Park et al. (2017). However, the authors' results show that under a knowledgeable stealthy attack, the detection delay can be too high to guarantee timely detection, i.e. well before damage can happen.

2.4.2.3 Other Physics-based Approaches

Ideas from the domains of networked, robust, fault-tolerant, and optimal control have also been applied to increase the resilience of control systems when faced with attacks in the control loop. Several works have considered resilience in the presence of DoS attacks (Amin et al. (2009); Gupta et al. (2010); Hammad et al. (2015); Li et al. (2019); Mousavinejad et al. (2019); Sun and Yang (2019)) and adapted results in networked control systems which deals with the design of control systems under various communication limitations. Other work has considered resilience under deception attacks (Chen and Sankaranarayanan (2017); Fawzi et al. (2014); Zhu and Martinez (2014); Zhu and Başar (2011)). However, the majority of these works propose resilient control strategies that are dependent on some technique to first detect attacks, similar to those reviewed in the previous section. Thus several of these works face the same previously described limitations under stealthy attacks.

The use of statistical modelling and machine learning in order to model the physical dynamics of a given system and perform anomaly detection can be also observed in the literature (Amin et al. (2019); Junejo and Goh (2016); Kravchik and Shabtai (2019); Mashima and Cárdenas (2012); Sargolzaei et al. (2019)). While these techniques may reduce or even remove the effort required to model the system using first principles, and may also be more effective against "low-and-slow" stealthy attacks, they also face several drawbacks. The first of which lies in their need for proper datasets for training and testing. The majority of public datasets in cybersecurity research include data concerning network behaviour and are geared towards traditional IT-based approaches to security. In addition, building datasets of anomalous behaviour by gathering data from real-world systems may be infeasible in practice, as simulating attacks that can cause physical damage can be costly and hazardous. Therefore, works like the one by Aoudi et al. (2018) focus on constructing datasets based on normal behaviour and training machine learning models to detect deviations from this behaviour. However, the work by Erba and Tippenhauer (2020) has shown empirically that several of the physics-agnostics techniques proposed in the literature can miss sophisticated attacks that take advantage of some dynamical properties of the system as these properties are not normally captured by the statistical modelling technique during the training phase.

Finally, we mention some works on offline risk assessment based on the quantification of the potential impact of stealthy attacks using physical models. Esfahani et al. (2010); Hashemi et al. (2018); Liu et al. (2021); Milošević et al. (2018); Murguia et al. (2017) have proposed reachability analysis techniques to quantify the impact of potential attacks using the size of the reachable set, which can then be used as a component of some risk measure. Other works such as the ones by Hendrickx et al. (2014); Milosevic et al. (2018); Sandberg et al. (2010); Teixeira et al. (2013); Teixeira, Sou, Sandberg and Johansson (2015) propose a so-called *security index* which characterises the minimum number of components where the allocation of security measures such as encryption would be needed in order to prevent stealthy attacks. Existing risk management-based approaches are mostly focused on offline risk assessment and do not consider transient operation of some realistic ICS, which may then introduce variations to the initial system state before an attack occurs, and subsequently to the potential impact of the attack. Techniques to allocate security measures using the security index have not yet considered realistic considerations of real-time requirements when implementing measures such as encryption. For example, real-time requirements may dictate that the number of sensors that can tolerate latency due to security measures is less than the minimum required to prevent stealthy attacks. In such case, an expensive redesign of the control architecture may be needed to achieve the desired results.

2.4.3 Early Warning Systems (EWS)

In the previous sections, we introduced the area of CPS/ICS security and provided some background on security issues facing safety-critical ICS. We also discussed work on the detection of stealthy attacks targeting physical processes, and we focused on the difficulty of detecting such attacks and the limitations of existing techniques. In particular, we observed that most work in this area focuses on performing attack detection up to a certain performance limit, but rarely addresses the issue of detecting or warning against the attack well before harm takes place. In this section, we discuss work on *Early Warning Systems* (EWS). EWS can be considered as a proactive approach to deal with potential harm in a variety of scenarios, and their design has the aim of supporting the selection of proactive measures. In the context of security, EWS have found applications primarily within traditional IT systems. We begin this section with some background on these systems and the features that differentiate them from traditional attack detection approaches, before providing an overview of work in this area, including the limited work applied to CPS/ICS security (Figure 2.6).

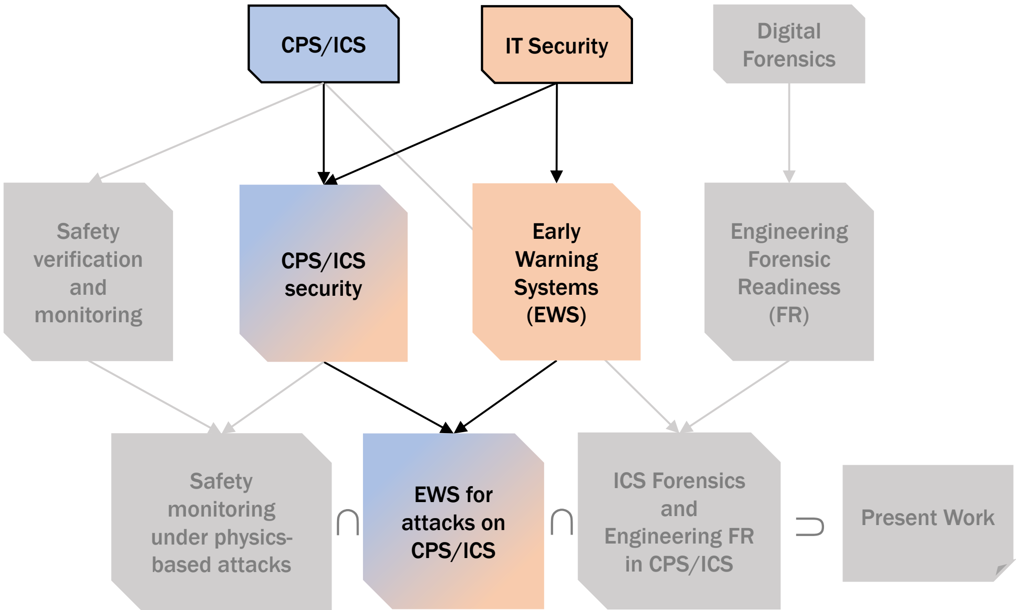


Figure 2.6: Research areas reviewed in Section 2.4.3.

2.4.3.1 EWS and Traditional Attack Detection

EWS are well established in the context of disasters that threaten infrastructure and individuals. For example, early warnings about natural disasters can help public authorities take measures in order to reduce material and human losses; such as reducing water levels in reservoirs, reinforcing structures, and increasing the capacity of healthcare services (Zhou et al. (2018)). EWS are also studied in different domains of healthcare; for example, an early warning score based on different physiological parameters is considered by Subbe et al. (2001) to prevent catastrophic deterioration in intensive care unit patients. Regardless of the application domain, an EWS often features two main characteristics: (1) a tool to process data from various sources and to predict — up to a degree of uncertainty — potential harm; and (2) a mechanism to guide the selection of measures that mitigate harm.

In recent years, with the increased complexity of IT systems and cyber-threats, the study of EWS against cyberattacks has received some attention. In the context of IT security, EWS combine preventive measures, such as risk and vulnerability assessment, with IDS, to provide a clearer "picture" of the security situation and send warnings about potential network intrusions (Apel et al. (2009)). They are regarded as a complementary solution to existing IDS and Anomaly Detection Systems (ADS), where their main benefit lies in providing predictions of potential harm in unfamiliar situations, typically with zero-day attacks (Ramaki and Atani (2016)).

Traditionally, IDS/ADS exist as a reactive security solution where alarms are generally based on either a clear deviation from normality or a strong evidence of misuse. However, intelligently crafted attacks that take advantage of unknown (zero-day) vulnerabilities may be able to successfully achieve their goals without getting detected. Dealing with the threat of zero-day attacks requires a more proactive approach. Differently from IDS/ADS, EWS proactively collect, accumulate, and combine events that don't necessarily form part of a known attack signature or a clear anomaly in order to construct a better picture of the security situation (Ramaki and Atani (2016)). This serves to inform operators about potential threats to their network in their early stages, and advise on necessary measures that prevent the violation of security properties. Thus, in contrast to an alert raised by an IDS or an ADS, an *"early warning"* does not necessarily imply that an attack is present, it is rather an uncertain prediction of potential future harm based on a variety of indicators.

The collection and aggregation of data (often considered to be of "low quality"(Kalutarage, Lee, Shaikh and Sung (2015))) from heterogeneous sources is a prominent feature of EWS proposed in the literature. For example, Chivers et al. (2009) considered events such as failed connections, failed logins, and anomalous phone calls to be weak indicators of malicious insider activity. Such events by themselves may not be considered by an IDS/ADS. These indicators are then aggregated and accumulated using a Bayesian score that reflects the probability of a node being subverted. Kalutarage, Lee, Shaikh and Sung (2015); Kalutarage et al. (2013) considered a similar Bayesian approach to tackle "low-and-slow" network attacks in their reconnaissance phase. Grobauer et al. (2006) and Apel et al. (2009, 2010) both proposed EWS that rely on intelligence sharing between different organisations and stakeholders in order to counter advanced coordinated cyber threats in their early stages. The main challenge with this class of EWS is to ensure the privacy and confidentiality requirements when data is moved between different stakeholders.

EWS go beyond IDS/ADS by incorporating a mechanism to predict potential harm to the system. This is performed using, for instance, statistical modelling as in the work of Abbaszadeh et al. (2018) or unsupervised learning as described by Brignoli et al. (2020). The EWS proposed by Apel et al. (2010) allows predictions of cyber threats by automatically deriving new attack signatures and reporting novel malware threats. While EWS by themselves may not necessarily detect attacks, they can guide the selection of actions that may in turn reveal a potential intrusion. For example, the framework proposed by Brignoli et al. (2020) allows the evaluation of the potential impact and subsequently the selection of active countermeasures in IoT networks. The approach proposed by Kalutarage et al. (2012) combines Bayesian reasoning and statistical normality testing for the attribution of malicious or suspicious activity in the context of cyber conflicts. Finally, it is worth mentioning that a variety of commercial EWS exist (Ramaki and Atani (2016)). Examples of these systems include the *Symantec DeepSight* system (Symantec (2014)) that gathers threat intelligence from a large number of sensors deployed globally, and similar country-level EWS such as *Herold* (Theilmann (2010)) deployed in Germany.

2.4.3.2 CPS and ICS Applications

To the best of our knowledge, work on EWS in ICS and specifically in the context of attacks targeting the physical process is scarce and limited. One exception is the work by Abbaszadeh et al. (2018), which generates early warnings based on potential anomalies predicted by learning time series behaviour. Other work which can be considered close to implementing EWS in ICS due to incorporating some form of prediction of unsafe states falls under the line of online safety monitoring (Carcano et al. (2011); Castellanos and Zhou (2019); Coletta (2018); Etigowni et al. (2018)) discussed in Section 2.5.2. For example, the work by Etigowni et al. (2018) incorporates a prediction of future system behaviour using an approximated model of the system and the monitoring of controller states.

However, none of these works consider attacks that take advantage of system and control-theoretic properties, such as measurement noise, to evade detection.

2.4.4 Summary of CPS/ICS Security Review

In Section 2.4, we presented an overview of relevant work in CPS/ICS security. We highlighted the main theoretical and practical limitations of securing ICS under the threat of stealthy attacks. Our survey of this area has revealed that most of the existing work dealing with stealthy attacks has focused on the reactive detection of attacks based on the resulting (physical) anomalies — up to a certain performance bound. This essentially implies that zero-day stealthy attacks that can cause significant damage (before resulting anomalies can be detected) may always be possible. In addition, practical limitations to approaches such as active monitoring and security measure allocation are not considered in the majority of the surveyed works.

Given the fundamental practical and theoretical limitations underpinning the detection of stealthy attacks, there is a need for a more proactive approach that can provide *early warnings* about stealthy attacks, particularly well before any harm can take place. However, in our review of the area of EWS, we noted a lack of techniques developed specifically for attacks on ICS. The limited work in this area assumes that raw data obtained from sensors and actuators can be trusted, and does not consider stealthy attacks. Consequently, we formulate the following research question:
RQ4. *How can we warn about stealthy attacks on ICS well before they are able to cause any harm?*

2.5 Monitoring for Safety in ICS

In the previous section, we provided some background in CPS/ICS security, and reviewed some related work within. We identified a gap in this area consisting of a lack of work on EWS that can warn well in advance of potential stealthy attacks in an ICS. In this section, we briefly introduce the area of safety verification and monitoring of control systems. Our work partly falls within this area as we would

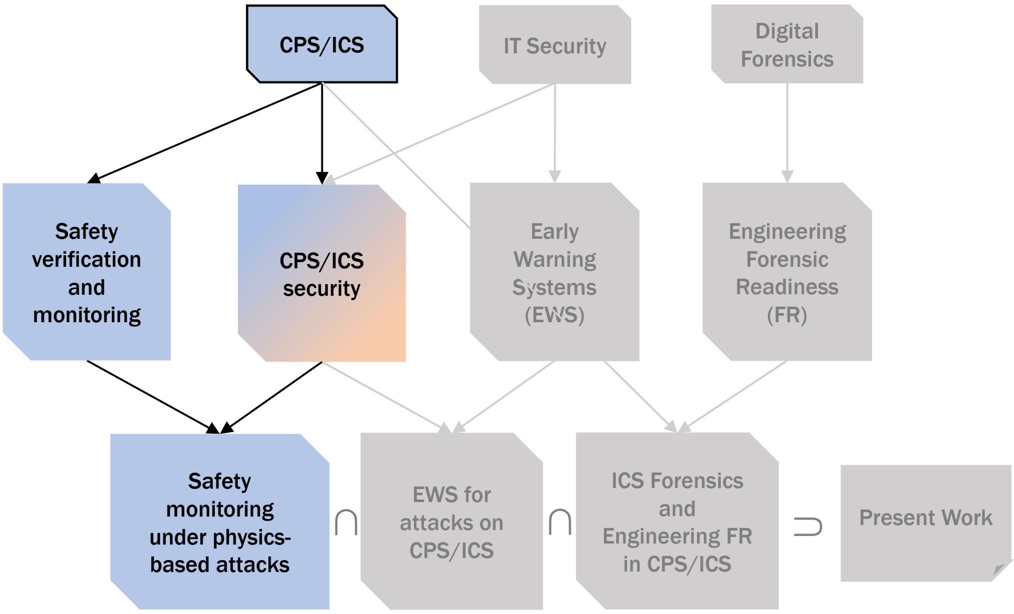


Figure 2.7: Research areas reviewed in Section 2.5.

like to monitor the effect of potential stealthy attacks on the physical process in an ICS, in order to warn against potential safety violations. Consequently, we review the relatively small body of work within the area of safety monitoring under attacks targeting the physical process (Figure 2.7).

2.5.1 Safety Verification and Monitoring

The process of verifying a control system aims to check whether its behaviour satisfies a given property (Alur (2015)). With *safety verification* of control systems, we aim to check in particular whether the system behaves in a way that guarantees the satisfaction of some predefined safety constraints. For a dynamical system, *reachability analysis* can be used to perform safety verification, where the problem is often formalised as asking whether a predefined set of unsafe states is reachable from a given initial state or from a set of initial states, at a given time during the system's execution (Le Guernic (2009)).

Computing the exact reachable set of a dynamical system, except in a limited number of cases, is an undecidable problem. Thus, the vast majority of work in this area focuses on approximating the reachable sets with geometric set representations and attempts to design computationally efficient algorithms to perform these approximations. Examples of such geometric approximations include boxes, zonotopes, ellipsoids — among others. Efficient and scalable algorithms for reachability analysis of different classes of dynamical systems have been extensively studied in the literature, with examples including algorithms employing support functions to guide reachability (Le Guernic and Girard (2009)), or results in ellipsoidal calculus (Kurzhanski and Varaiya (2000)).

The problem of *monitoring* the safety of a control system is similar in its objective to the more traditional verification problem in the sense that they both look for potential violations of safety constraints. However, monitoring is performed in real-time, and in the context of control systems, it specifically asks whether given the *current* state of the system, there exists a control input or a potential disturbance that can take the system outside its predefined set of safe states within a bounded period of time in the future (Chen and Sankaranarayanan (2017); Li et al. (2013)). In practice, the approach is more difficult to implement as real-time requirements impose that such monitoring techniques be fast and efficient. As reachability analysis is generally computationally expensive, performing it in real-time using existing techniques for safety verification may not be feasible.

However, with the advent of black-box control algorithms that are not amenable to the formal reasoning required to derive safety guarantees, the problem of real-time safety monitoring has gained increased attention in recent years. Examples of such black-box control laws include Model Predictive Control (MPC) and machine learning-based algorithms (Chen and Sankaranarayanan (2017)). The problem of real-time safety monitoring has also been motivated with the increased complexity of some systems whereby offline models that are relied upon to perform safety verification may not accurately represent the behaviour of the system at runtime. This is especially true in the area of medical device CPS where the large number of parameters and the complexity of biological systems have motivated the design of online model checking (Li et al. (2013)).

Several works in this area have attempted to adapt existing reachability analysis results in order to make them computationally feasible in real-time (Allen et al. (2014); Althoff et al. (2015); Bu et al. (2019); Chen and Sankaranarayanan (2017); Chou et al. (2020); Johnson et al. (2016); Li et al. (2013); Tran et al. (2019)). Althoff et al. (2015) and Chen and Sankaranarayanan (2017) proposed an approach to online safety monitoring whereby an approximation of reachable sets using symbolic representations is performed offline, and then instantiated online given concrete estimates of the current state of the system. This approach is promising in terms of real-time performance as it restricts the most computationally expensive steps to an offline, design-time activity. Allen et al. (2014) have adopted a similar approach with the difference being that the offline activity consisted of solving multiple two-point boundary value problems which were then used as a training dataset to construct linear regression and support vector machine models. At runtime, these models were used in query-based algorithms for an approximate real-time solution to the reachability problem.

Other works such as the one by Tran et al. (2019) took advantage of reachability tools designed to be efficient for short-time reachability analysis to perform it in real-time. This is feasible since real-time reachability analysis essentially attempts to compute the reachable set from one sampling time instant to the other, which is a short interval of time relative to the entire runtime duration of the system. Aside from using reachability analysis, the prediction of future system states using an identified model of the system was used to perform safe planning; such as in the work of Zhu and Başar (2011) focused on the management of high-speed train operations, and in that of Chou et al. (2020) where Bayesian inference was used to predict potential collisions between Unmanned Aerial Vehicles (UAV's).

2.5.2 Safety Monitoring Under Physics-based Attacks on CPS/ICS

Attacks targeting the physical process in an ICS as shown in Figure 2.5 can be considered to be exogenous inputs to the system. Safety monitoring under such attacks asks whether they can take the system from its current state to one that violates one or more safety constraints. In the more traditional case of safety monitoring or verification without attacks, the problem often assumes that disturbances or control inputs are restricted to some set of admissible inputs. In the case of attacks, especially deception attacks, it may not possible to know in advance the set of potential attacks. However, the assumption that such attacks may attempt to evade the anomaly detector can be used to restrict the space of possible attacks.

As described in Section 2.4.2.3, there exists some work on the safety verification of control systems under stealthy deception attacks (Esfahani et al. (2010); Hashemi et al. (2018); Liu et al. (2021); Milošević et al. (2018); Murguia and Ruths (2018); Murguia et al. (2018, 2017); Zhang et al. (2020)). However, these works use reachability analysis to quantify the potential impact of a stealthy deception attack rather than perform online monitoring. Although this is useful when a risk assessment activity is needed to identify certain vulnerable sensors or actuators; during execution, the system may experience transients where online monitoring is necessary to identify potential initial states of trajectories induced by attacks and leading the system outside its safe operating region.

To the best of our knowledge, a small body of work falls under the category of online safety monitoring under attacks. The work by Kwon and Hwang (2016*a*,*b*, 2018); Kwon et al. (2013), suggests a reachability analysis-based safety monitoring approach that relies on a recursive computation of a matrix structure describing the reachable set. While the algorithm performs well in real-time, it is not clear how it scales with the number of states as the tests focused on UAV's with up to four states describing the geographical position and orientation of the vehicle. In ICS, hundreds of state variables may be involved, which necessitates a deeper investigation into scalability and real-time performance. Furthermore, safety checking was tailored for UAV applications where safe sets are time-varying and can be interpreted as ellipsoids centred on the current state of the system. In ICS, unsafe states are usually time-invariant and consist of fixed bounds on certain operating variables which leads to a different geometric interpretation of safe/unsafe sets.

Another class of approaches for safety monitoring under attacks (Carcano et al. (2011); Castellanos and Zhou (2019); Coletta (2018); Etigowni et al. (2018))

relies on a notion of proximity to a predefined set of unsafe/critical states. This line of work does not consider formal safety guarantees; instead, it uses metrics reflecting the proximity of the system to unsafe states as a way to either determine the level of safety or to detect potential anomalies. For example, Carcano et al. (2011); Coletta (2018), considered the minimum Euclidean distance from current states to the unsafe operating region as a measure of proximity. Castellanos and Zhou (2019) extended this notion further by computing an approximate "time-to-critical-states" metric. However, these approaches rely only on raw sensor values and do not consider the effect of stealthy attacks. For example, intelligently crafted sensor attacks introduce "low-and-slow" modifications to sensor values, which may eventually not reflect the real state of the system. In such scenarios, employing reachability analysis tools becomes essential to identify the space of potentially altered system states. Based on this review of the literature on safety monitoring under attacks, we formulate the following research question:

RQ5. *How can we perform online safety monitoring to predict potential damage from a stealthy attack in large-scale ICS, with an acceptable real-time performance?*

2.6 Summary

Our work lies at the intersection of three domains: Digital Forensics (DF); Cyber-Physical/Industrial Control Systems CPS/ICS; and IT security. Within these three domains, the work is specifically related to the areas of ICS forensics and Forensic Readiness (FR); Early Warning Systems (EWS) for attacks on CPS/ICS; and safety monitoring of CPS/ICS under physics-based attacks. Our review of these areas has identified three main gaps with associated research questions as follows:

- Within the area of ICS Forensics and FR, we identified a lack of work aimed at supporting the FR of ICS, and specifically, the proactive collection of potential evidence under the threat of stealthy attacks that can cause physical damage. Consequently, we formulated the following research questions:

  - **[RQ1]** *When should we trigger proactive data collection activities in ICS given that anomaly detectors may not raise an alarm about a stealthy attack in a sufficient period of time (i.e. well before the attack is able to cause damage)?*
  - **[RQ2]** *How can we identify data that is relevant to a potential investigation of a stealthy attack in ICS?*
  - **[RQ3]** *How can we select a subset of relevant data such that the control performance overhead resulting from real-time data collection is reduced?*

- Within the body of work on the security of CPS/ICS, and specifically within the area of EWS, we identified a lack of approaches that provide timely warnings about a potential attack on an ICS (i.e. well before they can cause damage), and the limited body of existing work does not consider stealthy attacks. This led to the following research question: **[RQ4]** *How can we warn about stealthy attacks on ICS well before they are able to cause any harm?*

- Within the area of safety monitoring of CPS/ICS under attacks, we observed a lack of techniques to perform efficient and scalable online safety monitoring for large-scale ICS faced with the threat of stealthy attacks. This observation resulted in the following research question: **[RQ5]** *How can we perform online safety monitoring to predict potential damage from a stealthy attack in large-scale ICS, with an acceptable real-time performance?*

We attempt to provide answers to these research questions in the remainder of this thesis. In the next chapter, we provide an overview of our approach and motivate it with an example of a stealthy attack on a safety-critical ICS.

Chapter 3

Forensic Readiness of Industrial Control Systems under Stealthy Attacks

In the previous chapter, we discussed related work within the three main fields that include the work in this thesis. In particular, we reviewed work on ICS forensics and Forensic Readiness (FR), CPS/ICS security, Early Warning Systems (EWS), and on the monitoring for safety under attacks that target physical processes.

Our review of related work has identified three main gaps we aim to address, and consisting of a lack of: (1) approaches to support the FR of ICS, specifically in terms of proactive data collection before a potential stealthy attack is able to cause damage; (2) work on EWS that can warn about a stealthy attack affecting the physical process in an ICS well before potential damage happens; and (3) online safety monitoring techniques for ICS that can predict, with an acceptable real-time performance in a large-scale ICS, potential safety violations by a stealthy attack. We then formulated the following research questions:

- Within the first gap:
  - **[RQ1]** *When should we trigger proactive data collection activities in ICS given that anomaly detectors may not raise an alarm about a*

stealthy attack in a sufficient period of time (i.e. well before the attack is able to cause damage)?
  - [**RQ2**] *How can we identify data that is relevant to a potential investigation of a stealthy attack in ICS?*
  - [**RQ3**] *How can we select a subset of relevant data such that the control performance overhead resulting from real-time data collection is reduced?*

- Within the second gap:
  - [**RQ4**] *How can we warn about stealthy attacks on ICS well before they are able to cause any harm?*

- Within the third gap:
  - [**RQ5**] *How can we perform online safety monitoring to predict potential damage from a stealthy attack in large-scale ICS, with an acceptable real-time performance?*

In this thesis, we address these research gaps and research questions by proposing an approach to engineer the FR of ICS presented with the threat of stealthy attacks targeting safety-critical physical processes. We aim specifically to support the proactive collection of data that can constitute potential evidence of a stealthy attack. To achieve this aim, and given the gaps and research questions that we highlighted previously, we expand our main objectives (Section 1.2) as follows:

- [**OBJ1**] *Provide an alternative trigger for proactive data collection.* Achieving this objective helps in answering [**RQ1**]. To provide such a trigger, we need to warn about stealthy attacks in ICS well before they can cause any harm. However, due to literature Gap (2), we need to address the following objective as well:
- [**OBJ2**] *Provide an approach that can warn about safety violations from potential stealthy attacks on ICS well before any damage can happen.* This objective helps in answering [**RQ4**], and is concerned with our framework for physics-based EWS for stealthy attacks on ICS. This framework defines

preliminary indicators that compose a likelihood of damage if a stealthy attack takes place. Measures of these indicators can be computed through online safety monitoring under stealthy attacks. However, due to Gap (3), we need to address the following objective:

- [**OBJ3**] *Design an online safety monitoring algorithm for ICS faced with a stealthy attack that can perform well in real-time for large-scale systems.* This objective addresses [**RQ5**], and is concerned with our instantiation of the physics-based EWS to a widely used class of dynamical systems, Linear Time-Invariant (LTI) systems.
- [**OBJ4**] *Design a technique to identify ICS data that is relevant to potential forensic investigations, i.e. data that is at risk of being lost due to damage by an attack.* This objective is concerned with our automated technique that can identify a subset of data generated by an ICS that may be lost due to physical damage. Achieving this objective provides an answer to [**RQ2**].
- [**OBJ5**] *Design an approach that can select a subset of relevant data to collect, by achieving a trade-off between potential attack impact and the cost of data collection.* This objective is concerned with our framework for the selection of a subset of relevant data to collect by achieving a trade-off between a measure of collection cost and a measure of attack impact. Achieving this objective provides an answer to [**RQ3**].

In this chapter, in addition to presenting our overall approach to support the FR of ICS faced with stealthy attacks, we describe our framework for physics-based EWS in ICS which forms the core of this approach. The design of this framework primarily addresses [**RQ4**] and [**OBJ2**].

We begin the rest of this chapter with a clarification of the different facets of the problem and an overview of the approach proposed in this thesis in Section 3.1. We then illustrate the problem we tackle in this thesis through a motivating example in Section 3.2. We then describe our framework for physics-based EWS in Section 3.3, before concluding this chapter with a brief summary in Section 3.4.

3.1 Problem Statement and Overview

In this section, we clearly state the problems we tackle in this thesis, and provide an overview of our approach aimed at supporting Forensic Readiness (FR) in ICS presented with stealthy attacks.

3.1.1 Problem Statement

In this thesis, we aim to provide an approach to engineer FR in ICS by proactively collecting data that can explain how a stealthy attack occurred. In particular, we aim to collect this potential evidence before any suspected attack is able to cause damage and data loss. In Section 1.1, we highlighted two main facets of the problem: (i) determining a *timely trigger* for data collection (well before an attack can cause damage), and (ii) specifying *which* data to collect. In this section, we provide a more detailed statement of the problem which requires dividing the second part in two: identification of relevant data and the consideration of control performance overhead. This manner of stating the problem seeks to clarify the mapping of its different facets to the components of our approach. As a result, our problem can be restated as follows:

- First (P1), since stealthy attacks can avoid detection, it is not possible to rely on the alarms generated by anomaly detectors to trigger proactive data collection or live forensics as in the work of Taveras (2013). Thus, it is necessary to identify an alternative trigger for proactive data collection.

- Second (P2), not all data generated by an ICS may be required to be collected proactively, since only a subset of such data may be at risk of being lost due to damage. Consequently, there is a need for a technique to identify what data is faced with such a risk.

- Third (P3), ICS generally operate under strict real-time performance requirements and are expected to generate a profit; thus data collection activities can negatively affect the performance of the plant. To reduce the overhead introduced by data collection, this activity should only be performed when the cost of collection does not exceed the potential damage that the attack can cause to the plant.

3.1.2 Overview of the Approach

Our approach to engineer FR in ICS presented with stealthy attacks consists of three main components that respectively address the problems (P1-P3) described previously:

1. [COMP1] — **Physics-based Early Warning System (EWS):** To trigger data collection (P1), we rely on predictive safety checks of whether a potential stealthy attack can indeed take the system into an unsafe state and potentially cause the loss of data. We perform these safety checks in the context of a framework that we propose for physics-based EWS for stealthy attacks on ICS. This framework defines preliminary indicators, or *grounds for suspicion*, of a potential stealthy attack that can cause damage without getting detected by the existing anomaly detector. EWS instantiated according to this framework do not seek to replace existing ADS/IDS, but rather to complement them with (uncertain) predictions of whether a stealthy attack can evade them given the physics underlying the system. Therefore, our EWS does not necessarily reveal attacks, but can be used as a trigger for certain measures, particularly proactive data collection. We describe this framework in Section 3.3, and we instantiate it to Linear Time-Invariant (LTI) systems (a popular modelling framework for physical processes in ICS) in Chapter 4.

2. [COMP2] — **Relevant Data Identifier:** This component identifies what data is at risk of being lost due to a potentially damaging stealthy attack, i.e. relevant data (P2). We employ the safe Process Plant Layout (PPL) (Quiroz-Pérez et al. (2021)), provided by design, to identify areas of the plant that may be affected by the suspected breach. Within these areas, we identify devices and data originating from these devices that can represent potential evidence and that are at risk of being damaged.

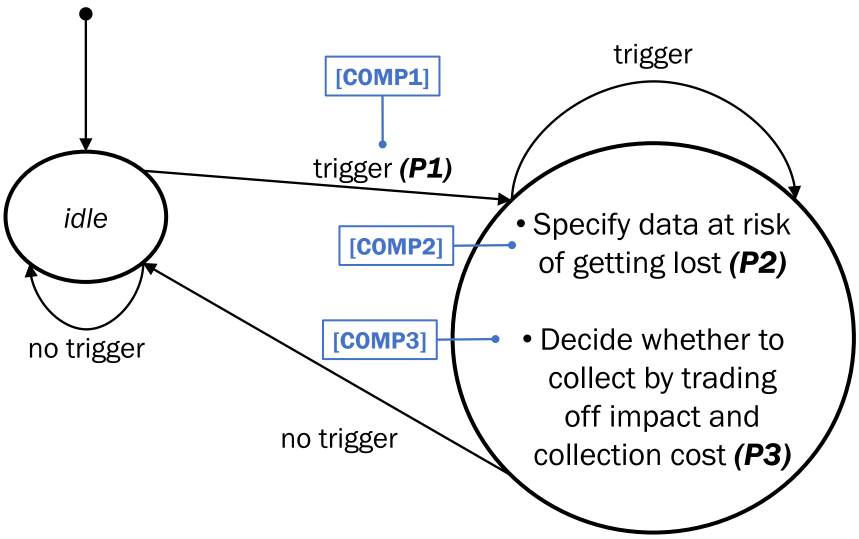


Figure 3.1: Overview of the proposed approach to engineering Forensic Readiness (FR) in Industrial Control Systems (ICS).

3. **[COMP3] — Relevant Data Selector:** Among the set of data that may potentially be lost, this component selects the ones that are worth collecting (P3) by existing data acquisition tools. To achieve this aim, we propose a framework that decides to collect a given piece of relevant data only if the expected impact of the attack, in terms of its potential effect on the system's performance, does not exceed potential data collection cost. As a result, we collect only a subset of relevant data (Pasquale et al. (2018)), resulting in a limited overhead on the system's real-time performance and a trade-off with potential attack impact.

   We describe both **[COMP2]** and **[COMP3]** in Chapter 5.

Figure 3.1 shows the three components of our approach in the form of an automaton reflecting its real-time deployment. Our main assumption in this work is that data acquisition tools, such as the one suggested by Yau et al. (2018), are available to be activated whenever data should be collected from a device or a network section of an ICS. In other words, our relevant data selector (**[COMP3]**) can activate the necessary data acquisition tools whenever it selects which relevant data is worth collecting. In addition, we assume that a secure server is available to store the collected data.

3.2 Motivating Example

We motivate the work in this thesis with an example based on the benchmark Tennessee-Eastman Process (TEP). We begin this section with a description of the TEP setup, before describing the attack scenario.

3.2.1 Tennessee-Eastman Process Setup

The TEP is a benchmark chemical process suggested by Downs and Vogel (1993) and based on a real industrial plant that produces two liquid products from four gaseous reactants. The TEP has been recently used as a virtual testbed for various works in ICS security as it represents a realistic chemical processing environment (Cárdenas et al. (2011); Geng et al. (2019); Genge and Siaterlis (2014); Krotofil and Cárdenas (2013); McEvoy and Wolthusen (2011*b*)). The process consists mainly of five major operating units: a reactor vessel, a product condenser, a vapour-liquid separator, a recycle compressor, and a product stripper. For the purposes of our work, we assume that the system layout is as shown in Figure 3.2. Each main operating unit in the TEP is housed in its own control room, which contains - aside from the process equipment - low-level control devices such as sensors/actuators, Programmable Logic Controllers (PLC's), Remote Terminal Units (RTU's) and engineering workstations. Each control room is connected over a network to a central room that houses supervisory control servers and process historians. In each control room, the relevant controllers' logic is installed on the PLC's which are in turn connected to the engineering workstations via an RTU. The RTU's relay control data to the central control room, as in the case with Supervisory Control and Data Acquisition (SCADA) architectures.

Furthermore, an anomaly detector located in the central control room monitors control data, i.e. sensor measurements and actuator inputs, to detect anomalies due to faults or malicious tampering. In particular, we consider a popular chi-squared anomaly detector (Hashemi et al. (2018); Murguia and Ruths (2019)). At each time step, the anomaly detector uses control inputs, historical sensor measurements, and a model of the system to predict sensor measurements. These predictions are then compared with the received measurements using a metric, called *residual*, that essentially computes the difference between predictions and

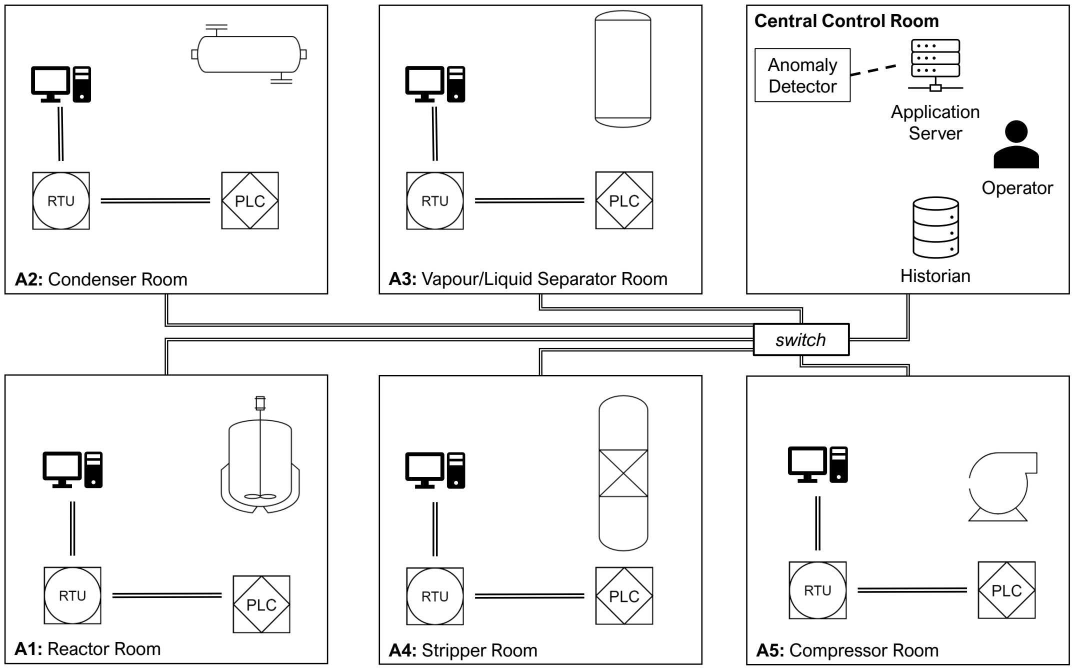


Figure 3.2: Layout of the Tennessee-Eastman Process (TEP).

received measurements. The residual is assumed to follow a certain statistical distribution with a known mean and covariance matrix. Deviations from these distribution are detected using a chi-squared test with a specified threshold.

3.2.2 Attack Scenario

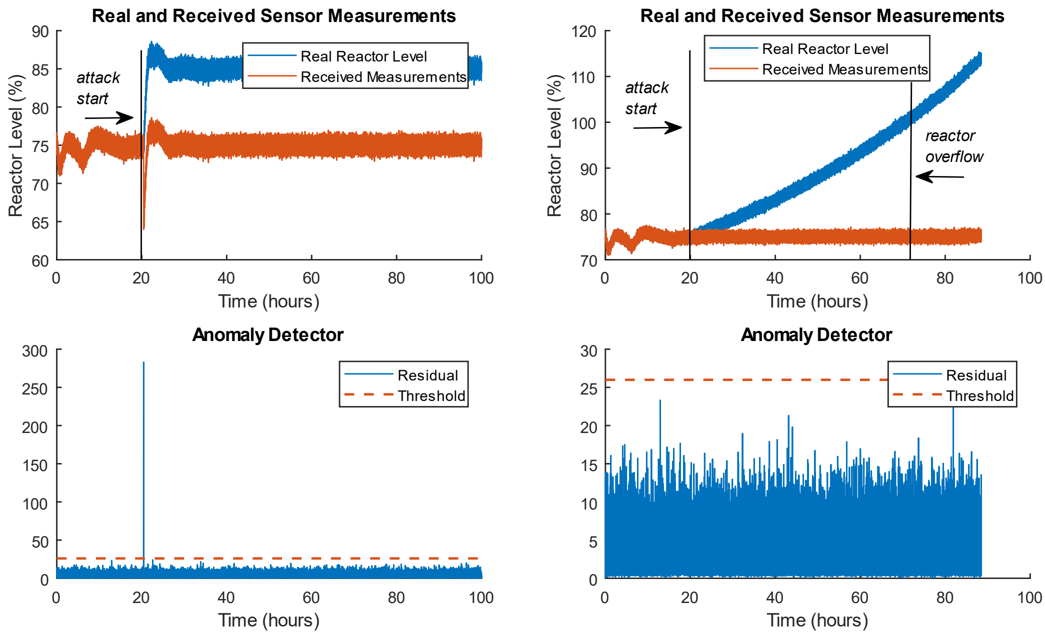
We adapt a scenario inspired by the Stuxnet (Albright et al. (2011)) and the German steel mill (Lee et al. (2014)) attacks. We consider the reactor stage of the TEP (area A_1) which features temperature, pressure, and level controllers for the main reactor that keep these operating variables at desired levels. We assume that each of these controllers is installed on a Siemens S7 PLC, which is connected to sensors and actuators using a network that employs a standard industrial *Profinet* ethernet. Due to control safety constraints, communications using the TCP/IP protocol are performed in plain text without any encryption (Ahmed et al. (2017)).

We consider a scenario involving an attacker that is seeking to damage valuable equipment in the plant. The attacker gains access to the network via a social engineering attack involving a fraudulent email that includes a malicious attachment. Their attack is divided into two phases: (1) in the reconnaissance phase, the attacker first identifies some details about the PLC's where temperature, pressure, and level controllers are implemented. The attacker particularly identifies the make, model, firmware, function codes, and addresses of the PLC's (Ahmed et al. (2017)). In addition, due to the lack of encryption, the attacker sniffs the data exchanged between the PLC and the physical system and uses it to extract more knowledge about the system, such as its model, noise properties, and details about the physics-based anomaly detector in use[1]. (2) In the second phase, the attacker manipulates the data exchanged between sensors and their respective controllers to cause damage by injecting false measurements (FDIA). We illustrate two different kinds of such manipulation. To showcase the physical effect of each attack, we use a simulation written in *MATLAB/Simulink* of the TEP (Bathelt et al. (2015); Ricker (1996)). For the purposes of this example, we augmented the simulation with a chi-squared anomaly detector (Murguia and Ruths (2018)). In the following, we discuss the simulation results based on two different kinds of FDIA attacks shown in Figure 3.3.

"Brute-Force" Attack. We illustrate first a "brute-force" FDIA on the reactor's level sensor. In this scenario, the attacker simply modifies the level measurement using an arbitrarily large value to force the controller into sharply increasing the level of liquids in the reactor. However, as shown in the simulations (Figure 3.3-a), the anomaly detector is able to easily detect this sensor modification, and raises an alarm almost immediately after the attack starts.

Stealthy Attack. In this scenario, the attacker takes advantage of the safety-critical operating mode of the reactor and the knowledge gathered during the reconnaissance phase (1). In particular, the attacker modifies level sensor values such that the bias between real and received values grows slowly over time while maintaining a residual below the anomaly detector's threshold. This in turn tricks the controller into slowly increasing the level of liquids in the reactor, which is

[1] We assume that the attacker possesses a vast amount of resources at their disposal to perform such activities. In the case of Stuxnet, it is generally agreed that the attackers had access to such resources (Lee et al. (2014)).



(a) – Brute Force Attack (b) – Stealthy Integrity Attack

Figure 3.3: Numerical simulations corresponding to the attack scenarios described in the motivating example (Section 3.2).

driven to the point of overflow (Figure 3.3-b). We can see from the simulation results that the chi-squared anomaly detector fails to raise an alarm before the reactor vessel overflows. This may have significant consequences, such as a fire, especially if the reactor is operating at high temperature or pressure.

3.2.3 Limitations of Existing Techniques

In this example, we illustrated through a simulation the limitations of existing anomaly detectors when faced with a knowledgeable attacker that can exploit system properties, like noise, along with the lack of security features within ICS network protocols. We note that while this particular example illustrates a physical model-based detector, similar limitations are featured within "model-agnostic" detectors that employ statistical modelling to reveal anomalies. Such detectors (e.g. the ones proposed by Aoudi et al. (2018) and by Krotofil et al. (2015)) may also fail in detecting attacks like the one illustrated in this section. This has been shown empirically in the work by Erba and Tippenhauer (2020). Such detectors cannot reliably learn important control-theoretical properties that a knowledgeable adversary may exploit to remain undetected. In this work, we focus on model-based anomaly detectors, and particularly the chi-squared anomaly detector as it is widely studied in the literature.

When it comes to online safety monitoring, most of the existing work on such monitoring under attacks (Carcano et al. (2011); Castellanos and Zhou (2019); Coletta (2018); Etigowni et al. (2018)) may not be able to predict potential damage from the illustrated attack since they rely on raw sensor values to measure a distance metric to unsafe states. In the case of the present example, the attacker has forced the received sensor values to appear lower than their real counterparts. Therefore, the evolution of the system towards an unsafe state will not be obvious if the proximity measure relies on raw sensor values.

In addition, data stored on low-level control devices (e.g. Programmable Logic Controllers' (PLC) configurations, sensor/actuator states) that could be useful during incident response, would be lost. With the lack of any alert about the attack before damage takes place, existing live ICS forensics techniques (Mohamed et al. (2020); Taveras (2013)) will not be able to recover potential evidence in time. As illustrated in the case study by van Vliet et al. (2015), where a wind turbine fire was being investigated, recovering data from damaged control devices can be extremely challenging, and some important data was lost.

It is worth noting that ICS are typically equipped with a process historian, a proprietary server that records process data over long periods of time. However, data recorded in process historians is generally geared towards supervising the physical process and informing high-level process-related decisions rather than detecting potential attacks or intrusions (Ahmed et al. (2017); Altschaffel et al. (2019); Janicke et al. (2015)). The data collected on these servers can be used to detect potential faults or anomalies arising in physical processes, but may not necessarily explain how a potential attack may have taken place. Hence, process historian data, even though it may be protected from being lost to physical damage, may not be sufficient on its own to inform forensic investigations. For example, in the work by Salvador et al. (2020), process historian data was used to design model predictive controllers for a water distribution system. The process historian database included values and trends of different control variables. Assuming that a similar process historian database was available for the TEP example provided previously, and that the server was protected from physical damage, operators may have access to reactor level measurements that were received by the PLC implementing the level control logic (the red line in Figure 3.3). However, this data may not be useful for detecting potential intrusions in a forensic investigation, as the measurements do not reflect the real physical state of the process. Furthermore, without any information about the configuration of the PLC, a piece of data that may not be collected by historians, a post-incident forensic investigation may not be able to reveal the code modifications performed in order to cause damage to the system.

3.3 Proposed EWS Framework

In this section, we present the framework for physics-based early warnings in ICS. We use this framework to instantiate our *Physics-based EWS* ([**COMP1**]). This framework primarily addresses [**RQ4**] and [**OBJ2**].

The proposed framework is inspired by the tiered approach for EWS used by Bradford et al. (2004) and Chivers et al. (2009). These approaches monitor "cyber-based" preliminary indicators, often called *weak evidence*, until some measure of their *evidentiary weight* crosses a certain threshold and triggers a warning. The main novelty of the present work lies in its consideration of stealthy attacks on ICS that affect the physical process. To this end, our framework defines *physics-based* preliminary indicators or weak evidence based on the monitoring of physical processes in ICS. Our framework then proposes a manner by which physics-based weak evidence can be accumulated and monitored in order to raise warnings and guide potential actions. In the following, we illustrate an instantiation of this framework before detailing the nature of physics-based "weak evidence" and the structure of our framework.

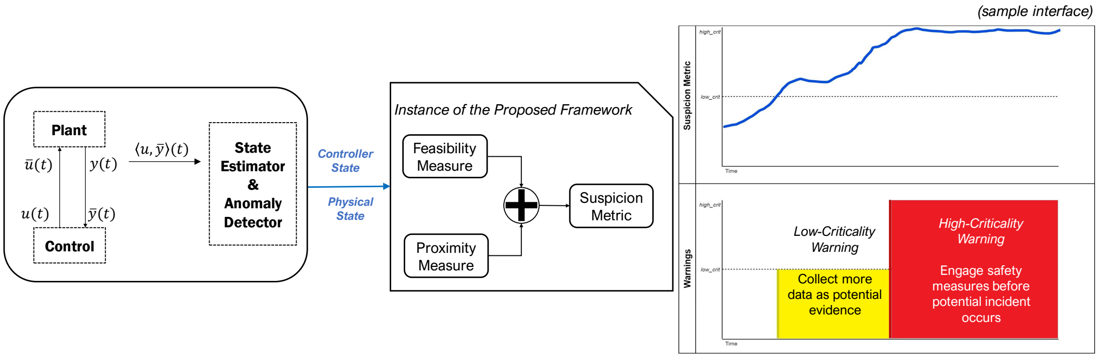


Figure 3.4: An instance of the proposed framework within a typical control system and a sample interface. *($y(t)$ = sensor measurements, $u(t)$ = control signal, $\bar{y}(t)$ = sensor measurements received by the controller, $\bar{u}(t)$ = control signal received by system — respectively at time t.)*

3.3.1 Framework Structure

Figure 3.4 shows an instantiation of our framework to a control system. The latter mainly takes as input estimates of physical state variables provided by some state estimator as well as the current state of the controller(s). These estimates are then used to measure two proposed physics-based preliminary indicators (called *grounds for suspicion*): *feasibility* of an attack and *proximity* of the system to unsafe states. These indicators act as weak evidence of a stealthy attack, and we consider them to be components of the likelihood of such an attack causing damage, which we refer to as the *suspicion metric*. The metric triggers warnings when crossing certain thresholds. Depending on the criticality of the crossed threshold, different actions may follow, such as further evidence collection or safety measures initiation. In our work we focus on the former; and we show how our framework defines different thresholds that can guide operators on whether to restrict their post-warning actions to data collection, or engage potentially more expensive preemptive safety measures.

3.3.1.1 Grounds for Suspicion

An attacker targeting the physical process and wishing to avoid detection will manipulate the system in a way that keeps the difference between estimated predictions and actual sensor readings sufficiently small. Our framework does not attempt to provide means to distinguish an anomalous behaviour by comparing the two. Instead, it defines the following *grounds for suspicion*:

- *Feasibility of a Stealthy Attack*: given the dynamics of the system and the constraints imposed by the anomaly detector, we check whether the current state of the system can be taken to an unsafe state (hence causing damage) while avoiding any alarm in the process, *if* a stealthy attack is to take place.

- *Proximity to Unsafe States*: given the current state of the system, *if* a stealthy attack is actually taking place without having been detected, we monitor how far would the system be from reaching an unsafe operating region. The closer the system is to this region, the more likely a stealthy attack is successful, as the attack may take less time to achieve its goal.

The feasibility of a *potential* stealthy attack and the proximity of the system to an unsafe state, as defined previously, do not necessarily imply that a malicious activity is taking place. However, they can indicate that *if* a stealthy attack were to take place, it may successfully damage the system. Hence, we consider them to be weak evidence of a potential stealthy attack, in the same manner a failed login attempt may be suspicious but cannot be used as evidence of an intrusion (Chivers et al. (2009)).

3.3.1.2 Suspicion Metric

The evidentiary weight of events that can trigger early warnings is typically measured using a certain metric. In traditional IT systems, existing EWS rely on monitoring certain software and network events that may constitute weak evidence of a potential attack. These events can be stored and aggregated over a certain period time to be later examined. However, the volume of such events renders such an approach infeasible (Kalutarage et al. (2012)). Work on EWS in IT systems usually employs a certain score or metric that reflects the weight of the observed weak evidence, as a more efficient means to aggregate such evidence. For example, Chivers et al. (2009) assign a Bayesian score to network nodes that generate events considered as weak evidence. This Bayesian score reflects the probability that a given node is subverted.

In the context of our model/physics-based approach, the proposed grounds for suspicion may be less numerous than the weak evidence usually proposed in the context of IT systems. However, the physics-based grounds for suspicion do not present themselves as concrete events that we can store in a given manner. Such preliminary indicators are features of the time-dependent behaviour of the physical process and can have model-specific measures. Consequently, we propose a score analagous to ones used in the context of IT systems (Chivers et al. (2009); Kalutarage, Lee, Shaikh and Sung (2015)), called *Suspicion Metric*. This metric essentially measures in real-time the likelihood that a *potential* stealthy attack will cause damage to the system before being detected by combining these two grounds for suspicion. At runtime, a human operator would be provided with the evolution of the suspicion metric over time (Figure 3.2). If the metric crosses a certain threshold, a warning is raised.

3.3.2 Suspicion Metric Soundness Principles

The proposed suspicion metric, being a measure of indicators that present as features of the physical process, will have a realisation that depends on the model we choose for the concerned physical process. As several types of physical systems can be modelled with different formalisms (e.g. continuous-state vs. discrete-event/hybrid), we do not attempt to propose a general formula to compute a suspicion metric. Instead, we provide *soundness principles*, so that irrespective of the system where the framework is instantiated, the metric can reflect the grounds for suspicion in a sound manner:

1. The metric must include at least one clear threshold which if crossed, a warning is issued. We propose two thresholds (Figure 3.2): (i) one of *low-criticality*, which may trigger intensive data collection to proactively check for intrusions; and another (ii) of *high-criticality* typically triggering measures preventing a safety incident. The main difference between the two

warning levels is that the first may suggest that operators still have time before potential damage can happen; therefore, they can restrict their actions to a collection of data that may eventually reveal a certain intrusion and in turn could be used to stop a potential attack. The second level of warning implies a higher likelihood of damage taking place and as a result requires more immediate measures; for example engaging backup redundant controllers.

2. The metric should increase over a certain time interval if (a) the likelihood of the real physical state of the system diverging from the provided estimate and evolving into an unsafe operating region is increasing (feasibility); or (b) if the system is evolving closer to unsafe states meaning that a potential attack is less and less time consuming for the attacker (proximity).

The first principle ensures that the EWS can advise an operator about the current security/safety situation, and can trigger in a sound manner different types of proactive measures. In particular, our thresholds seek to ensure that a proper justification is in place for potentially restricting proactive measures to the collection of more data rather than engaging preemptive measures. For example, if the physics-based EWS predicts that *if* the system is breached, the attacker would need a large amount of time to cause damage (before getting detected) with a low probability, then simple data collection is a reasonable proactive measure to undertake. In this scenario, engaging potentially expensive preemptive safety measures, such as switching to backup controllers or activating a certain fail-safe, may incur unnecessary costs given the low likelihood of damage.

The second soundness principle ensures that the metric provides a sound measure of the evidentiary weight of the grounds for suspicion and can in fact inform operators about the likelihood of success of a potential stealthy attack.

3.3.3 Framework Configuration Requirements

The configuration of the framework involves mainly providing model-based measures of feasibility and proximity to construct a suspicion metric according to the soundness principles provided earlier. This can be performed by reusing existing techniques proposed in the domain of CPS/ICS security. For example, the feasibility indicator intuitively lends to a reachability/safety monitoring problem. Therefore, techniques to compute reachable sets under a given stealthy attack (Kwon and Hwang (2018); Murguia and Ruths (2018)) can be used to measure feasibility, while distance metrics such as the Euclidean or Hamming distance, can be used to measure proximity given the real-valued nature of most physical state variables.

Therefore, to configure the framework, we require mainly (i) a mathematical model of the system, including its controller and the unsafe operating region; (ii) information on the used anomaly detection method; and (iii) a model of an attack at the level of physical process. Note that (i) is standard in control engineering and process safety analysis, while the physics-based threat model (iii) is only required to show the effect of a potential attack on the control loop. There exist several works (Huang et al. (2009); Teixeira, Shames, Sandberg and Johansson (2015)) that provide effective ways of modelling such effects.

3.4 Summary

In this chapter, we provided an overview of the approach we propose to support the forensic readiness of ICS faced with the threat of stealthy attacks on physical processes. Our problem consists of three main facets: (P1) a trigger for data collection that is more reliable than anomaly detector alarms; (P2) a way to identify relevant data or data at risk of being damaged; and (P3) a way to decide which relevant data is worth collecting.

We described in this chapter a framework for physics-based EWS that seeks to define a sound manner by which warnings about potential damage from a stealthy attack could be provided well ahead of time. The framework defines physics-based preliminary indicators (grounds for suspicion) of a stealthy attack that incorporate some awareness of the limitations of the existing anomaly detector. We propose to reason about these indicators using model-based measures combined in a so-called suspicion metric. This metric reflects the probability that a certain stealthy attack is able to cause damage without being detected.

We provide soundness principles to guide the instantiation of this framework to a particular system model. Aside from ensuring that a suspicion metric presents a sound measure of preliminary indicators, our soundness principles also ensure that sound triggers for different proactive measures are provided. In particular, we propose that a low-criticality warning implying a low likelihood of damage is used as a trigger for proactive data collection. Having a low likelihood of damage may not justify the use of more expensive safety measures.

Chapter 4

Online Safety Monitoring and Early Warnings for Stealthy Attacks

In the previous chapter, we described the general approach that we propose to address Forensic Readiness (FR) in ICS faced with the threat of stealthy attacks. The first part of the problem, (P1), is concerned with the need for a timely trigger for proactive data collection, which can provide an alternative to anomaly detection alarms. As illustrated in our motivating example (Section 3.2), anomaly detectors may be unreliable in triggering timely collection of data when the system is under a stealthy attack.

The *Physics-based EWS* ([**COMP1**]) component of our approach is defined through a framework which seeks to warn about potential damage to the system using two preliminary indicators (grounds for suspicion). These indicators compose a suspicion metric, which measures the probability that *if* the system is subjected to a stealthy attack, then the latter can cause damage before any alarm can be raised by the existing anomaly detector. As physical processes in ICS can be modelled in a variety of different formalisms, we described the framework in a general manner without attempting to provide a solution that covers all model classes. In this chapter, we focus on a particular class of control systems, the Linear Time-Invariant (LTI) system, for which we demonstrate the applicability of the framework. LTI systems have been widely studied in the literature (Lun et al. (2019)) and in most industrial settings, nonlinear systems are linearised around a certain operating point (Polisetty et al. (2019)). In order to instantiate the framework, we need a predictive online safety monitoring algorithm that can check whether the system can be taken to an unsafe state through a stealthy attack. In this chapter, we first address [**OBJ3**] and [**RQ5**] — *How can we perform online safety monitoring to predict potential damage from a stealthy attack in large-scale ICS, with an acceptable real-time performance?*

To enable efficient and scalable monitoring in real-time, we propose an approach that consists of performing the most computationally expensive operations offline which would result in a symbolic ellipsoid approximation of reachable sets. We then instantiate these symbolic sets given an online physical state estimate, and we take advantage of existing results in ellipsoid calculus to perform fast real-time safety checks, effectively reducing them to checking the sign of a distance measure.

In the second part of this chapter, we address [**OBJ1**] and [**RQ1**] — *When should we trigger proactive data collection activities in ICS given that anomaly detectors may not raise an alarm about a stealthy attack in a sufficient period of time (i.e. before damage can happen)?*

We provide specific measures of feasibility and proximity based on the predictive safety checks performed by the online safety monitoring algorithm. We then combine these measures into one *Suspicion Metric* while accounting for Soundness Principle 1 that we proposed within our EWS framework. Then, given Soundness Principle 2, we propose sound thresholds that provide meaningful warnings to operators, and we propose certain post-warning measures based on the criticality of the warning. We set a threshold for a *low-criticality* warning which implies that the likelihood of the system being damaged by a stealthy attack is low, and that the system, if it is indeed under attack, would still need quite some time to reach unsafe states. Consequently, such a warning constitutes a suitable trigger for proactive data collection; thus addressing problem (P1).

We begin this chapter with an overview of the *Physics-based EWS* ([**COMP1**]) approach for LTI systems faced with the threat of stealthy attacks in Section 4.1. We then proceed with a description of the system and attack model in Section 4.2, before detailing the proposed monitoring algorithm in Section 4.3. We describe

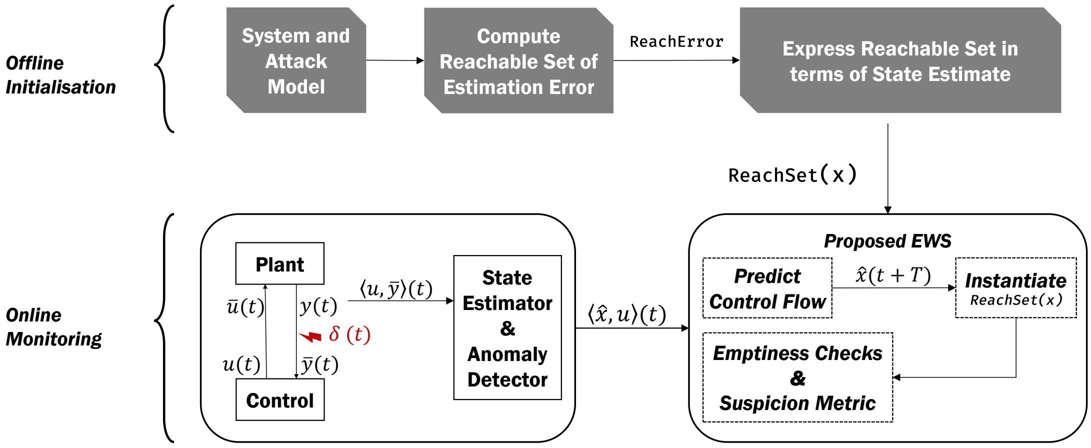


Figure 4.1: Proposed instantiation of the framework for *Physics-based Early Warning System (EWS)* (**[COMP1]**) to Linear Time-Invariant (LTI) systems. (*Respectively, $y(t)$ and $\bar{y}(t)$ denote sent and received sensor measurements; $u(t)$ and $\bar{u}(t)$ denote sent and received actuator signals (control inputs); $\hat{x}(t)$ denotes estimated physical state; $\delta(t)$ denotes potential attack signal; at a time t. T refers to a certain time horizon for the prediction of future states.*)

our design of the suspicion metric and its thresholds in Section 4.4. Finally, we provide some notes on the complexity of our real-time computations and we propose some improvements to their performance in Section 4.5 before concluding this chapter with Section 4.6.

4.1 Overview

The proposed instantiation of the *Physics-based (EWS)* (**[COMP1]**) framework to LTI systems is outlined in Figure 4.1. We formulate the suspicion metric computation as a reachability problem. Namely, given the real-time estimated physical state of the system, the reachability problem asks whether a stealthy attack can cause damage to the system without being detected. In other words, given the current state estimate of the system $\hat{x}(t)$, we check whether there exists an attack conforming to a predefined stealthy attack model, that can bring the system into an unsafe set over the next T time instants.

If this check returns a negative result, then operators can be reassured that even if an intrusion is present, the alleged attacker may not be able to cause any damage without being detected. Otherwise, the check can serve as an early warning within the context of the framework we proposed in Chapter 3. To enable efficient reachability analysis in real-time, the proposed online safety monitoring algorithm takes advantage of an offline initialisation step that eases some of the computational burden that needs to be performed online. Accordingly, the proposed approach, shown in Figure 4.1, is composed of two main steps:

- *Offline Initialisation:* this step consists of computing symbolic reachable sets under a stealthy false data injection attack in terms of the state estimate based on a model of the system and attack. This is possible by considering the evolution of the state estimation error under a stealthy attack instead of the physical state itself. As a result, we express the reachable set of errors in terms of the actual state estimate. This allows us to perform the bulk of the computation offline, leading to more efficient real-time safety checking.

- *Online Monitoring:* at runtime, the proposed monitor takes as input the current physical state estimate and the state of the controller and predicts the value of the state up to T time units into the future. We assume that this prediction can be done using an identified physical model of the system, with T chosen to maintain an acceptable degradation in the confidence level of the predicted state. We then instantiate the pre-computed symbolic reachable set at each predicted state value, and perform an emptiness check of its intersection with a predefined set of unsafe states. The prediction stops when a non-empty intersection is encountered, or when T is exhausted. Furthermore, we compute two security metrics when the intersection is non-empty: the first is a measure of (i) *feasibility* of the stealthy attack, and the second is a measure of (ii) *proximity* of the system to unsafe state. We then combine these metrics into a *suspicion metric* that reflects the probability of a potential stealthy attack to cause damage to the system. We design these metrics in such a way that the soundness principles proposed in our EWS framework are satisfied.

In this chapter, we apply this general approach to LTI systems, where we propose the use of ellipsoids as over-approximations of the symbolic reachable sets. Ellipsoids have been extensively used for safety verification for control systems (Hu et al. (2020); Kurzhanski and Varaiya (2000)). They feature an efficient quadratic representation in terms of the dimension of the state of the system (Le Guernic (2009)), which presents an advantage in real-time monitoring. Furthermore, in most practical applications of process control, unsafe operating regions can be represented as unions of half-spaces. With reachable sets represented as ellipsoids, safety checking reduces to checking the sign of the distance from the ellipsoid to each of the hyperplanes composing the unsafe set (Boyd et al. (2004); Kurzhanski and Varaiya (2000)). As a result, real-time safety monitoring is enabled with minimal resource utilisation.

Based on this reachability analysis and geometric setup, our instantiation of the feasibility metric is given in terms of the size of the ellipsoid approximation of the reachable set's intersection with the set of unsafe states. This computation is made possible in real-time by taking advantage of certain results in ellipsoid calculus (Dai and Pelckmans (2012)). For the proximity metric, we rely simply on the approximate time to unsafe states as given by our prediction of the future (potentially violating) state of the system.

4.1.1 Existing Monitoring Techniques

To the best of our knowledge, existing online safety monitoring techniques under physics-based attacks, reviewed in Section 2.5.2 (Carcano et al. (2011); Castellanos and Zhou (2019); Coletta (2018); Etigowni et al. (2018)), rely mainly on a proximity metric to assess the safety of the system against attacks targeting physical processes. In this section, we provide a comparison based on intuition between our approach and these existing techniques which all feature the same idea detailed in the following.

Given a set of unsafe states $\mathcal{S}_u$, and the current estimated state $\hat{x}(t)$, existing online monitoring techniques (Carcano et al. (2011); Castellanos and Zhou (2019); Coletta (2018); Etigowni et al. (2018)) compute a distance metric $d_u =$

$\min d(\hat{x}(t), \mathcal{S}_u)$. The most commonly used distance metric is the Euclidean distance, which is mainly suitable for continuous or hybrid systems where state variables of interest assume real values. Given a set of thresholds $\tau_1 > \tau_2 > \tau_3 \ldots$, alarms of different levels of criticality can be raised based on the value of d_u. The proximity metric can further account for the dynamics of the system by computing a *time-to-unsafe/critical states* metric, $t_u = d_u/r$; where r is the approximate rate of change of the system state over a given time period (Castellanos and Zhou (2019)). In the rest of the thesis, the term "traditional time-to-unsafe metric" refers to t_u computed using the aforementioned formula.

However, under a stealthy attack which slowly drives the system into unsafe states to avoid detection, the real state of the system $x(t)$ may diverge from the estimated state $\hat{x}(t)$. Thus, the metrics d_u and t_u may not provide an accurate measure of the proximity of the system to unsafe states. Instead of relying merely on a proximity measure based on $\hat{x}(t)$, our online monitoring approach accounts for the possibility of stealthy attacks by considering reachable sets under such attacks. If $\mathcal{R}_x(t)$ is the reachable set of states under a stealthy attack given the used anomaly detector at time t, then $x(t) \in \mathcal{R}_x(t)$ even if $x(t) \neq \hat{x}(t)$. Otherwise, by definition of the reachable set $\mathcal{R}_x(t)$, the attack would be detected by the anomaly detector. If over the next T time period, $\mathcal{R}_x(t') \cap \mathcal{S}_u \neq \emptyset$ for some $t' \in [t; t+T]$, then it is possible for an attacker to drive the system into the unsafe operating region within at least $t' - t$ time units. Hence, our safety monitoring approach relies on emptiness checking of this intersection, instead of a mere proximity measure. Furthermore, we consider a version of the term $t' - t$ as a measure of proximity and a component of the suspicion metric as defined in our EWS framework.

Note that in this problem setup, we assume that sensors may only be subjected to a stealthy attack satisfying a given model. In particular, we assume that the sensors are not subjected to potential faults (Park et al. (2017)). This guarantees that in our setup, the real system state belongs (up to a degree of uncertainty) to the reachable set of states under attacks. Before moving into the details of the proposed online safety monitoring algorithm, we describe in the next section the LTI modelling framework to which the EWS framework is instantiated.

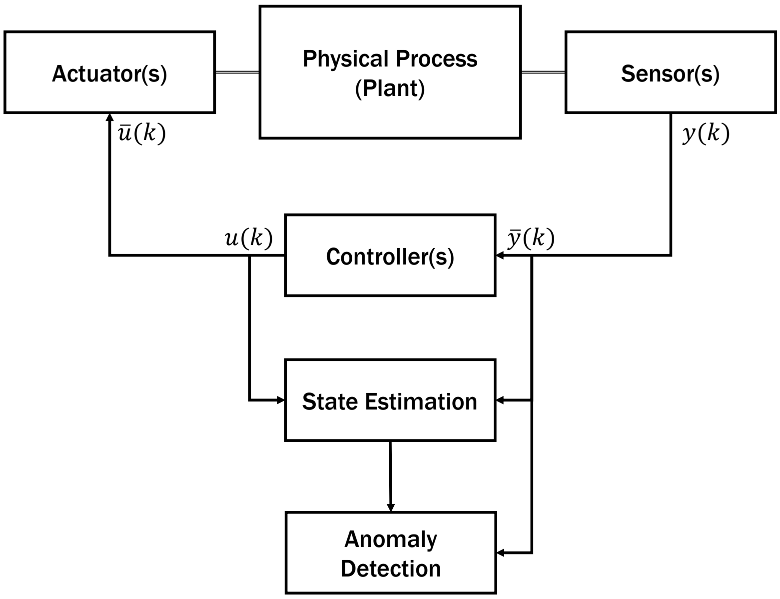


Figure 4.2: A typical feedback control loop with a state estimator and and an anomaly detector.

4.2 Modelling Framework

We describe in this section the layout of the system, the main elements of a Linear Time-Invariant (LTI) model, and the stealthy attack model.

4.2.1 System Layout

Consider the control system architecture in Figure 4.2. We assume that the physical process can be approximated using a Linear Time-Invariant (LTI) model:

$$\begin{cases} x(k+1) = Ax(k) + B\bar{u}(k) + w(k) \\ \quad y(k) = Cx(k) + v(k) \end{cases} \tag{4.1}$$

Where $x(k) \in \mathbb{R}^n$ is the state vector, $\bar{u}(k) \in \mathbb{R}^l$ denotes control signals received by the system, and $y(k) \in \mathbb{R}^m$ is a vector of sensor measurements. $w(k) \in \mathbb{R}^n$ and $v(k) \in \mathbb{R}^m$ denote process and measurement noise, respectively, and are assumed to follow a zero-mean Gaussian distribution with respective covariance matrices Σ_1 and Σ_2. $k = t/\Delta_t \in \mathbb{N}$ denotes discrete time instants, where Δ_t is the sampling period and t is continuous time. We adopt a discrete-time LTI model as in practical applications, differential equations that reflect the behaviour of physical processes are discretized to allow for the design of digital controllers. A, B, and C are real, time-invariant matrices of appropriate dimensions. The system is assumed to be equipped with an output feedback control loop, such that $u(k) = \mathcal{K}(\bar{y}(k) - y_r(k))$, where $u(k) \in \mathbb{R}^l$ denotes control signals originally sent to the process, $\mathcal{K}(.) : \mathbb{R}^m \to \mathbb{R}^l$ is the control law, $y_r(k)$ denotes reference output, and $\bar{y}(k)$ denotes measurements received by the controller. To allow for the modelling of attacks that modify the signals exchanged between the physical process and the controllers, we distinguish between *actual* sensor measurements or system output, denoted by $y(k)$, and *received* sensor measurements, denoted by $\bar{y}(k)$. Similarly, we adopt the same distinction between actual controller inputs (actuator signals) $u(k)$ and received controller inputs $\bar{u}(k)$.

Furthermore, we assume that a subset of state variables, denoted as critical, are grouped in the vector $x_c = C_c x,\ x_c \in \mathbb{R}^{n_c},\ C_c \in \mathbb{R}^{n_c \times n}$, and are required to remain within a certain safe set to ensure safe operation. Let $\mathcal{S}_u$ be the set of unsafe states; we assume that unsafe conditions are given as a linear combination of the critical state variables, such that the unsafe set becomes a union of half-spaces[1]:

$$\mathcal{S}_u = \left\{ x(k) \in \mathbb{R}^n \mid \bigcup_{i=0}^{n_c} C_{c,i} x_i(k) \geq b_i \right\} \tag{4.2}$$

Where $b_i \in \mathbb{R}$ is the i^{th} half-space scalar, $C_{c,i}$ denotes the i^{th} row of the matrix C_c, and $x_i(k)$ the i^{th} element of $x(k)$.

At a time k, given previous sensor measurements and control actions, a Kalman filter generates an estimate of the physical state and expected sensor measurements as follows:

$$\begin{cases} \hat{x}(k) = A\hat{x}(k-1) + Bu(k-1) + L(\bar{y}(k-1) - C\hat{x}(k-1)) \\ \hat{y}(k) = C\hat{x}(k) \end{cases} \tag{4.3}$$

Where L denotes the Kalman gain, and $\hat{x}$ and $\hat{y}$ denote estimated state and measurements, respectively.

[1]This assumption is typical in several process control applications.

In addition, a chi-squared anomaly detector compares received measurements with the generated estimate through a residual $r(k) := \bar{y}(k) - \hat{y}(k)$. Under nominal conditions, the residual metric has a zero-mean and a covariance matrix $\Sigma \in \mathbb{R}^{m \times m}$. To check for this hypothesis, a chi-squared metric, $z(k) = r^T(k)\Sigma^{-1}r(k) \in \mathbb{R}$ is computed and compared with a threshold τ, such that exceeding this threshold implies a possible anomaly and raises an alarm. The threshold τ is designed to maintain a certain false alarm rate β such that $\Pr[z(k) \leq \beta] = 1 - \beta$ under nominal operation, and can be set as described by Murguia and Ruths (2018) (Proposition 1).

4.2.2 Attack Model

Attacks affecting physical processes in an ICS can be classified into two main categories (Section 2.4.2.1, Figure 2.5): (i) Deception attacks that compromise control data integrity, and (ii) Denial-of-Service (DoS) attacks that compromise their availability. DoS attacks are easier to detect than deception attacks since the absence of sensor measurements or control signals can be revealed by digital controllers. The main challenge with DoS attacks is to distinguish them from network faults (Teixeira, Shames, Sandberg and Johansson (2015)), which falls outside the scope of this work. Therefore, we focus in this work on deception attacks — which include replay attacks and False Data Injection Attacks (FDIA's).

We consider in this work FDIA's on sensors which are masked by the system noise, in order to drive the latter slowly to the unsafe set (4.2). While different stealthy attack strategies exist in the literature, we choose to focus on one that is feasible in practice. Hence, we provide the following justification for our particular choice as opposed to other possible stealthy attacks:

1. We consider FDIA's specifically on *sensors* as it has been shown both theoretically (Kwon et al. (2013)) and empirically (Urbina et al. (2016)) that their stealthiness is easier to maintain than attacks on actuators. FDIA's that consist of corrupting all sensor and actuator channels, also known as "covert" attacks (Smith (2015)), even though undetectable, require a significant amount of resources to be executed (Teixeira, Shames, Sandberg and Johansson (2015)). In practice, it may not be possible for an attacker

to have simultaneous access to all sensor/actuator communication channels or associated control devices. Each sensor/actuator may require different kinds of software/hardware tools in order to corrupt the data they send or receive (Akowuah and Kong (2021)).

2. Replay attacks, similarly to FDIA's, threaten the integrity of sensor measurements. Although they require less resources to remain stealthy (Teixeira, Shames, Sandberg and Johansson (2015)), the stealthiness of replaying old sensor measurements relies on whether these measurements could be admissible at a current time. Therefore, the system needs to be operating in a steady state for the replayed measurements to be considered nominal by the anomaly detector. If transients are experienced, then replaying old measurements would reveal the attack since they no longer correspond to the current control inputs. In this work, we consider process control systems whereby long transients may be experienced, thus limiting the extent to which stealthy replay attacks may be successful.

In our attack model, we assume that the attacker has sufficient resources and knowledge about the system, including knowledge of the system dynamics and anomaly detector properties. Let $\{k_s, \ldots, k_f\}$ denote the time period of the attack; we model the attack as a bias imposed on sensor measurements:

$$\bar{y}(k) := y(k) + \delta(k) \; \forall \; k \in \{k_s, \ldots, k_f\} \tag{4.4}$$

The attack signal $\delta(k)$ remains stealthy by ensuring that the false alarm rate is maintained throughout the attack period. We use this characterisation because anomaly detectors do raise alarms under nominal operation. A sudden disappearance of these alarms in practice may raise suspicion in operators and lead them to uncover the attack before it can cause damage (Hashemi et al. (2018)). Under the attack in (4.4), the residual is given by: $r(k) = \bar{y}(k) - \hat{y}(k) = y(k) - \hat{y}k + \delta(k)$. Hence, the chi-squared metric under attack is given by:

$$z(k) = (y(k) - \hat{y}(k) + \delta(k))\Sigma^{-1}(y(k) - \hat{y}(k) + \delta(k)) \tag{4.5}$$

By selecting $\delta(k)$ to be such that $\Pr[z(k) \leq \tau] = 1 - \beta$, the attacker manages to remain undetected. For example, given K time steps, the attacker may choose to raise alarms for βK steps so that the false alarm rate is mimicked as closely as possible (Hashemi et al. (2018)). This is possible since we assume that the attacker knows the detector's parameters, i.e. Σ, β and τ, in addition to the system and estimator outputs; i.e. $y(k)$ and $\hat{y}(k)$ respectively. In practice, such information could be obtained, for example, through reconnaissance attacks or insider knowledge. In addition, while this attack strategy is specifically designed for chi-squared detectors, most existing work in model-based anomaly detection employs this statistical change detection test (Murguia and Ruths (2019)).

We note that optimal stealthy FDIA strategies that take advantage of system noise are studied in the literature (Guo et al. (2018); Zhang and Yang (2021); Zhang et al. (2021)). However, our choice for the previous attack model stems from its relative simplicity and practicality. In works describing optimal stealthy attacks, the attacker requires a deeper knowledge and understanding of the dynamics of the system. With the adversary model that we adopt, the attacker is required to only know the basic parameters of the system model and its anomaly detector, which are relatively easily obtained through a reconnaissance phase. Furthermore, optimising the attack strategy comes at a significant computational cost for the attacker and thus limits the attack's real-time performance, especially if the attacker needs to adapt to an adaptive defender. We also note that our attack model requires only the attack to mimic the false alarm rate of the anomaly detector. Consequently, it can be applied to optimised attack strategies that also take advantage of the false alarm rate to remain stealthy.

4.3 Proposed Monitoring Algorithm

The proposed approach relies mainly on the offline computation of the symbolic reachable set of estimation error under the attack described in (4.4). This set is an ellipsoidal over-approximation of the exact reachable set, parametrised by the state estimate. In real-time, given the state estimate at time k, the symbolic set is instantiated at the K-step predicted state. Emptiness checks of its intersection with $\mathcal{S}_u$ are then performed. We detail both the offline and online computations in this section.

4.3.1 Offline Computation of Symbolic Reachable Set

To compute the reachable set of the estimation error under the attack in (4.4), we use the method described by Murguia and Ruths (2018). We define this error as $e(k) := x(k) - \hat{x}(k)$, and assume that at the start of an attack the estimation error is always almost zero. The reachable set of the estimation error under the attack in (4.4) is independent of the actual physical state at the start of the attack. Hence, this set serves as a symbolic reachable set parametrised by the state estimate.

By setting $e(k) = x(k) - \hat{x}(k)$, and performing some algebraic manipulations of Equation (4.3), the evolution of the estimation error under an attack is given by:

$$\begin{aligned} e(k+1) &= Ae(k) - L(Ce(k) + v(k) + \delta(k)) + w(k) \\ &= Ae(k) - L(y(k) - \hat{y}(k) + \delta(k)) + w(k) \end{aligned} \tag{4.6}$$

Since the error is partially driven by the Gaussian noise $w(k)$ and the attack-dependent sequence $\bar{\delta}(k) = y(k) - \hat{y}(k) + \delta(k)$, using a deterministic approach to compute the reachable set will yield an unbounded set, as the support of $w(k)$ and $\delta(k)$ (as characterised in (4.4) and (4.5)) is infinite. This issue can be overcome by setting a confidence level on the energy of both of these vectors. For the attack, the sequence $\bar{\delta}(k)$ is already constrained to be such that $\Pr[z(k) \leq \tau] = \Pr[\|\Sigma^{-1/2}\bar{\delta}(k)\|^2 \leq \tau] = 1 - \beta$ where $\|.\|$ denotes the L_2-norm. For the noise, let $p = \Pr[\|w(k)\|^2 \leq \bar{w}]$; since $w(k)$ follows a zero-mean Gaussian distribution, the bound $\bar{w}$ on $\|w(k)\|^2$ can be determined using the gamma distribution for a desired confidence p.

By using this assumption, the resulting reachable set can be interpreted as a level set of the distribution of the reachable error. A larger confidence level would lead to a larger set, at the cost of being overly conservative with the safety checking. A reasonable choice for p would be $1-\beta$, as the false alarm β is designed to be small. This also simplifies the computation of the reachable set, since for $p = 1 - \beta$, we readily have $\Pr[\|w(k)\|^2 \leq \bar{w}] = \Pr[z(k) \leq \tau]$ under the attack in (4.4). The following is based on this choice; for a more detailed treatment of this confidence level and a comparison of reachable sets under different choices of p, the reader is referred to Murguia and Ruths (2018) and Hashemi et al. (2018).

Let $\mathcal{R}_e^p$ denote the reachable set of error under the attack in (4.4) and a confidence level $p = 1 - \beta$:

$$\mathcal{R}_e^p := \{e(k) \in \mathbb{R}^n \mid e(k) \text{ is s.t. } (4.6),\ p = \Pr[\|w(k)\|^2 \leq \bar{w}] = 1 - \beta\} \tag{4.7}$$

While computing $\mathcal{R}_e^p$ is intractable, it is possible to over-approximate the set using an ellipsoid in $\mathbb{R}^n$, given by:

$$\mathcal{R}_e^p \subseteq \mathcal{E}_e^p = \{e(k) \mid e^T(k)\mathbf{\Pi}^{-1}e(k) \leq 1\} \tag{4.8}$$

Where the positive definite matrix $\mathbf{\Pi}$ is the ellipsoid's shape matrix. Letting $\mathcal{P} = \mathbf{\Pi}^{-1}$, the minimum volume ellipsoid containing the set $\mathcal{R}_e^p$ can be obtained by solving the following semi-definite programme (Murguia and Ruths (2018)):

$$\begin{aligned} \mathcal{P} = \arg\ & \min -\log\det\mathcal{P} \\ & \text{s.t.} \\ & \mathcal{P} > 0\ ;\ \mathcal{Q} \geq \mathbf{0} \end{aligned} \tag{4.9}$$

Where:

$$\mathcal{Q} = \begin{bmatrix} b\mathcal{P} & A^T\mathcal{P} & \mathbf{0} & \mathbf{0} \\ \mathcal{P}A & \mathcal{P} & \mathcal{P} & -\mathcal{P}L\Sigma^{1/2} \\ \mathbf{0} & \mathcal{P} & \frac{1-b}{\tau+\bar{w}}I & \mathbf{0} \\ \mathbf{0} & -\Sigma^{1/2}L^T\mathcal{P} & \mathbf{0} & \frac{1-b}{\tau+\bar{w}}I \end{bmatrix}, \tag{4.10}$$

$$b \in (0,1)$$

Note that while b is an optimisation variable, it is necessary to fix it to ensure the convexity of the programme. A grid search can then be performed over the interval $(0, 1)$ to find the optimal shape matrix corresponding to the minimum-volume ellipsoid.

Given the shape matrix $\mathbf{\Pi} = \mathcal{P}^{-1}$, and replacing $e(k)$ by its definition, we obtain a symbolic ellipsoidal approximation $\mathcal{E}_x^p(k)$ of the reachable set $\mathcal{R}_x^p(k)$ of the actual system state $x(k)$, parametrised by the current state estimate $\hat{x}(k)$:

$$\mathcal{R}_x^p(k) \subseteq \mathcal{E}_x^p(k) = \{x(k) \in \mathbb{R}^n \mid (x(k) - \hat{x}(k))^T\mathbf{\Pi}^{-1}(x(k) - \hat{x}(k)) \leq 1\} \tag{4.11}$$

Algorithm 1 summarises the offline steps to obtain $\mathcal{E}_x^p(k)$. Given the system matrix A, the Kalman gain L, the residual covariance matrix Σ, the anomaly

Algorithm 1 Offline Symbolic Reachable Set Computation

INPUTS: $(A, L, \Sigma, \tau, \bar{w}, \Delta h)$; $0 < \Delta h < 1$

OUTPUT: Reachable set shape matrix $\mathbf{\Pi}$

```
b ← Δh;
SolutionList ← EmptyList();
while b < 1 do
    ▷ Solve the programme in (4.9) for the current value of b
    SolveSemiDefiniteProgramme(A, L, Σ, τ, w̄, b);
    SolutionList.append(CurrentSolution);
    b ← b + Δh;
end while
BestShapeMatrix ← MinObjectiveValue(SolutionList);
return BestShapeMatrix;
```

detector's threshold τ and the confidence bound $\bar{w}$, the algorithm performs a grid search over $(0, 1)$ by partitioning the interval into segments of length Δh. The choice of Δh will depend on the desired tightness of the ellipsoidal approximation given the computational resources available. Note that this step only needs to be performed offline once, and only the matrix $\mathbf{\Pi}$ needs to be stored to instantiate $\mathcal{E}_x^p(k)$ online given a state estimate $\hat{x}(k)$.

4.3.2 Online Safety Checks

Algorithm 2 outlines the steps needed to perform online safety checks. Given the current state estimate $\hat{x}(k)$ and the state of the controller, we estimate the state of the system for K time steps into the future using the identified model of the plant. At each time step $l \in \{0, \ldots, K\}$, we instantiate $\mathcal{E}_x^p(k + l)$ and we check whether it intersects the set $\mathcal{S}_u$. The algorithm halts and reports an unsafe state when a non-empty intersection is encountered. If the prediction horizon is exhausted, the algorithm reports a safe state. In the following, we detail the procedure we use to perform the emptiness checks.

Let $\mathcal{H}_i = \{x \in \mathbb{R}^n \mid C_{c,i}x \geq b_i\}$ be a half-space representing one of the safety constraints composing the set $\mathcal{S}_u$ (Equation (4.2)). Checking whether $\mathcal{E}_x^p(k_f) \cap$

$\mathcal{S}_u = \emptyset$ involves checking whether $\mathcal{E}_x^p(k_f) \cap \mathcal{H}_i = \emptyset$ for each $i \in \{1, \ldots, n_c\}$. If the latter is true for all i, then the former is also true, since $\mathcal{S}_u = \cup_{i=1}^{n_c} \mathcal{H}_i$. To check whether $\mathcal{E}_x^p(k_f) \cap \mathcal{H}_i = \emptyset$, it suffices to compute the minimum distance from $\mathcal{E}_x^p(k_f)$ to the hyperplane that delimits the half-space $\mathcal{H}_i$.

Let $\mathcal{H}_{p,i} = \{x \in \mathbb{R}^n \mid C_{c,i}x = b_i\}$ be such hyperplane, the minimum distance from $\mathcal{E}_x^p(k_f)$ to $\mathcal{H}_{p,i}$ is given by Kurzhanskiy and Varaiya (2006):

$$d_i(k_f) = \frac{|b_i - C_{c,i}x(k_f)| - \sqrt{x(k_f)^T \mathbf{\Pi} x(k_f)}}{\|C_{c,i}^T\|} \tag{4.12}$$

If $d_i(k_f) < 0$, then $\mathcal{E}_x^p(k_f) \cap \mathcal{H}_i \neq \emptyset$. Otherwise, if $d_i(k_f) \geq 0$, then the ellipsoid $\mathcal{E}_x^p(k_f)$ is either contained in $\mathcal{H}_i$ or does not intersect the half-space, depending on whether its centre $\hat{x}(k_f)$ belongs to $\mathcal{H}_i$. However, since the state estimate is assumed to be within the safe region (otherwise, it would be clear that the system is evolving to an unsafe state and Algorithm 2 in this case would become obsolete), i.e. $\hat{x}(k_f) \notin \mathcal{H}_i$, then in our case, $d_i(k_f) > 0$ always implies that $\mathcal{E}_x^p(k_f) \cap \mathcal{H}_i = \emptyset$.

4.4 Real-time Measures of Grounds for Suspicion

In the previous section, we proposed an online safety monitoring algorithm that takes advantage of an offline (expensive) computation of symbolic reachable sets to reduce the online safety checking to a computation of a distance measure. Using the ellipsoid nature of the approximated reachable sets, we derive in this section metrics that reflect the feasibility of the attack and the proximity of the system to unsafe states. We then combine these two metrics into what we call a suspicion metric, which essentially reflects the likelihood of damage happening due to a potential undetected stealthy attack; specifically the one described in Section 4.2.2.

Algorithm 2 Online Safety Checking

```
INPUTS: (K, Π, x̂(k), ControllerState, UnsafeSet)
OUTPUT: true if the system is safe under a potential stealthy attack; false otherwise

x̂_p ← x̂(k)
for all l ∈ {0, 1, ..., K} do
    ReachEll ← Ellipsoid(x̂_p, Π);
    for all Hyperplane ⊂ UnsafeSet do
        DistToUnsafe ← dist(ReachEll, Hyperplane);
        if DistToUnsafe ≤ 0 then
            return false;
        end if
    end for
    x̂_p ← PredictControlFlow(x̂_p, ControllerState);
end for
return true;
```

4.4.1 Feasibility of a Stealthy Attack

In our physics-based EWS framework, we defined the *feasibility* ground for suspicion as the ability of a stealthy attack to drive the system into an unsafe state without getting detected. The proposed online safety checks provide an answer to whether a stealthy attack is feasible according to this definition, as we formally check in real-time whether the reachable set under an attack intersects the predefined set of unsafe states. Intuitively, the "larger" this intersection is, the more likely it is for an attacker to be able to cause damage to the system without getting detected. Based on this intuition, we can derive a certain "measure of feasibility" by considering the size of this intersection. Note that the total size of reachable sets has been used before in work on risk assessment (Esfahani et al. (2010); Murguia et al. (2018)) as a measure of impact of the concerned stealthy attack. In our work, instead of using the entire reachable set, we consider only the subset that is also contained within the set of unsafe states. Therefore, this can be interpreted as a measure of how feasible it is to cause damage through the attack from the point of view of the attacker; which is consistent with our definition of feasibility within the EWS framework.

Our use of ellipsoids to approximate the reachable set of states provides an efficient geometric approach to approximate its intersection with the set of unsafe states. Given the current estimated state $\hat{x}(k)$, the proposed online monitoring Algorithm 2 starts by predicting the evolution of this state for at most K time instants into the future. For a given time instant $k_f \in \{k, k+1, \ldots, k+K\}$, if $\mathcal{E}_x^p(k_f) \cap \mathcal{S}_u \neq \emptyset$, we can quantify the feasibility of a potential stealthy false data-injection attack using the approximate size of this intersection. Having $\mathcal{E}_x^p(k_f) \cap \mathcal{S}_u \neq \emptyset$ implies that there exists at least one half-space $\mathcal{H}_i \subset \mathcal{S}_u$ such that $\mathcal{H}_i \cap \mathcal{E}_x^p(k_f) \neq \emptyset$; in other words, it implies that at least one safety constraint is predicted to become potentially violated. Let $\mathcal{H}_v(k_f) \subseteq \mathcal{S}_u$ be the subset of potentially violated safety constraints at a time $k_f \in \{k, k+1, \ldots, k+K\}$:

$$\mathcal{H}_v(k_f) := \left\{ \bigcup_i \mathcal{H}_i \mid \mathcal{H}_i \cap \mathcal{E}_x^p(k_f) \neq \emptyset \right\} \tag{4.13}$$

For each half-space $\mathcal{H}_i \subseteq \mathcal{H}_v(k_f)$, it is possible to over-approximate $\mathcal{E}_x^p(k_f) \cap \mathcal{H}_i$ using a minimum-volume ellipsoid $\mathcal{E}_i(k_f)$ of centre $q_i(k_f) \in \mathbb{R}^n$ and shape matrix $\mathbf{\Pi}_i(k_f)$, as follows (Boyd et al. (2004); Dai and Pelckmans (2012)):

$$\begin{aligned} q_i(k_f) &= \hat{x}(k_f) - \frac{1+\alpha_i n}{n+1}\mathbf{\Pi}\bar{c}_i \\ \mathbf{\Pi}_i(k_f) &= \frac{n^2(1-\alpha_i^2)}{n^2-1} \times \left(\mathbf{\Pi} - \frac{2(1+\alpha_i n)}{(n+1)(\alpha_i+1)} \mathbf{\Pi}\bar{c}_i\bar{c}_i^T\mathbf{\Pi} \right) \end{aligned} \tag{4.14}$$

Where we have the following:

$$\begin{aligned} \bar{c}_i &:= C_{c,i}/(C_{c,i}\mathbf{\Pi}C_{c,i}^T)^{0.5} \\ \alpha_i &:= (C_{c,i}\hat{x}(k_f) - b_i)/(C_{c,i}\mathbf{\Pi}C_{c,i}^T)^{0.5} \end{aligned} \tag{4.15}$$

We quantify the feasibility of a potential stealthy FDIA on sensors using the volume of $\mathcal{E}_i(k_f)$. The volume of a general ellipsoid $\mathcal{E}$ in $\mathbb{R}^n$ with a shape matrix Q is given by (Kurzhanskiy and Varaiya (2006)):

$$\mathbf{vol}(\mathcal{E}) = \mathbf{vol}[\mathcal{B}_n]\sqrt{\det Q} \tag{4.16}$$

Where $\mathbf{vol}[\mathcal{B}_n]$ and $\det Q$ denote the volume of the unit n-ball and the determinant of the matrix Q, respectively. It is worthwhile to note that different system dimensions may lead to vastly different number ranges for the volume of the intersection ellipsoid. Thus, in order to make the feasibility metric more meaningful, we propose to use the ratio of the volume of the intersection ellipsoid to that of the ellipsoid approximating the reachable set. This guarantees that the feasibility metric will fall in the range $[0;1]$, thus becoming more intuitive to interpret. From (4.16), the real-time feasibility metric reduces to the following:

$$\mathbf{FEAS}(k) := \begin{cases} \sqrt{[\max_i \det \mathbf{\Pi}_i(k_f)]/\det \mathbf{\Pi}}, & \text{if } \exists k_f \in \{k, \dots, k+K\} \\ & \text{s.t. } \mathcal{E}_x^p(k_f) \cap \mathcal{S}_u \neq \emptyset \\ 0, & \text{otherwise} \end{cases} \tag{4.17}$$

4.4.2 Proximity/Approximate Time to Unsafe States

In the case $\mathcal{E}_x^p(k_f) \cap \mathcal{S}_u \neq \emptyset$ for some $k_f \in \{k, \dots, k+K\}$, we use the time instant k_f as an approximate *time to unsafe* states metric in our approach as follows:

$$\mathbf{Tc}(k) := \begin{cases} (k_f - k)\Delta_t, & \text{if } \exists k_f \in \{k, \dots, k+K\} \text{ s.t. } \mathcal{E}_x^p(k_f) \cap \mathcal{S}_u \neq \emptyset \\ -1, & \text{otherwise} \end{cases} \tag{4.18}$$

Where Δ_t is the system's sampling period. The advantage of using k_f instead of the distance from the state estimate $\hat{x}(k)$ itself is that the former approach accounts for the fact that if an undetected attack is present, then the actual state may be different from the estimated one. However, even if $x(k_f) \neq \hat{x}(k_f)$, $x(k_f)$ still lies within $\mathcal{E}_x^p(k_f)$ as otherwise, the attack would be detected. This metric provides operators with an idea of the minimum time the system would need to reach an unsafe state *if* it is subjected to a stealthy attack conforming to the threat model in Section 4.2.2.

As for the *proximity* ground for suspicion, we recall that according to the soundness principles we proposed for our EWS framework (Section 3.3), particularly principle (2), the suspicion metric should increase in value if the system appears to evolve closer to unsafe states in order to soundly reflect proximity.

Consequently, we propose to use the following as a measure of proximity:

$$\mathbf{PROX}(k) := \begin{cases} 1/(1+k_f-k), & \text{if } \exists k_f \in \{k,\dots,k+K\} \\ & \text{s.t. } \mathcal{E}_x^p(k_f) \cap \mathcal{S}_u \neq \emptyset \\ 0, & \text{otherwise} \end{cases} \tag{4.19}$$

Similarly to the case with feasibility, this formulation of the measure of proximity ensures that the latter falls in the same $[0;1]$ range as the feasibility measure, thus allowing for a more meaningful interpretation. Next, we combine the two measures of feasibility and proximity into a *suspicion metric* and we describe how it satisfies the soundness principles proposed in Section 3.3.

4.4.3 Suspicion Metric and its Soundness

We propose to combine the measures of feasibility and proximity (equations (4.17) and (4.19)) into a *suspicion metric* as follows:

$$\mathbf{SUSP}(k) := \mathbf{FEAS}(k) \times \mathbf{PROX}(k) \tag{4.20}$$

We note that since both terms of Equation (4.20) fall in the range $[0;1]$, then the suspicion metric also falls within the same range. To show how this metric can be used as a measure of the likelihood of damage taking place due to a stealthy attack, and as a way to raise early warnings and guide potential preemptive actions, we discuss its soundness in the context of the framework proposed in Section 3.3; particularly within the suspicion metric soundness principles

Soundness Principle 1. An increase in the value of the metric can imply one of the following: (i) the maximum volume of $\mathcal{E}_x^p(k_f) \cap \mathcal{S}_u$ is increasing, indicating that it is becoming increasingly feasible for the actual state of the system to diverge from the estimate and enter an unsafe operating region due to a stealthy attack; (ii) the term $k_f - k$ in (4.19) is decreasing, indicating that the system is in increasing proximity to unsafe operation regions. Thus, the proposed metric serves as a measure of likelihood of the attacker being able to take the system into unsafe states (feasibility) penalised by the number of time steps required to damage the system (proximity).

Soundness Principle 2. To raise physics-based early warnings, we propose two thresholds based on the value of $\mathbf{Tc}(k)$ (Equation 4.18). This value indicates the minimum time that would be needed by a potential attack to cause damage before being detected. Let t_1 be the time required by the operators to perform necessary preemptive actions after a low-criticality warning, and let t_2 be the time necessary to perform potentially more drastic actions after a high-criticality warning, with $t_2 < t_1$. t_1 and t_2 can be set based on expert knowledge of the system in question and the post-warning measures to be taken.

Accordingly, we set two conditions for different criticality thresholds in this work:

1. A "*low-criticality*" warning is raised when $t_2 < \Delta_t(k_f - k) \leq t_1$ and $\mathbf{FEAS}(k) \geq 0.5$. These conditions imply that if the system is under a potential stealthy attack, the damage will likely take place after at most t_1 time units. However, t_1 is long enough to perform low-criticality actions, such as collecting more data that may help explain whether an attack is taking place.
2. A "*high-criticality*" warning is raised if the suspicion metric shows that the attacker is likely to cause damage in a smaller time frame. In other words, this type of warning can be raised when $\Delta_t(k_f - k) \leq t_2$ and $\mathbf{FEAS}(k) \geq 0.5$ with $t_2 < t_1$. In this case, more urgent preemptive actions can be taken, such as engaging a potential fail-safe.

The above discussion shows that the proposed suspicion metric satisfies the soundness principles proposed in Section 3.3. Our distinction between low and high criticality warnings can help operators decide on appropriate post-warning measures. In the case of a low-criticality warning, the system is "far" from reaching unsafe states, even if it may be under a stealthy attack. Hence, operators can restrict post-warning actions to the collection of more data that can explain whether the system is indeed under attack. As opposed to a high-criticality warning, a low-criticality warning may not warrant more expensive actions, such engaging a fail-safe or a backup (potentially less efficient) controller (Chen and Sankaranarayanan (2017)). Consequently, we propose that a low-criticality warning, as defined previously, be used as a trigger for proactive data collection ((P1), Figure 3.1).

4.5 Algorithm Complexity

In this section, we make some remarks on the complexity of Algorithm 2. We also mention some results in ellipsoid calculus that can render some of the computations more efficient, and we propose an improved version of Algorithm 2 that includes an efficient computation of the suspicion metric.

In Algorithm 1, the prediction of the state of the system relies on an identified LTI model. As evident from Equation (4.1), the computation of the next state involves mainly vector addition and matrix-vector multiplication — operations that scale polynomially with the number of physical states. Thus, given a fixed number of physical states, the prediction is expected to scale linearly with the maximum number of time steps for prediction K.

Checking the emptiness of the intersection of the current reachable ellipsoid with the set of unsafe states relies on computing the distance between the two sets (Equation 4.12). While this computation relies on matrix-vector multiplication, thus scaling polynomially with the number of states and safety constraints, it can be made more efficient by considering the result by Dai and Pelckmans (2012). For a given half-space $\mathcal{H}_i = \{x \in \mathbb{R}^n \mid C_{c,i}x \geq b_i\} \subseteq \mathcal{S}_u$, recall from Equation (4.15), the variable $\alpha_i := (C_{c,i}\hat{x}(k_f) - b_i)/(C_{c,i}\mathbf{\Pi}C_{c,i}^T)^{0.5}$. It can be shown that if $|\alpha_i| > 1$, then $\mathcal{H}_i \cap \mathcal{E}_x^p(k_f) = \emptyset$ (Dai and Pelckmans (2012), Lemma 1). As a result, we can further reduce the intersection test to checking whether $|\alpha_i| > 1$ holds true for each $\mathcal{H}_i \subset \mathcal{S}_u$. Furthermore, the constant denominator term $(C_{c,i}\mathbf{\Pi}C_{c,i}^T)^{0.5}$ can be pre-computed offline to increase real-time performance.

For the computation of the intersection volume $\mathcal{E}_i = \mathcal{E}_x^p(k_f) \cap \mathcal{H}_i$, since we only use the ratio of this volume to that of $\mathcal{E}_x^p(k_f)$, we can also take advantage of a result by Dai and Pelckmans (2012) to avoid the computation of the potentially large matrix $\mathbf{\Pi}_i(k_f)$ and its determinant. Given an $0 \leq \alpha_i \leq 1$, we can simply compute the ratio of the two volumes as follows (Dai and Pelckmans (2012), Lemma 2):

$$\frac{\mathbf{vol}(\mathcal{E}_i)}{\mathbf{vol}(\mathcal{E}_x^p(k_i))} = \frac{n^n}{(1+n)(n^2-1)^{(n-1)/2}}(1-\alpha_i)(1-\alpha_i^2)^{(n-1)/2} \tag{4.21}$$

As a result, we would only need to compute α_i in order to obtain the measure of feasibility (Equation (4.17)), with the remaining terms in Equation 4.21 involving mere scalar operations.

Given this discussion, Algorithm 3 summarises the steps we employ to perform efficient online safety monitoring along with the computation of the suspicion metric.

4.6 Summary

In this chapter, we instantiated our framework for *Physics-based EWS* for the widely studied LTI systems, which forms the first component [**COMP1**] of our overall approach to supporting Forensic Readiness (FR) in ICS (Figure 3.1). We presented a predictive online safety monitoring approach for LTI systems under potential stealthy sensor attacks. Our approach pre-computes offline symbolic reachable sets in terms of the system's state estimate, by considering the evolution of the estimation error under a potential stealthy attack. Given the current state of the system and controllers, we predict in real time the control flow of the system for a certain umber of steps in the future. The pre-computed sets are then instantiated at the predicted estimates. We use ellipsoidal calculus techniques to perform emptiness checks of the intersection of the pre-computed set with a set of unsafe states.

We then proposed concrete measures of the feasibility and proximity preliminary indicators as part of our framework. For feasibility, we used the size of the intersection with the unsafe set; and for proximity, we used the time step at which we predict that such intersection will not be empty. We showed how existing results in ellipsoid calculus can enable efficient real-time computation of these measures. Finally, we combined these measures in a suspicion metric which satisfies the soundness principles that we proposed in our framework. We provided concrete thresholds for warnings of different criticality that can guide certain proactive actions. In particular, and to address part (P1) of the problem we tackle in this thesis, we specified a low-criticality threshold for the suspicion metric. Warnings based on this threshold imply that even if the system is under attack, the attacker has a low likelihood of causing damage before they are revealed by existing anomaly detectors, due to the system being "far" from unsafe

Algorithm 3 Online Suspicion Metric Computation

INPUTS: $(K, \mathbf{\Pi}, \hat{x}(k)$, ControllerState, UnsafeSet$)$

OUTPUT: Suspicion metric $\mathbf{SUSP}(k)$

$\hat{x}_p \leftarrow \hat{x}(k)$
for all $l \in \{0, 1, \ldots, K\}$ **do**
  ReachEll $\leftarrow$*Ellipsoid*$(\hat{x}_p$,$\mathbf{\Pi})$;
  for all Hyperplane $\subset$ UnsafeSet **do**
    $\alpha_i \leftarrow$ Equation (4.15);
    if $|\alpha_i| > 1$ **then**
▷ *If the instantiated ellipsoid intersects one of the "unsafe" hyperplanes, raise a "non-empty" intersection flag and compute the ratio of the intersection volume to the reachable set volume*
      isNonEmpty $\leftarrow$ *false*
      $\mathbf{FEAS}_i \leftarrow$ Equation (4.21)
    else
      $\mathbf{FEAS}_i \leftarrow 0$
    end if
  end for
  if !isNonEmpty **then**
▷ *When a non-empty intersection is found, stop prediction and return suspicion metric*
    $\mathbf{FEAS}(k) \leftarrow max_i(\mathbf{FEAS}_i)$
    $\mathbf{PROX}(k) \leftarrow 1/(1 + l)$
    $\mathbf{SUSP}(k) \leftarrow \mathbf{FEAS}(k) \times \mathbf{PROX}(k)$
    return $\mathbf{SUSP}(k)$
  end if
  $\hat{x}_p \leftarrow$ *PredictControlFlow*$(\hat{x}_p$,ControllerState$)$;
end for ▷ *If the time horizon is exhausted, the suspicion metric is zero.*
return 0;

states. Consequently, such a warning constitutes a suitable trigger for proactive data collection. While other safety measures that can stop an attack can be taken, we propose that such (potentially more expensive) actions be taken following a high-criticality warning, where a suspected attacker is more likely to cause damage.

In the next chapter, we address parts (P2) and (P3) of our problem. We extend the algorithms that we proposed in this chapter with automated techniques to identify data that is relevant to potential investigations, and to decide on collecting a subset of this data.

Chapter 5

Safety-Driven Collection of Potential Evidence of Stealthy Attacks

In the previous chapter, we described an instantiation of the framework for *Physics-based EWS* ([**COMP1**]) to Linear Time-Invariant (LTI) systems. We proposed an online safety monitoring algorithm to compute in real-time the probability that a potential stealthy attack is able to cause damage to the system (suspicion metric). In addition, we designed thresholds to raise warnings of different criticality based on this probability of damage. In particular, we proposed a threshold for *low-criticality* warnings, which imply that there is a low likelihood that an attacker would be able to cause damage without getting detected, and that there is sufficient time to collect more data before engaging potentially more expensive safety measures. We considered such a warning to be suitable trigger for proactive collection of potential evidence, hence addressing part (P1) of our problem (Figure 3.1).

In this chapter, we turn our focus to parts (P2) and (P3) of the ICS Forensic Readiness (FR) problem, and we describe our *Relevant Data Identifier* ([**COMP2**]) and *Relevant Data Selector* ([**COMP3**]) components (Figure 3.1) of our overall approach.

First, we address [**OBJ4**] and [**RQ2**] — *How can we identify data that is relevant to a potential investigation of a stealthy attack in ICS?*

Attempting to collect all data generated by an ICS at all times can be infeasible due to the large volume (Kiss et al. (2014)) of this data. Additionally, not all of this data may be *relevant* (Pasquale et al. (2018)) to potential forensic investigations. The stealthy attack that we consider in this thesis (Section 4.2.2) involves manipulations of data that exists at the control level of the ICS network architecture (Figure 2.2). Consequently, the data that originates from these devices, and in particular those at risk of being damaged by the considered stealthy attack, can be considered to be relevant in our context. Although data present at higher levels of the ICS network architecture (Figure 2.2) may also help explain how an attack may have occurred, we are primarily concerned in this work with data that is at risk of being lost damage due to originating from low-level devices located in close proximity to safety-critical equipment. We propose an automated technique to identify such relevant data in this chapter. Our technique takes advantage of the fact that the layout of geographically distributed, large-scale ICS is designed such that damage in a particular physical process has a low chance of spreading to other areas of the system. This layout design is known in the process industry as safe Process Plant Layout (PPL) (Quiroz-Pérez et al. (2021)). By mapping the safety constraints that we predict to be violated by a stealthy attack to the area(s) of the plant that may be damaged due to this violation, we can identify a subset of data that is likely to be lost as a result of such an attack. Using the set safety constraints and a layout of the ICS, as determined by process engineers, we extend Algorithm 4 with a mapping of the safety constraints that are predicted to be likely violated in the future, to a subset of relevant data. This technique forms the *Relevant Data Identifier* (**[COMP2]**) component of our overall approach.

Second, we address **[OBJ5]** and **[RQ3]** — *How can we select a subset of relevant data such that the control performance overhead resulting from real-time data collection is reduced?*

The proactive collection of data should account for storage and computational resources available in a given system (Pasquale et al. (2018)). This requirement is important in ICS where real-time performance should not be compromised as a result of data collection activities. Consequently, there is a need for a trade-off between the potential impact of a stealthy attack and the impact of real-time data collection on control operations. To achieve this trade-off and reduce the amount of data to be collected, we propose a decision-theoretic framework involving a *Value of Information* (VoI) analysis that decides on a subset of data to collect. Our framework quantifies the value of collecting more potential evidence in terms of the (expected) attack impact measured as the expected reduction in revenue from the ICS. We particularly assume that the collection of data could lead to the detection of a potential intrusion and eventually the prevention of damage from a potential attack. Thus, the "value" of collecting more data that could reveal such an intrusion can be quantified in terms of the impact of the suspected attack. Then, given a certain measure of data collection cost that depends on the type of data to collect, we simply decide whether certain data is "worth" collecting based on whether its collection cost exceeds the expected attack impact. This framework forms the final component of our approach — the *Relevant Data Selector* ([**COMP3**]).

We detail components [**COMP2**] and [**COMP3**] of our approach to proactive data collection in sections 5.1 and 5.2 respectively. We conclude this chapter with a summary in Section 5.3.

5.1 Identifying Relevant Data

In this section, we detail the *Relevant Data Identifier* ([**COMP2**]) component of our approach. Of the large volume of data generated by an ICS, we would like to only collect data that is likely to be lost due to damage caused by a potential stealthy attack. This data is likely to constitute potential evidence of such an attack, and thus is *relevant* (Pasquale et al. (2018)) to our proactive data collection activities.

In this thesis, since we are concerned with a stealthy attack (Section 4.2.2) that affects the physical process, we only focus on data originating from control devices that are at the lowest level of the ICS network architecture (Figure 2.2) (Eden et al. (2016); van der Knijff (2014)). In particular, we focus on Programmable Logic Controllers (PLC's) and on sensor measurements. We denote by $\mathcal{O} = \{o_1, \ldots, o_m\}$ the set of the system's output variables (e.g. reactor temperature and pressure for the Tennessee-Eastman Process (TEP)) and by

$PLC = \{PLC_1, \ldots, PLC_{n_{plc}}\}$ the set of PLC's in the system. Table 5.1 summarises the kind of data we can collect from these devices. For each PLC_i, we don't restrict our data collection to mere values of inputs or outputs, but we rather assume that data acquisition tools (e.g. as proposed by Janicke et al. (2015) and Yau and Chow (2015); Yau et al. (2018)) can also extract memory values and the ladder logic configuration installed on the PLC. We refer to this kind of data in an aggregate manner as the "configuration" of a PLC and we denote by $Config[PLC_i]$ the configuration that we can collect from PLC_i.

Note that the data that we propose to collect in real-time (Table 5.1) is different from the diagnostic data typically found in process historians, which may not be always useful in detecting potential intrusions or understanding how they occurred (Ahmed et al. (2017); Janicke et al. (2015)). We illustrated this difference in our Tennessee-Eastman Process (TEP) example in Section 3.2. The data that we consider is acquired using forensics tools that include information that may help a forensic investigator uncover potential intrusions. For example, the sensor measurements that we propose to collect may be already collected by a process historian. However, in our case, specific sensor measurements would be collected through network forensic tools that include information, for instance, about source and target IP addresses. Unlike the data that would be stored in process historians, this information may be helpful in understanding how a potential attack occurred — e.g. tracing IP addresses to an infected host machine. Similarly, for PLC data, raw values of control variables may be already collected by historians, but the forensic data acquisition tools would further include, for example, the ladder logic configuration currently installed. While raw values of control variables may help diagnose potential faults or anomalies arising from attacks, ladder logic configurations may help investigators understand whether and/or how a malicious code modification was performed.

5.1.1 Using the Geographical Layout of the ICS

Given that the *Physics-based EWS* ([**COMP1**]) component of our approach (Chapter 4) has identified the safety constraints that may be violated *if* the system is subjected to a stealthy attack, the *Relevant Data Identifier* ([**COMP2**])

Table 5.1: Potentially relevant data (in the context of the stealthy attack that we consider) at the control level of the Industrial Control System (ICS) network architecture and ways to collect it in real-time Eden et al. (2016); van der Knijff (2014).

Source	**Data**	**Ways to Collect in Real-time**
Network	Sensor measurements of operating variables in $\mathcal{O} = \{o_1, \ldots, o_m\}$	Network taps and widely available tools like *Wireshark* and *TCPdump*
PLC's	Memory values, states, ladder logic, inputs/outputs (we aggregate this data and denote it as $Config[PLC_i]$ for a given PLC_i)	The proprietary software of the PLC Janicke et al. (2015) and PLC data acquisition tools (e.g. Yau and Chow (2015))

identifies the subset of data that may be lost to the resulting damage — in other words, the data that we consider to be relevant in our context.

In a geographically distributed ICS, the layout of process equipment is designed with safety considerations in mind. Safety-critical process equipment, such as reactors, have an associated *explosion energy*, a measure of the energy that would be released should a safety failure take place within the equipment. This measure determines safe distances that separate different process equipment such that physical damage in one piece of equipment has a low probability of affecting another safety-critical physical process. These energy measures and associated distances are then used to design the layout of the process control system. The result is a partition of the land area available for the plant into different sub-areas where safety-critical equipment should be placed, in a similar fashion as shown in Figure 3.2 (Quiroz-Pérez et al. (2021)). This design-time activity is referred to in the process industry as safe Process Plant Layout (PPL) (Quiroz-Pérez et al. (2021)). To reduce the chance of damage spreading from one area into another, protection devices can be installed for that purpose (Penteado and Ciric (1996)). Note that such layout design does not only account for damage that could result from an explosion, but can also involve protection devices to prevent damages from incidents like chemical spills and fires.

In the TEP example (Figure 3.2), we assume that the equipment running the safety-critical processes is geographically distributed according to a safe PPL design. In this case, if damage occurs in the reactor for instance, then we assume that the damage will be restricted to the control devices located in the vicinity of the reactor (e.g. a PLC implementing the reactor's temperature control), and will not extend to devices elsewhere. For example, in the attack illustrated in Section 3.2 where a chemical overflow is induced in the reactor, a barrier wall around the area containing the reactor can lower the chance that potentially corrosive chemicals would reach devices in adjacent areas. Consequently, consider for instance the case where our physics-based EWS predicts that if the system is to be subjected to a stealthy attack conforming to the model described in Section 4.2.2, then there is a chance that the reactor level will exceed its safety limit. In this case, we would only consider data originating from devices in the vicinity of the reactor, as these are the only devices that are likely to be damaged.

To model the identification of relevant data in this sense, we can define the set of process sub-areas as a partition of the set of control devices PLC. Let[1] $\mathcal{A} = \{A_1, A_2, \ldots, A_{n_A}\} \subset 2^{PLC}$ be the set of the ICS sub-areas, then $A_i \neq \emptyset,\ A_i \cap A_j = \emptyset,\ \bigcup_i A_i = PLC;\ \forall i, j \in \{1, 2, \ldots, n_A\},\ i \neq j$.

In such ICS, safety constraints are usually expressed as linear combinations of output variables (e.g. Equation (4.2)). Let $\phi := \Sigma_{i=1}^{m} a_i o_i \Diamond b$ be a safety constraint, with $a_i, b \in \mathbb{R}$, $\Diamond \in \{\geq, \leq\}$, and $o_i \in \mathcal{O}$, and let Φ be the set of safety constraints in a given system. Damage in a sub-area A_i can happen due to a violated safety constraint $\phi_i \in \Phi$. We denote by $\mathrm{safe}(\phi_i) \in \mathcal{A}$ the plant area or the subset of control devices that will be affected by the violation of the safety constraint ϕ_i, and by $\mathrm{var}(\phi_i)$ the subset of output variables associated with ϕ_i. This is normally determined by process engineers at the PPL phase (Quiroz-Pérez et al. (2021)). Table 5.2 lists the main safety constraints of the TEP and associates each constraint with the plant area (Figure 3.2) where the corresponding safety-critical process is located. For example, constraint ϕ_3 sets an upper limit on the level of reactants inside the reactor. Since the reactor is located in area A_1, a violation of this constraint could lead to damaging the equipment in A_1; i.e. $\mathrm{safe}(\phi_3) = A_1$.

[1]Given a set A, 2^A denotes its power set.

Table 5.2: Safety constraints considered for the Tennessee-Eastman Process (TEP) with corresponding affected plant area(s) (Downs and Vogel (1993)).

Constraint Label	**Output Variable(s)** [**var**(ϕ_i)]	**Constraint Formula**	**Affected Plant Area** [**safe**(ϕ_i)]
ϕ_1	Reactor Pressure,o_1	$o_1 \leq 2895 kPa$	A_1
ϕ_2	Reactor Temperature o_2	$o_2 \leq 150°C$	A_1
ϕ_3	Reactor Level o_3	$o_3 \leq 21.3\ m^3$	A_1
ϕ_4	Reactor Level o_3	$o_3 \geq 11.8\ m^3$	A_1
ϕ_4	Product Separator Level o_4	$o_4 \leq 9.0\ m^3$	A_3
ϕ_5	Product Separator Level o_4	$o_4 \geq 9.0\ m^3$	A_3
ϕ_6	Stripper Base Level o_5	$o_5 \leq 6.6 m^3$	A_4
ϕ_7	Stripper Base Level o_5	$o_5 \geq 3.5 m^3$	A_4

Recall that the proposed data collection activity is initiated by a low-criticality warning at a time instant k from the *Physics-based EWS* ([**COMP1**], Figure 3.1) that return a subset $\Phi_v(k) \subseteq \Phi$ of safety constraints that may be violated in the future (Equation (4.13)). Consequently, given this subset of safety constraints, we can use Table 5.2 to determine a set of relevant device and network data (Table 5.1) at time k, respectively, as follows:

$$Config[PLC]_{rel}(k) = \{Config[PLC_i] \mid PLC_i \in \text{safe}(\phi) \forall\ \phi \in \Phi_v(k)\} \quad (5.1)$$

$$\mathcal{O}_{rel}(k) = \{o \in \text{var}(\phi)\ \forall\ \phi \in \Phi_v(k)\} \quad (5.2)$$

Hence, the set of relevant data that we are in a position to collect at a time k is $D_{rel}(k) = Config[PLC]_{rel}(k) \cup \mathcal{O}_{rel}(k)$.

5.1.2 Example

To illustrate this setup, consider again the case where the *Physics-based EWS* ([**COMP1**]) predicts, with a low-criticality warning at a time instant k, that if the TEP is subjected to a stealthy attack, then constraint ϕ_3, i.e. the upper limit on reactor level, may be violated.

Consider that area A_1 = safe(ϕ_3) (Table 5.2) contains the reactor, three PLC's implementing the pressure, temperature, and level control for the reactor — denote them by PLC_1, PLC_2 and PLC_3 respectively. Therefore, safe(ϕ_3) $= A_1 = \{PLC_1,\ PLC_2,\ PLC_3\}$.

Applying Equation (5.1), the relevant PLC configurations given a potential violation of ϕ_3 would be $Config[PLC]_{rel}(k) = \{Config[PLC_1], Config[PLC_2], Config[PLC_3]\}$. Additionally, the output variables associated with ϕ_3 are var(ϕ_3) $= \{o_3\}$ (Table 5.2). Applying Equation (5.2), $\mathcal{O}_{rel}(k) = \{o_3\}$. Finally, the resulting set of relevant data that we are in a position to collect at a time instant k given a low-criticality warning that ϕ_3 may be violated is $D_{rel}(k) = Config[PLC]_{rel}(k) \cup \mathcal{O}_{rel}(k) = \{Config[PLC_1], Config[PLC_2], Config[PLC_3], o_3\}$.

5.2 Selecting Relevant Data to Collect

Real-time data collection activities, especially the forensically-relevant data that we consider, can be associated with a certain cost. For example, PLC logging tools may influence the PLC performance, and sending such data to a secure server can consume network bandwidth that may be otherwise used to communicate control variables. Additionally, our original predictive safety checks are uncertain as they return a probability of damage, as given by the suspicion metric at a time instant k, which we denote in this chapter by $s(k)$. From this observation, it is necessary to decide whether to collect each data in $D_{rel}(k)$ based on a trade-off between the *expected* impact of the attack and the collection cost associated with each data in $D_{rel}(k)$. Accordingly, we seek to collect a subset of relevant data, $D_{col}(k) \subseteq D_{rel}(k)$. To achieve this trade-off, we propose a decision-theoretic framework involving a *Value of Information* (VoI) analysis. Our main assumption in the development of this framework is that each of the data identified in $D_{rel}(k)$ is equally likely to constitute potential evidence. For example, if $D_{rel}(k)$ consists of a pressure sensor measurement and a configuration of the PLC implementing the pressure controller, then we assume that it is equally likely that the attacker performed their attack by either changing the code of the PLC (Section 3.2) or intercepting and modifying the pressure sensor measurement. Thus, the only trade-off we are concerned with is the one between the expected impact of the attack and the cost of data collection. In the following, we detail this framework which forms the *Relevant Data Selector* (**[COMP3]**) of our approach.

5.2.1 Value of Information (VoI) Representation

We assume that the ICS, represented as $\Pi(\theta)$, operates in nominal conditions under a certain performance measured by a revenue $W(\theta)$ that depends on some system parameters grouped in θ. For example, the TEP's performance is usually measured by the revenue resulting from the amount of chemical product produced (Downs and Vogel (1993); Ricker (1996)). The operational cost of $\Pi(\theta)$ is denoted by $C(\theta)$, and can be estimated based on a variety of parameters; such as the amount of reactants consumed in the TEP case. In this thesis, we restrict this cost to the cost of data collection, i.e., collecting the data in the set $D_{rel}(k)$.

Under a given parameter setting θ, the value (profit) of $\Pi(\theta)$ is given by $V(\theta) = W(\theta) - C(\theta)$. However, due to uncertainties associated with parameters θ, we must use an *expected* parameter estimate to compute $V(\theta)$. Consequently, we consider the expected value (profit), denoted by $\mathbb{E}[V(\theta)]$, under expected parameters $\mathbb{E}[\theta]$.

Suppose that we predict that a potential stealthy attack, may bring the system into a new operating mode with $E[V(\tilde{\theta})] < E[V(\theta)]$, by causing damage to one or more of the plant areas A_i (e.g. we are presented with the low-criticality warning). Then we can ask whether it is worthwhile to collect certain relevant data (as identified in $D_{rel}(k)$) that may allow us to reveal a breach like the one illustrated in Section 3.2. Namely, we may want to collect a given $\delta_i \in D_{rel}(k)$ only if its collection cost does not exceed the potential reduction in the value or the revenue of the system's operation (i.e. attack impact) if the suspected stealthy attack is successful. The change in this expected value is given by:

$$\begin{aligned} \Delta_V(\tilde{\theta}, \theta) &= \mathbb{E}[V(\tilde{\theta})] - \mathbb{E}[V(\theta)] \\ &= \mathbb{E}[W(\tilde{\theta}) - C(\tilde{\theta})] - \mathbb{E}[W(\theta) - C(\theta)] \end{aligned} \tag{5.3}$$

As a simplifying assumption, we can take the cost of data collection under nominal conditions $C[\theta]$ as 0. On the one hand, if we choose not to collect any data and subsequently not attempt to prevent the damage from the breach, then $C[\tilde{\theta}] = 0$ and in this case $|\Delta_V(\tilde{\theta}, \theta)| = |\mathbb{E}[W(\tilde{\theta}) - W(\theta)]| = \mathbb{E}[|\Delta_W|]$. On the other hand, if we choose to collect data that can help us identify the location of the breach and prevent damage from happening, then we can assume that $W(\tilde{\theta}) = W(\theta)$ and $|\Delta_V(\tilde{\theta}, \theta)| = \mathbb{E}[C(\tilde{\theta})]$. In other words, the collection of data that may reveal a suspected breach is worthwhile only if the expected cost of collection does not exceed the expected reduction in the operating revenue (performance), $\mathbb{E}[|\Delta_W|]$, i.e. the expected impact of the attack.

5.2.2 Example

Consider the TEP system generating a value of 0.8 under nominal operation. Our online safety monitor predicts at a time instant k that a potential breach may be able to cause damage to two out of the five plant areas in $\mathcal{A}$ with a probability (given by the suspicion metric) of $s(k) = P(\text{damage}) = 0.4$. If this happens, then the new revenue of operation is 0.48. As a result, the new expected revenue of operation is given by:

$$\begin{aligned} \mathbb{E}[\tilde{W}] &= P(\text{damage})(W|\text{damage}) + P(\neg\text{damage})(W|\neg\text{damage}) \\ &= 0.4 \times 0.48 + 0.6 \times 0.8 \\ &= 0.612 \end{aligned} \tag{5.4}$$

Hence, the expected reduction in revenue of operation (i.e. attack impact) is $\mathbb{E}[|\Delta_W|] = 0.8 - 0.612 = 0.188$. This is the maximum we will "pay" to identify the location of the suspected breach, i.e. collect each of the data in $D_{rel}(k)$.

5.2.3 Cost of Data Collection

In the set $D_{rel}(k)$ that we identified in Section 5.1, the collection of each of the data $\delta_i \in D_{rel}(k)$ is associated with a certain cost $C(\delta_i)$. This measure of collection cost can be estimated, for example, based on the capabilities of the existing data acquisition tools. If the collected data is to be stored in some secure server, one could set a limit on the total size of data that we can collect at a given time instant k. Then, we can define a collection cost $C(\delta_i)$ as the ratio of the space occupied by δ_i to the size limit for data collection at a time

Algorithm 4 Proactive Real-time Collection of Relevant Data

INPUTS: A subset of potentially violated constraints $\Phi_v(k)$; the suspicion metric $s(k) = P(\text{damage})$

PARAMETERS: collection cost of potential data types $C(\delta_i)$, ICS operating value under nominal conditions $(W|\neg\text{damage})$

OUTPUTS: $D_{\text{col}}(k)$ (a set of data to collect)

for all $\phi \in \Phi_v(k)$ **do**
    $Config[PLC]_{rel}(k) \leftarrow Config[PLC]_{rel}(k) \cup \{Config[PLC_i] \mid PLC_i \in \text{safe}(\phi)\}$
    $\mathcal{O}(k)_{rel} \leftarrow \mathcal{O}_{rel}(k) \cup \text{var}(\phi)$
end for
$D_{rel}(k) \leftarrow \mathcal{O}_{rel}(k) \cup Config[PLC]_{rel}(k)$
$\mathbb{E}[\tilde{W}] \leftarrow s(k)(W|\text{damage}) + (1 - s(k))(W|\neg\text{damage})$
$\text{MaxCollectionCost} \leftarrow (W|\neg B) - \mathbb{E}[\tilde{W}]$
for all $\delta_{rel} \in D_{rel}(k)$ **do**
    if $C(\delta_{rel}) < \text{MaxCollectionCost}$ **then**
        $D_{col}(k) \leftarrow D_{col}(k) \cup \{\delta_{rel}\}$
    end if
end for

k. Regardless of the exact definition of $C(\delta_i)$, we propose that this measure be a dimensionless number in the same range as the measure of revenue of operation $(W(\theta))$, so that we can soundly perform the trade-off between the two measures.

Going back to the previous numerical example, assume that the identified relevant data consists of the set in the example presented in Section 5.1.2. Since a single sensor measurement may occupy significantly less space that a PLC ladder logic configuration, we assume that the cost of collecting o_3 is $C(o_3) = 0.02$ and that of $Config[PLC_1]$ is $C[Config[PLC_1]] = 0.2$. Since the maximum we will pay to identify the location of the suspected breach is 0.188, then we can conclude that o_3 is worth collecting while $Config[PLC_1]$ is not.

5.2.4 Real-Time Proactive Data Collection

We implement the steps of our approach proposed to identify ([**COMP2**]) and select ([**COMP3**]) relevant data to collect in Algorithm 4. The execution of Algorithm 4 at a time instant k is triggered by a low-criticality warning from the *Physics-based EWS* ([**COMP1**]) component (Chapter 4, Algorithm 3) which returns the suspicion metric $s(k)$, i.e. probability of damage taking place, in addition to the subset of safety constraints $\Phi_v(k)$ that may be violated in the future. We first use the safe PPL and Table 5.2 to identify data that is more likely to be lost due to the predicted damage ([**COMP2**]). We then perform the VoI analysis to select a subset of the identified relevant data in $D_{rel}(k)$ ([**COMP3**]). First, we estimate the expected reduction in revenue (impact) from the predicted attack, as illustrated in the previous example. Then, we use this computed impact as the maximum we will pay to collect a certain $\delta_{rel} \in D_{rel}(k)$. We finally make our collection decision given an associated $C(\delta_{rel})$.

Note that the estimation of the revenue if damage happens, $(W|\text{damage})$, can be performed using the number of areas A_i that are predicted to be damaged by the online safety monitor. In our numerical example, the online safety monitoring algorithm predicted that two out of five areas in $\mathcal{A}$ may be damaged. Hence, we can estimate a $2/5 = 40\%$ reduction in the original revenue under nominal operation. Note that a more accurate estimate can be obtained by incorporating some process engineering knowledge about the contribution of each plant area to the revenue (Peters et al. (2003)); however, such considerations are beyond the scope of the present work.

5.3 Summary

In this chapter, we turned our focus to parts (P2) and (P3) of the ICS FR problem, and we described the *Relevant Data Identifier* ([**COMP2**]) and *Relevant Data Selector* ([**COMP3**]) components of our approach.

First, for our *Relevant Data Identifier*, we proposed an automated technique that can specify data that is relevant to potential investigations of a stealthy attack on ICS following the threat model described in Section 4.2.2. In particular, we considered in our context that data originating from components at risk of being damaged is relevant. To identify this data, we mapped the potentially violated safety constraints given by our *Physics-based EWS* ([**COMP1**]), to specific areas of the plant where damage can happen due to such violations. We made use of the safe PPL to identify these areas.

Second, for our *Relevant Data Selector*, we proposed a decision-theoretic framework that can select a subset of relevant to collect, by achieving a trade-off between the expected attack impact and the potential cost of collecting more data. We characterised the impact of a suspected attack as a reduction in the revenue generated by an ICS. This reduction in revenue sets a limit on how much we are willing to "pay" for more data that may otherwise reveal an intrusion. By having a measure of data collection cost at hand, which is dependent on the type of data and the capabilities of data acquisition tools, we can decide for each relevant data whether it is "worth" collecting. Namely, we select data whose collection cost does not exceed the potential attack impact.

Having addressed parts (P1, P2, P3) of our problem, we perform in the next chapter a thorough evaluation of the three components of our work.

Chapter 6

Evaluation

In this chapter, we evaluate different aspects of the approach described in this thesis using a virtual testbed consisting of a simulation of the benchmark Tennessee-Eastman Process (TEP). Initially, we focus on the *Physics-based EWS* (**[COMP1]**) component of our approach (Figure 3.1). We first evaluate the accuracy of our online safety monitoring algorithm for LTI systems. We check whether the proposed algorithm can indeed predict potential safety violations if the system is under a stealthy attack. We perform these assessments through extensive Monte Carlo simulations under stealthy attacks, where we either vary the number of sensors being attacked simultaneously, or we vary the length of the prediction horizon.

Second, we compare our reachability-based approach to safety monitoring to existing work in this area. In particular, we show that the use of reachability tools instead of relying on mere proximity metrics that are based on raw sensor measurements is well justified when the system is under a stealthy attack. We perform these comparisons through a set of attack scenarios.

Third, we evaluate whether our setup of the suspicion metric, with the suggested thresholds, can provide meaningful warnings to operators. Using attack scenarios, we illustrate how the criticality (i.e. low and high) of the fired warnings reflects how fast the system is steered into an unsafe state by a stealthy attacker. We particularly show that the low-criticality warning is a suitable trigger for proactive data collection.

Fourth, we assess the real-time performance of the proposed *Physics-based EWS*. Through extensive Monte Carlo simulations, where we either vary the number of safety constraints or the length of the prediction horizon, we evaluate the execution time of the algorithm and we assess how it scales with each of these parameters. In addition, by simulating network behaviour, we also assess the effect of the algorithm on communication delays between controllers, sensors, and actuators.

Finally, we evaluate the relevance of the data identified by the *Relevant Data Identifier* ([**COMP2**]) and the effect of the *Relevant Data Selector* ([**COMP3**]) on the control performance overhead resulting from real-time data collection. We show that [**COMP2**] rarely misses any relevant data (i.e. data that is at risk of being lost to damage). We also compare [**COMP3**] with an agnostic "all-data" collection approach and demonstrate the reduction in performance overhead on the system's controllers as a result of our approach. We evaluate these aspects through additional Monte Carlo simulations of the TEP under a stealthy attack. Additionally, we demonstrate a use case for our proactive data collection approach where we show that it can improve the efficiency of PLC log-based live forensics tools. We demonstrate this use case by simulating a live forensics activity in TEP under a Stuxnet-inspired attack scenario similar to the one illustrated in Section 3.2. We compare the case where this simulation includes an implementation of our approach with the case where all data in an ICS is collected at all times.

All activities conducted to support the evaluation were performed on a machine with an Intel i7–9750H CPU clocked at 2.6 GHz with 16GB of RAM memory. The rest of this chapter is organised as follows: we begin with a detailed description of the virtual testbed that we employ for our evaluation in Section 6.1. We then evaluate the accuracy of the online safety monitoring algorithm and compare it with existing techniques in Section 6.2. We evaluate the ability of the suspicion metric to warn well in advance of potential damage and to guide post-warning actions in Section 6.3. We evaluate the real-time performance of the *Physics-based EWS* ([**COMP1**]) in Section 6.4. We finally evaluate the relevance of the data identified by the *Relevant Data Identifier* ([**COMP2**]), the reduction in control performance overhead resulting from *Relevant Data Selector*

([**COMP3**]), and we present a use case for our approach in Section 6.5. We conclude this chapter with Section 6.6.

6.1 Virtual Testbed Description

To evaluate the work in this thesis, we rely on a simulation of the Tennessee-Eastman Process (TEP) (Downs and Vogel (1993)). This is a benchmark chemical process that has been used extensively to study problems in the process control field and is widely regarded as one that reflects to a high degree of accuracy a real-life chemical process (Ricker (1993)). The process involves an exothermic reactor and units to separate and purify chemical products. Several control loops maintain critical operating variables, such as reactor temperature and pressure, at their desired set-points (Table 5.2), (Ricker (1996)). In the context of ICS security, the complexity of this process has allowed the realistic simulation of physical processes under attack and to test model-based approaches to ICS security (Cárdenas et al. (2011); Geng et al. (2019); Genge and Siaterlis (2014); Krotofil and Cárdenas (2013); McEvoy and Wolthusen (2011*b*)). In addition, the simulation-based TEP allows for a low-cost and safe testing of the effect of potentially damaging attacks on physical operation.

We modified the MATLAB/Simulink simulation of the TEP built by Bathelt et al. (2015) by adding blocks to simulate the real-time behaviour of the control network, sensors, actuators, and controllers. We also implemented a Kalman filter-based anomaly detector to estimate process measurements and detect anomalies. We derived the Kalman filter for the TEP process using MATLAB's built-in `kalman` Simulink block. A diagram showing the structure of our vritual testbed is shown in Figure 6.1.

The network is divided into four segments connected by a "gateway" router to emulate the distributed nature of modern ICS environments. In the control rooms, where different stages of the chemical process reside, sensors send measurements over the first network segment to the gateway, which forwards them to the appropriate node on the controllers' network. The controllers employ a similar procedure to send control signals to the appropriate actuator nodes. The gateway forwards sensor measurements and controller states to the supervisory

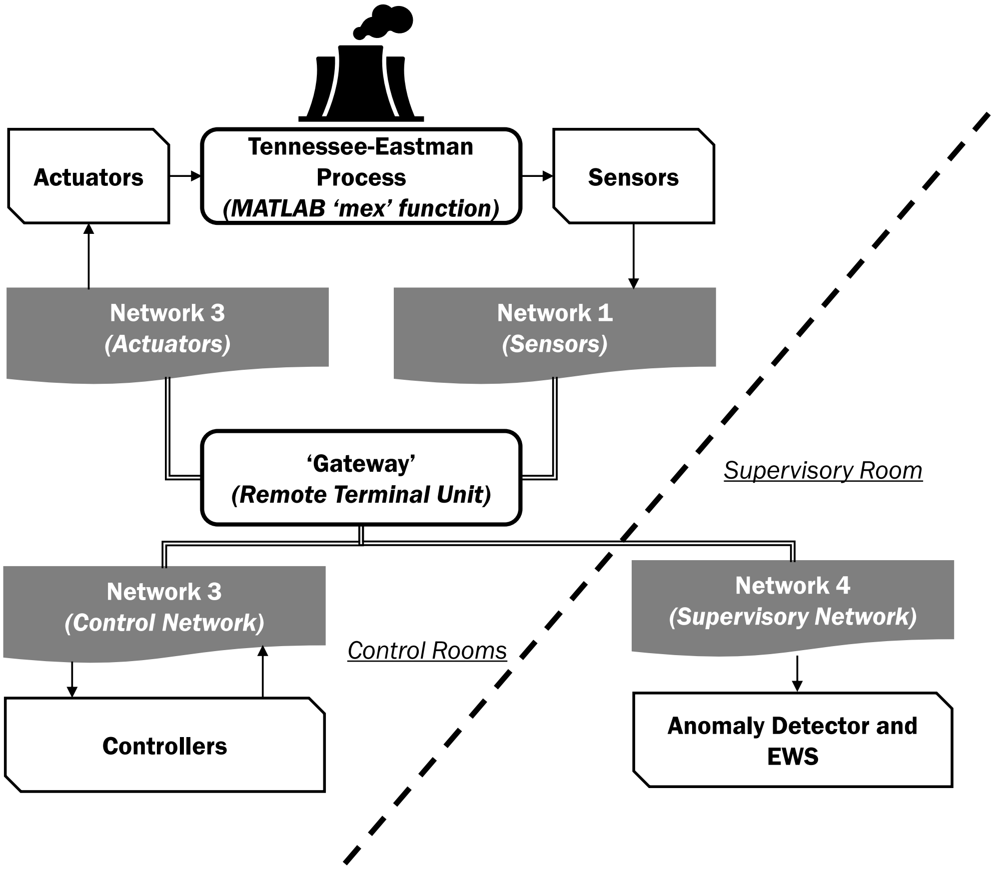


Figure 6.1: Networked Tennessee-Eastman Process (TEP) testbed diagram.

control room where the anomaly detector and the proposed instantiated framework (indicated as EWS in Figure 6.1) reside. The gateway then emulates Remote Terminal Units (RTU's) which are used to provide an interface between control devices and servers in control rooms.

To simulate the real-time behaviour of sensors, controllers, anomaly detector, and the network, we use the MATLAB/Simulink-based TrueTime library (Cervin et al. (2003)). This library is a collection of MATLAB/C++ functions and Simulink blocks designed to simulate real-time control networks. It has been primarily adopted to study the performance of networked control systems (Brahimi et al. (2006); Kalaivani and Kalaiarasi (2019)), and also in the context of ICS security (Farooqui et al. (2014)). The TrueTime library provides

Simulink blocks to simulate medium access and packet transmission for different industrial network models, such as CAN, Round-Robin, PROFINET, etc. It also provides "kernel blocks" for which custom MATLAB or C++ code can be implemented to simulate different nodes (e.g. actuators, sensors, controllers etc.) with specified scheduling policies. The TrueTime library simulates medium-access and packet-level network protocols, which is a sufficient simulation level to study the overhead incurred by the different components of our approach.

We compiled the physical behaviour of the TEP implemented in C by Bathelt et al. (2015) into a MATLAB "`mex` S-function" to be incorporated in the Simulink-based simulation. We implemented sensor, actuator, and controller codes as TrueTime Kernels, representing the real-time behaviour of control devices with a fixed-priority scheduling policy. While the controllers are implemented as Simulink blocks, the physical process itself is simulated in continuous-time and incorporated into Simulink using MATLAB's S-function blocks. In our implementation of sensor and actuator nodes using the Truetime library, we simulated the periodic discrete-time sampling of sensor measurements and actuator signals; in a fashion that mimics real-life situations in process control systems.

The Truetime-simulated control network employs a Carrier Sense Multiple Access with Arbitration on Message Priority (CSMA/AMP) model, which is widely used in industrial Controller Area Network (CAN) bus applications (Gupta and Chow (2009)). The TrueTime library assumes that higher-level network protocols process messages into packets.

6.2 Online Safety Monitoring Algorithm

In this section, we focus on evaluating the accuracy of our proposed online safety monitoring algorithm for stealthy attacks on ICS that can be modelled as Linear Time-Invariant (LTI) systems (Algorithm 2). As this algorithm forms the core of the *Physics-based EWS* ([**COMP1**]) component, we focus on the evaluation of its accuracy in this section. We implemented Algorithms 1 and 2 in MATLAB, with the ellipsoidal techniques based on the Ellipsoidal Toolbox written by Kurzhanskiy and Varaiya (2006). We approximated the TEP process as an

Table 6.1: Safety constraints considered for the Tennessee-Eastman Process (TEP) (Downs and Vogel (1993)).

Output	**Low Limit**	**High Limit**
Reactor Pressure	none	2895 kPa
Reactor Temperature	none	150 $°C$
Reactor Level	11.8 m^3	21.3 m^3
Product Separator Level	3.3 m^3	9.0 m^3
Stripper Base Level	3.5 m^3	6.6 m^3

LTI system with 50 state variables using MATLAB's `n4sid` algorithm. We considered the safety constraints discussed by Downs and Vogel (1993), shown here in Table 6.1. To initialise the online monitoring tool, we used the Kalman filter and chi-squared anomaly detector parameters to run Algorithm 1 and determine the reachable ellipsoid's shape matrix. We performed the grid search for parameter b by dividing the interval into segments of length 0.01.

Our evaluation of the online safety monitoring algorithm for LTI systems under stealthy attacks consists of three main parts. First, we validated our approach by assessing the accuracy of the safety checks using extensive Monte Carlo simulations. Second, we compared our approach with existing online monitoring approaches. Finally, we assessed the real-time performance of the algorithm. We deferred this part of the evaluation however to Section 6.4 following our discussion of the suspicion metric, as we would like our performance evaluation to account for the computation of this metric as well. Therefore, this section only presents the results of our accuracy tests and the comparison of Algorithm 2 with existing techniques.

6.2.1 Accuracy

The objective of Algorithm 2 is not to detect attacks, but rather to perform safety checking under *potential* stealthy attacks that seek to cause damage. Algorithm 2 checks whether the current state of the system can be taken to an unsafe state by a stealthy attack within the next K time instants and cause damage *before* the anomaly detector detects the attack. Thus, to evaluate our approach, we ran several simulations of the TEP only considering different stealthy attacks on the sensors that report values of safety critical parameters shown in Table 6.1. We avoided using the true/false positive/negative rate performance metrics as traditionally defined in the attack detection literature. Instead, we considered a notion of true/false positives/negatives similar to the one adopted in previous work involving reachability analysis for safety verification, specifically by Chou et al. (2020):

- A *true positive* occurs when Algorithm 2 predicts a potential safety violation within K time instants before damage occurs due to an attack, and the system reaches an unsafe state before the anomaly detector raises an alarm.
- A *true negative* occurs when Algorithm 2 does not predict any potential safety violation within K time instants before damage occurs, but the attack is detected by the anomaly detector before it causes damage.
- A *false positive* occurs when Algorithm 2 predicts a potential safety violation within K time instants before damage occurs, but the anomaly detector manages to detect the attack before the system is damaged.
- A *false negative* occurs when Algorithm 2 does not predict any potential safety violation within K time instants before damage occurs, even though the stealthy attack manages to cause damage, and the anomaly detector fails to raise any alarm before such damage occurs.

To evaluate the true/false positive/negative rates for Algorithm 2, as defined previously, our Monte Carlo simulations considered varying two parameters: (i) the length of the prediction horizon K, and (ii) the number of safety-critical sensors attacked at the same time. For the length of the prediction horizon K, the results are presented in Figure 6.2. For each value of K that we tested, we ran 500 simulations where we picked the attacked sensors at random (uniform distribution), and we simulated the attack as a slowly growing bias on sensor measurements.

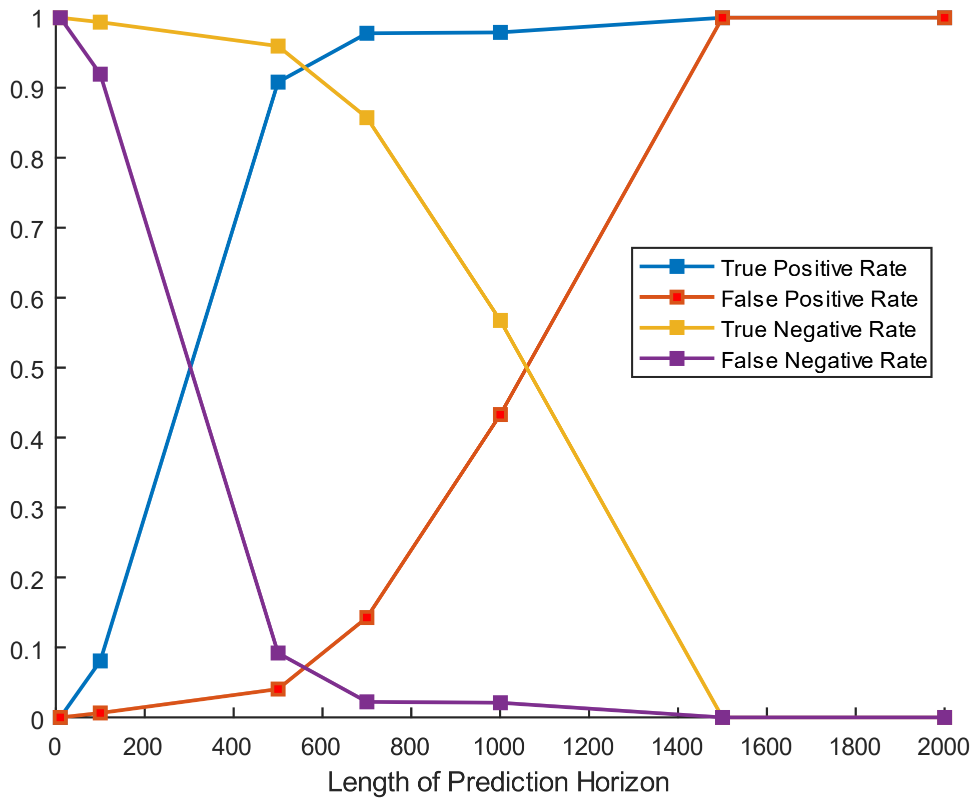


Figure 6.2: True/false positive/negative rates for Algorithm 2 as a function of the length of prediction horizon K.

We can see from Figure 6.2 that for a small prediction horizon length, Algorithm 2, returns mostly negative checks, with true and false negatives accounting for the vast majority of predictions for $K < 500$. As K grows, the number of true and false positives increases, with the false positive rate increasing in a much slower manner. For $K \geq 1500$, although the rate of true positives is high, Algorithm 2 returns a high number of false positives as well. This behaviour is the result of the design of Algorithm 2. First, for small K, the algorithm will likely not be exploring a sufficient number of states where a stealthy attack would cause a violation of safety constraints. Thus, it is expected to observe a high rate of both false and true negatives, with true and false positive rates remaining very low. As K increases, the algorithm is allowed to explore more states, therefore increasing the number of true positives. The slow parallel increase in false posi-

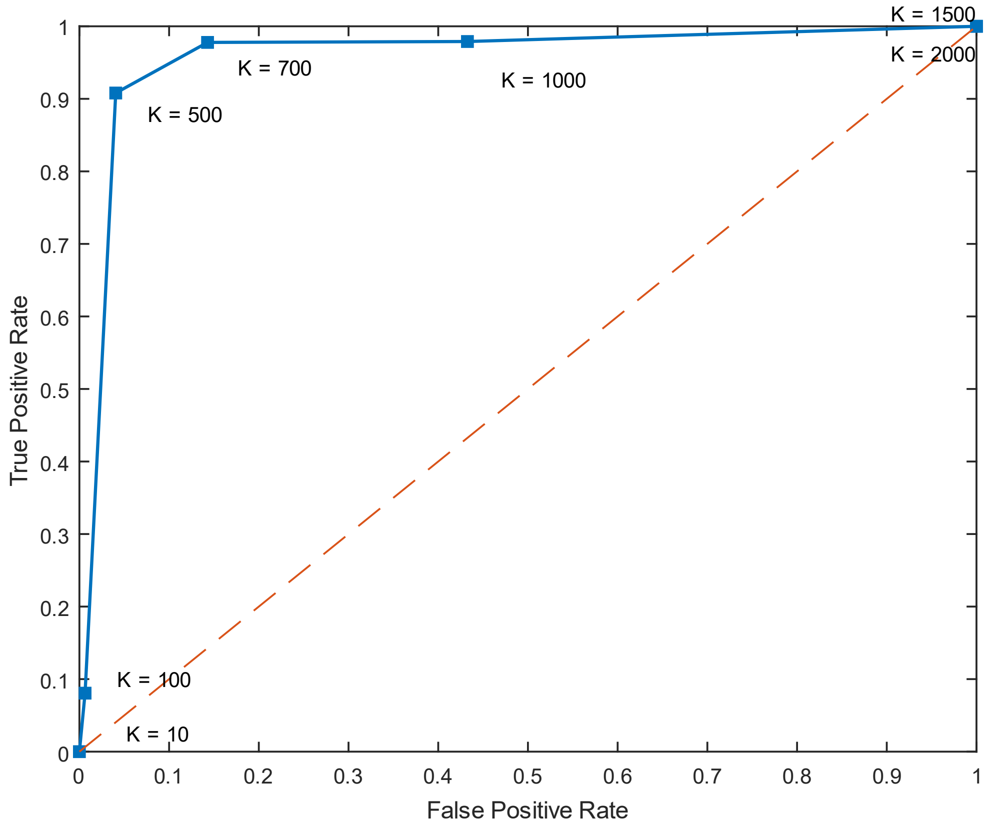


Figure 6.3: Receiver Operating Characteristic (ROC) curve for Algorithm 2 with the length of prediction horizon K as the third dimension.

tives shows that Algorithm 2 exhibits high accuracy for intermediate values of K. However, at high values of K, the accuracy of the predicted states is expected to decrease, which explains the high false positive rates.

These simulations show that there exists a trade-off between how early we would like to raise warnings about potential safety violations due to a stealthy attack and the accuracy of Algorithm 2. These experiments can also serve as a method to tune the choice of K. To showcase these ideas, we have plotted in Figure 6.3 the Receiver Operating Curve (ROC) for Algorithm 2, with the length of prediction horizon K as the third dimension. We note that there exists a "cut-off" point at $K = 500$ time steps where we obtain acceptable values for the true/false positive rates (90.8% for true positives, 4.05% for false positives). This

Table 6.2: True/false positive rates vs. the number of sensors attacked at the same time for Algorithm 2

Number of Attacked Sensors	**1**	**2**	**3**	**4**	**5**
True Positive Rate	0.908	0.905	0.91	0.905	0.92
False Positive Rate	0.0405	0.041	0.04	0.0408	0.04

is equivalent to about 15 minutes ahead-of-time prediction, which is a reasonable choice in practice for K.

For $K = 500$, we tested the accuracy of Algorithm 2 under different numbers of sensors being attacked at the same time. We ran 500 simulations for each different number of sensors being attacked. In each simulation, we picked the attacked sensors at random (uniform distribution), and we ran Algorithm 2 while considering the safety constraints associated with the sensor(s) under attack (Table 6.1). The results in Table 6.2 show high true positive and low false positive rates in each case. These experiments demonstrate the accuracy of Algorithm 2 with respect to all safety-critical sensors. Given the large number of random simulations we ran, we can conclude that Algorithm 2 can report potential safety violations due to a stealthy attack with respect to all the safety constraints imposed on the system.

6.2.2 Comparison with Existing Monitoring Approaches

In this section, we empirically showcase the advantage of using Algorithm 2 for online monitoring compared to the approach adopted in the works discussed in Section 4.1.1. We specifically show that relying on a mere proximity metric based on raw sensor measurements is ineffective against stealthy attacks that can make the system appear to be moving away from unsafe states. However, our approach, which considers reachable sets under attacks, can effectively warn about potential safety violations even in such scenarios.

Our comparison with existing techniques does not extend to work focused on attack detection, as our algorithm is designed for safety monitoring and not for attack detection; as evident from our previous definitions of accuracy metrics. To this end, we implemented an online monitoring tool measuring the time-to-unsafe states metric based on a simple euclidean distance from the current state estimate to the set of unsafe states. To obtain this traditional measure of time-to-unsafe states, we divide the euclidean distance by the average rate of change of the estimated system state. We avoided a comparison based on the accuracy metrics depicted in the previous section. This is due to the fact that the traditional time-to-unsafe states metric relies on the selection of different thresholds to raise alarms of different criticality. With the lack of precise methods to select these thresholds, it is hard to perform a meaningful quantitative comparison between the metric proposed in this paper and the traditional one. Therefore, we used a set of attack scenarios involving safety-critical sensors to empirically demonstrate the advantage of our approach to online safety monitoring.

We chose three attack scenarios. Scenarios 1 and 2 depict individual attacks targeting the level and pressure sensors, respectively, of the main reactor in the TEP. Scenario 3 depicts an attack performed simultaneously on the level, pressure, and temperature sensors of the reactor. We chose Scenarios 1 and 2 to illustrate the typical kind of attacks targeting safety-critical sensors individually. We can obtain similar results for individual attacks on other safety-critical sensors. Scenario 3 illustrates a more dangerous coordinated attack where all main reactor sensors in the TEP are manipulated at the same time. Again, similar results can be obtained for other combination of sensors for safety-critical variables listed in Table 6.1. Figures 6.4, 6.5 and 6.6 present the results obtained for each scenario. In each of these plots, we show the feasibility metric (Equation (4.17)) and the time-to-unsafe metric (Equation (4.18)). We show the former to indicate when Algorithm 2 returned with an unsafe check (i.e. non-zero feasibility metric, safe check otherwise) and the latter in order to compare our approach with techniques that employ a traditional time-to-unsafe states.

Scenario 1. In this scenario, we simulate an attack on the reactor's level sensor, where a growing bias on level measurements is introduced to trick the controller into overflowing the reactor. This simulation is shown in Figure 6.4 where we can see that the anomaly detector raises an alarm almost at the moment

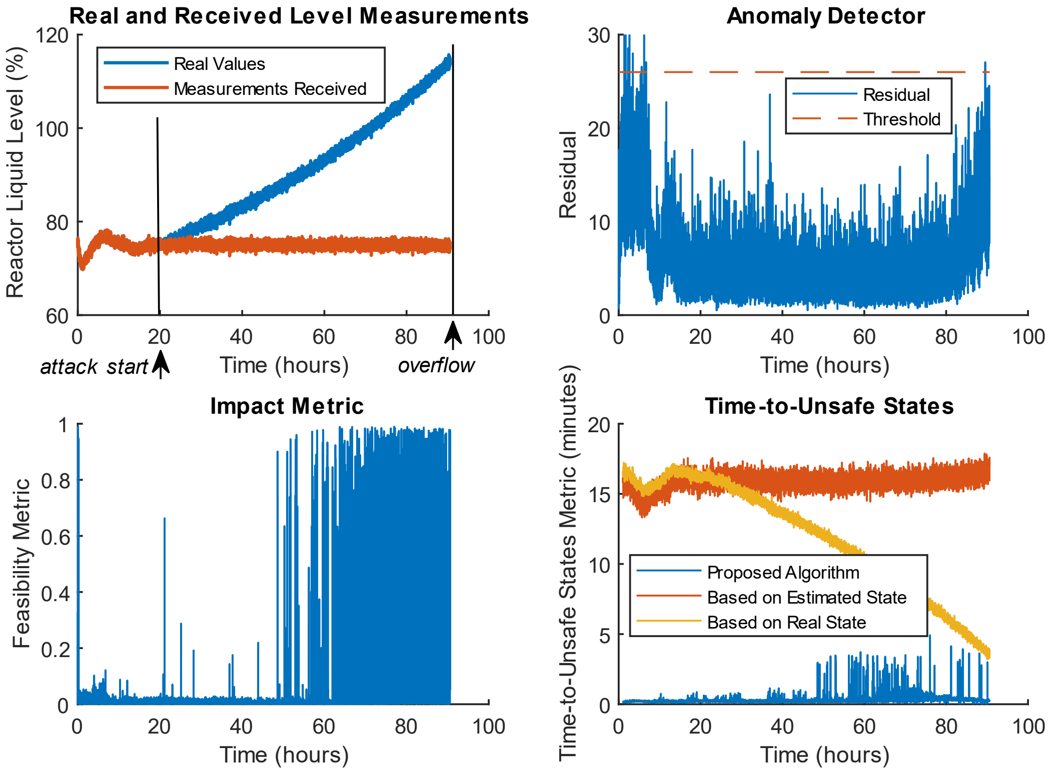


Figure 6.4: Results from simulation Scenario 1.

the overflow takes place. Our feasibility metric increases significantly over the period preceding the physical damage to the reactor, and our time-to-unsafe states metric shows that the plant is dangerously close to damage. Conversely, the traditional time-to-unsafe states computed based on the estimated state alone shows a slight increase, indicating that the plant appears to be moving away from the unsafe operating region.

Scenario 2. In this scenario, we perform an attack on the reactor's pressure sensor. The attack is a slowly growing bias on pressure measurements seeking to trick the pressure controller into increasing the pressure inside the reactor while it appears lower than its set-point. We can see from Figure 6.5 that the anomaly detector fails to raise any alarm before excessive pressure builds up in the reactor. However, our online monitoring algorithm correctly predicts the damage from the stealthy attack as shown by a non-zero feasibility metric. This

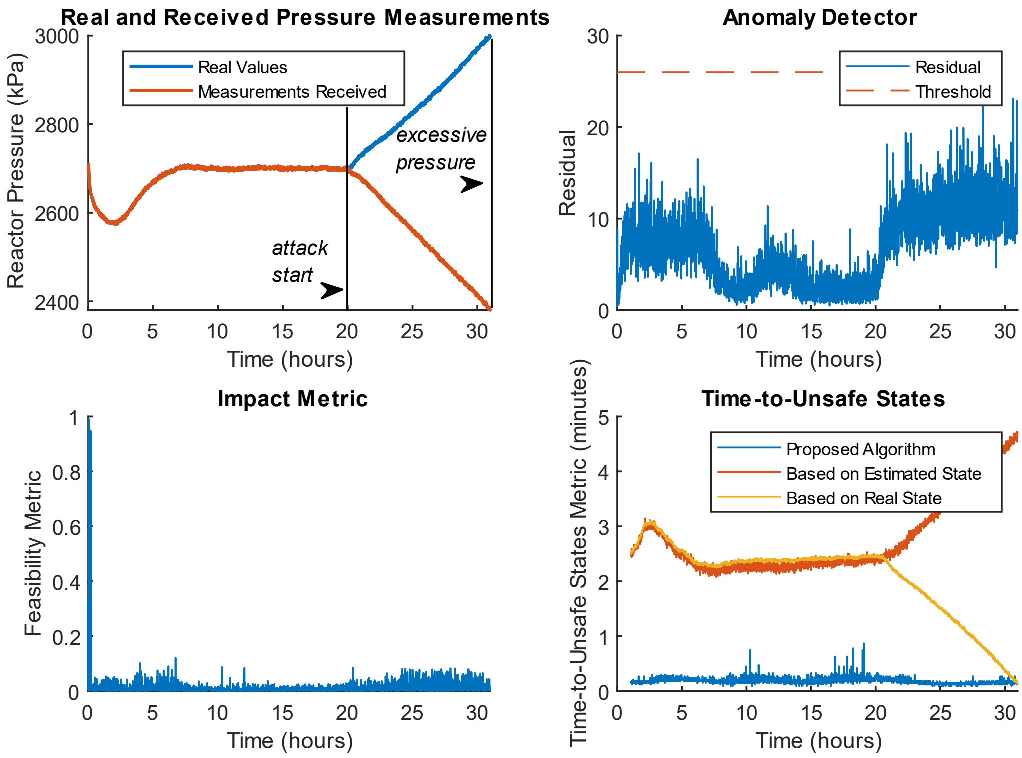


Figure 6.5: Results from simulation Scenario 2.

metric shows an increase over the few hours between the start of the attack and the damage taking place. Instead, the traditional time-to-unsafe states metric shows the plant moving away from unsafe states.

Scenario 3. In this scenario, we simulate simultaneous attacks on the main reactor's pressure, temperature, and level sensors. All three attacks are slowly growing biases. Figure 6.6 shows that damage occurs faster in this scenario than in the previous two, with the anomaly detector again failing to raise any alarms. Our monitoring algorithm however correctly predicts again potential damage from a stealthy attack (non-zero feasibility metric). Instead, the traditional time-to-unsafe states metric depicts the plant moving away from the unsafe operating region.

These scenarios demonstrate the usefulness of our approach in the presence of stealthy attacks when compared to simple time-to-unsafe metrics. Relying on the traditional time-to-unsafe metric may relay an inaccurate idea of the current

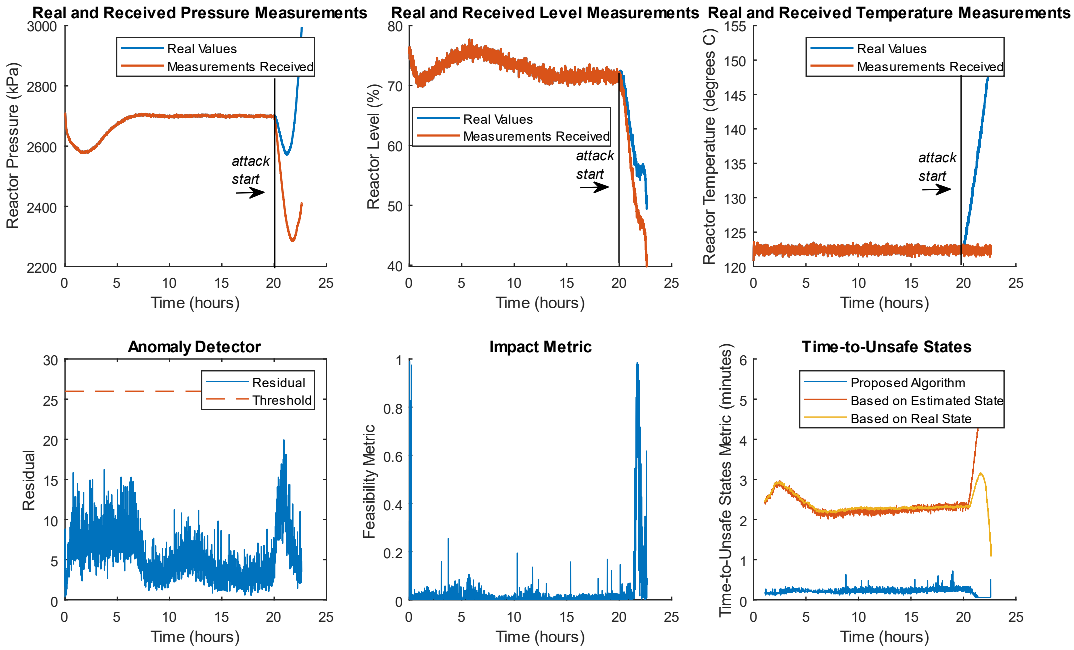


Figure 6.6: Results from simulation Scenario 3.

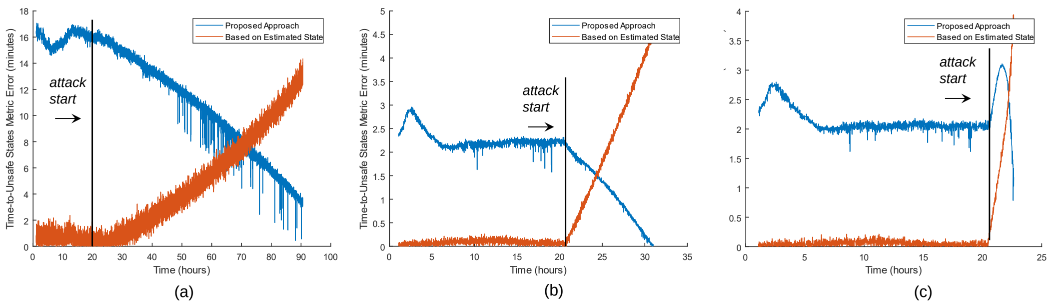


Figure 6.7: Difference (error) between time-to-unsafe metric computed based on state estimate and based on Algorithm 2 vs. the time-to-unsafe states based on the real state of the system; (a) - Scenario 1, (b) - Scenario 2, (c) - Scenario 3.

security or safety conditions. This was especially highlighted in Scenarios 2 and 3. While the plant appears to drift away from the unsafe operating region, our monitoring approach can still warn operators that an attacker is able to damage the system without being detected.

Figure 6.7 shows a comparison between the time-to-unsafe metric computed using our algorithm and the same metric computed based on the raw estimated state. We plot the difference (error) between the metric in each case and the time-to-unsafe states computed based on the real state of the system. In each scenario, we observe that the metric based on the raw estimated state is relatively accurate before the attack starts (the error is close to zero). However, the error starts to grow as the stealthy attack progresses and the real state diverges from its estimate. Conversely, as the stealthy attack progresses, this error decreases for the time-to-unsafe states metric computed according to our algorithm and reaches almost zero towards the end of the attack.

This demonstrates the usefulness of our algorithm in the worst case, where the actual state of the system significantly diverges from the real estimate under a stealthy attack. While this may be overly conservative when the system is not under attack, the safety criticality of the systems we consider justifies the need to employ a monitoring that can generate early warnings in case a dangerous stealthy attack is taking place. Hence, the task of pre-computing reachable sets under attacks and using them for online monitoring is well justified. Additionally, the results that we described previously show that our proposed algorithm for online monitoring can be used as part of the instantiation of our EWS framework for LTI systems.

6.3 Suspicion Metric

In this section, we focus on the suspicion metric component of our *Physics-based EWS* and its ability to raise warnings that can guide different post-warning measures. As the suspicion metric does not necessarily reveal attacks, we do not consider previous work on attack detection (such as the work surveyed by Giraldo et al. (2018)) a suitable baseline to compare our work against. Rather, our evaluation of the suspicion metric is focused on the timeliness of the generated warnings, in other words, whether they are generated well before damage happens. We also demonstrate the usefulness of having different types of warnings in terms of guiding different types of proactive measures. We illustrate such measures, and we show that in particular, the low-criticality warning is a suitable trigger for the proactive collection of potential evidence (i.e. (P1) in Figure 3.1).

As we already quantitatively evaluated the accuracy of the online monitoring algorithm in the previous section, we qualitatively demonstrate the suitability of the low criticality warning as a trigger for data collection through attack scenarios. This approach is in line with previous work on EWS (Abbaszadeh et al. (2018); Apel et al. (2009, 2010); Husák et al. (2018); Kalutarage, Lee, Shaikh and Sung (2015); Kalutarage et al. (2012)) whereby the evaluation of an EWS focuses on its ability to guide certain proactive measures. For instance, Brignoli et al. (2020) evaluate their EWS for network attacks in IoT networks by illustrating how it can be used to evaluate the impact of active countermeasures.

As described in Section 4.4, the criticality of our warnings depends on a comparison of the time-to-unsafe states to the time that would be sufficient to perform different proactive measures, as determined by expert knowledge of the system. For the sake of our current experiments, where we use a $K = 500$ as the length of prediction time horizon (as determined by our previous experiments), we assume that a time period of $t_1 = \Delta_t K/2 \approx 8$ mins would be needed to perform data collection; and $t_2 = 1$ min to perform "more critical" actions, such as switching to a secure backup controller (Chen and Sankaranarayanan (2017)). We implemented these warning levels (within Algorithm 3) accordingly.

The results obtained for the attack scenarios are shown in Figure 6.8. For each scenario, we plot the value of the reactor level or pressure, residual, suspicion metric, and warning level over time. Our results can be interpreted as follows:

1. In Scenario (i) (Figure 6.8-a), we use similar operating conditions and attack as the ones described in the motivating example (Section 3.2). The attack on the reactor starts at $t = 30h$, and the liquids level increases until damage takes place approximately 37 hours later. A high-criticality warning is however issued at around $t = 48h$, 19 hours before damage takes place.

   Before the attack starts, a low-criticality warning level is maintained most of the time. A high-criticality warning is raised for a brief moment after the attack starts, but well before any damage can happen (20 hours). As the attack takes a long time to cause damage, actions associated with a low-criticality level warning are suitable. The operators can for example

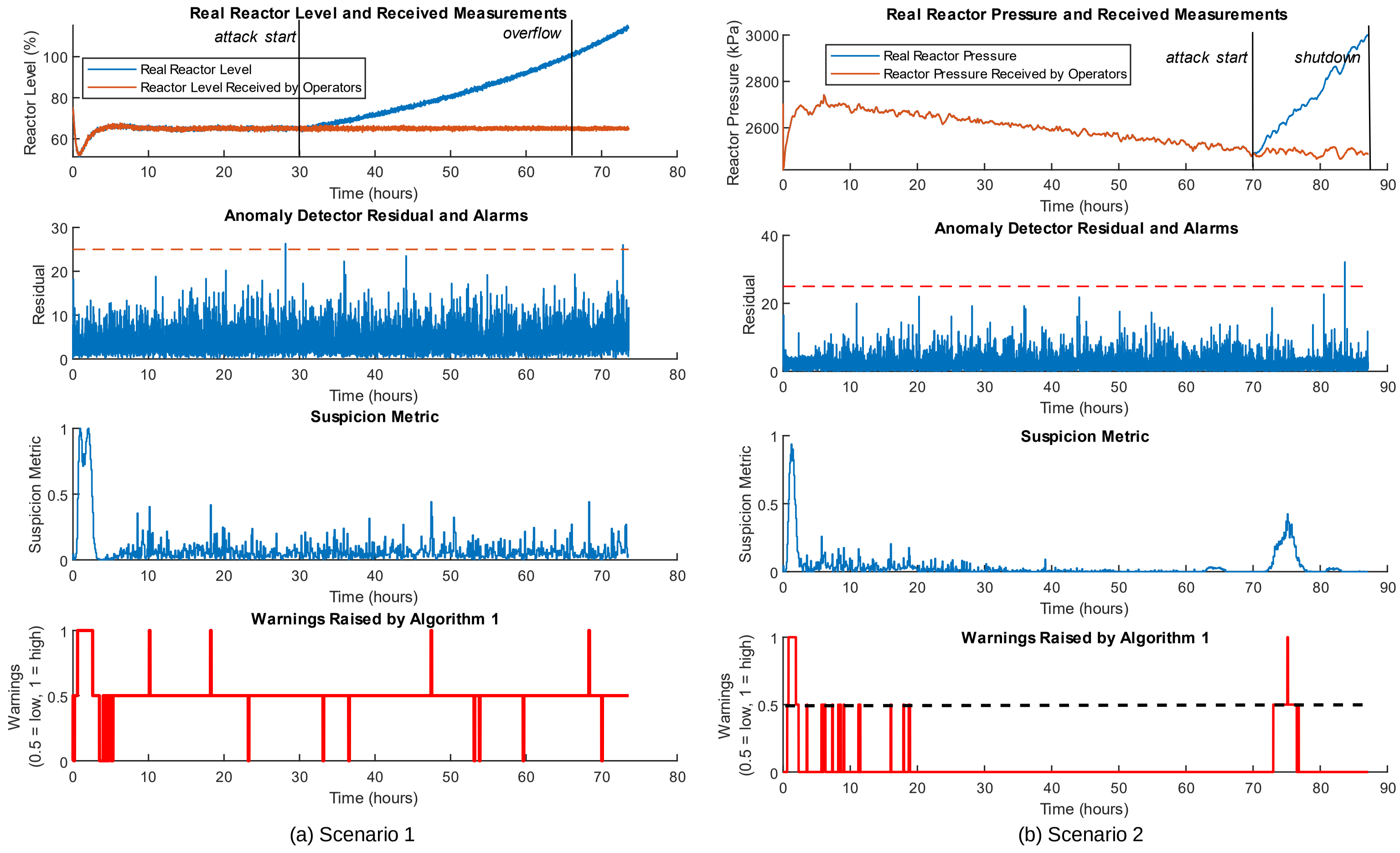


Figure 6.8: Numerical simulation results corresponding to the attack scenarios described in Section 6.3. (a) scenario (i); (b) scenario (ii).

collect more data from different devices to check whether the plant is under a certain breach. Although a low-criticality warning is raised constantly for a long period of time, this does not imply that proactive actions should be taken each time a warning is raised. Indeed, operators can initially collect more data from the concerned area of the system to confirm or refute the hypothesis that an intrusion is present. This data can include network traffic from the concerned area of the system, logs from engineering workstations, PLC configurations - among others (e.g. Table 5.1). If the hypothesis is refuted yet a low-criticality warning level is maintained, then data collection can either stop or take place periodically in order to make sure that no intrusion is present. This task can be automated using existing dedicated tools (Eden et al. (2016)). Moreover, we expect the warnings generated by our framework to be correlated with other alerts (e.g. cyber intelligence, insider activity) generated by the EWS. This will provide a more accurate picture of the security situation. For example, these warnings can feed into a Bayesian score similar to the one proposed by Kalutarage, Lee, Shaikh and Sung (2015); Kalutarage et al. (2013) to monitor potential network intrusions.

2. In Scenario (ii), we consider transient operating conditions (Figure 6.8-b) where the reactor's pressure is steadily brought to lower levels over a long period of time. The attack on the reactor's pressure starts at $t = 70h$, and excessive pressure starts building up in the reactor until damage takes place approximately 18 hours later. A high-criticality warning is however issued at around $t = 75h$, 13 hours before damage takes place. This warning level is consistent with the significantly shorter time taken by this attack when compared with the previous scenario.

The scenarios above show that our proposed *Physics-based EWS* ([**COMP1**]) can warn well in advance of potential harm and can guide the selection of post-warning actions. We have considered attacks on two sensors — reactor level and pressure — which measure safety-critical process variables related to the most important stage of the TEP — its reactor. Scenario (i) shows a case where only a low-criticality warning is needed as the attack takes a long time to cause damage. This illustrates how our choice for a low-criticality warning as a trigger for proactive data collection is sound. Scenario (ii) illustrates a situation where a high-criticality warning with more drastic actions are needed to prevent potential damage to the system.

6.4 Real-Time Performance of the EWS and Discussion

In this section, we present our evaluation of the real-time performance of our proposed *Physics-based EWS* (**[COMP1]**) instantiated for LTI systems. In particular, we evaluate the performance of Algorithm 3 which includes both the safety checking and the computation of the suspicion metric. In light of the performance evaluation results and the results from the previous two sections, we discuss some aspects related to the implementation and the deployment of the algorithms in industrial settings.

6.4.1 Scalability and Network Overhead

Scalability. We assessed how the execution time of Algorithm 3 scales with respect to (i) the number of safety constraints, and (ii) the length of the prediction horizon K set by operators. For both cases, we averaged the execution time over a 100-hour simulation of the networked TEP, equivalent to 2×10^5 executions of the algorithm given the sampling time $\Delta_t = 5 \times 10^{-4}$ hours ≈ 1.8 seconds. In addition, for the purposes of performance testing, we modified Algorithm 3 to simulate the worst-case execution scenario where emptiness checks are performed at every predicted state. To test for scalability against the number of safety constraints, we generated random half-spaces representing potential safety constraints. We also fixed the length of the prediction horizon at $K = 500$, equivalent to about 15 mins into the future. For scalability with the length of the prediction horizon, we used the safety constraints in Table 6.1. Results are shown in Figures 6.9 and 6.10.

The worst-case execution time of the algorithm scales linearly with respect to both the number of safety constraints and the length of the prediction horizon.

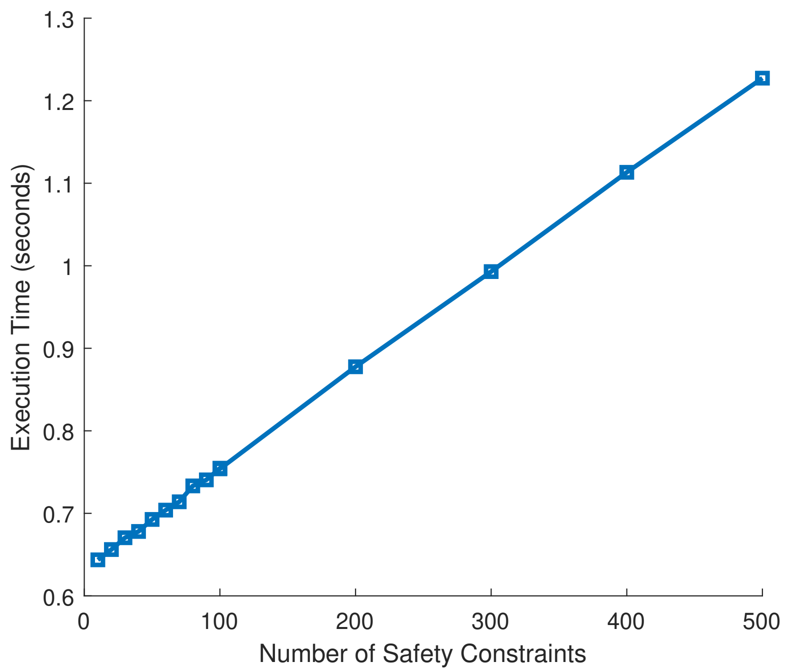


Figure 6.9: Average execution time of Algorithm 3 vs. the number of safety constraints.

These results prove the ability of the proposed algorithm to scale in safety-critical scenarios, where a larger number of safety constraints are imposed. Furthermore, at $K = 1000$ time steps, equivalent to about 30 min ahead-of-time prediction, the worst-case execution time ≈ 1.3 sec is less than the sampling period, 1.8 sec, which guarantees a satisfactory real-time response. Note that real-time response can be further improved by performing checks only when the estimated physical and controller states of the system (i.e. the main real-time inputs to Algorithm 3) are undergoing significant changes, i.e. during transient operation. Finally, the algorithm's worst-case execution time can be improved significantly by considering an implementation in a compiled language such as C rather than an interpreted language like MATLAB.

Network Overhead. To assess the effect of the proposed algorithm on the performance of the network, we measured end-to-end time delays at two critical locations in the network: (i) between each sensor and its corresponding controller;

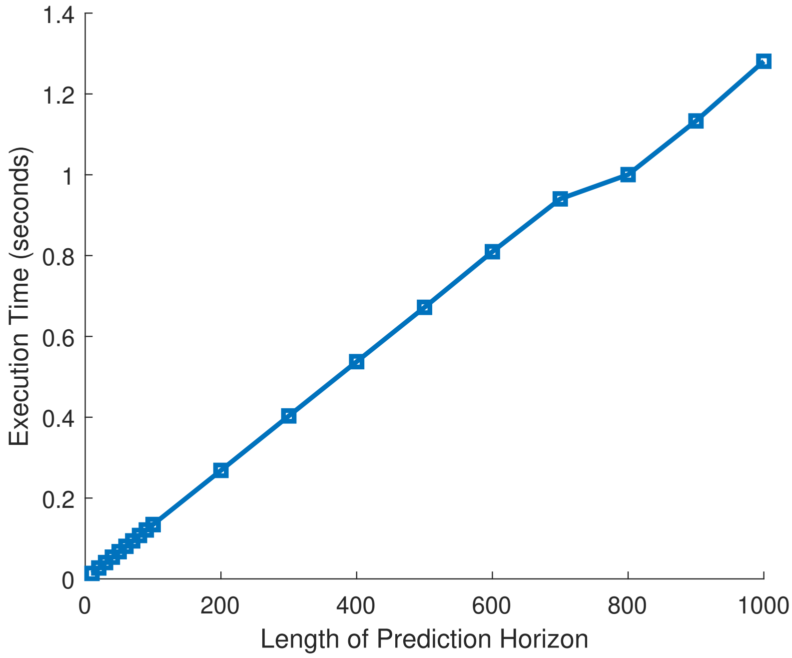


Figure 6.10: Average execution time of Algorithm 3 vs. the length of prediction horizon K.

(ii) between each controller and its corresponding actuator. For each location, we averaged these measurements over a 100-hour simulation. The values in Table 6.3 shows the average, standard deviation, maximum, and minimum time delays considering all sensors, controllers, and actuators on the network. The proposed algorithm incurs little additional delays on the network. This result is expected since the installation of the proposed scheme requires that only sensor values and controller states are uploaded to the supervisory control room area (Figure 6.1). These values are already uploaded to perform anomaly detection in the absence of the proposed framework. It is worth noting that these values are usually uploaded also to process historians for various process control and diagnostics-related logging activities. Therefore, the proposed *Physics-based EWS* is expected to incur little overhead on the network.

Table 6.3: End-to-End transmission time delays in ms between various components of the Tennessee-Eastmane Process (TEP) in the presence of the proposed *Physics-based Early Warning System* (EWS) - [**COMP1**] component, (without it).

	Sensors to Controllers	**Controllers to Actuators**
Average	6.017 (6.013)	22.90 (19.05)
Standard Deviation	2.78 (2.77)	9.17 (8.41)
Maximum	9.65 (9.64)	35.66 (33.91)
Minimum	1.52 (1.52)	10.19 (1.15)

6.4.2 Discussion

In this secion, we make some final remarks regarding the proposed approach to raise physics-based warnings about stealthy attacks in ICS ([**COMP1**]). In particular, we discuss some practical implementation challenges and potential limitations.

6.4.2.1 Implementation Challenges

Our evaluation has shown that the proposed monitoring approach exhibits high accuracy and real-time performance. Our comparison of our reachability-based algorithm with existing monitoring techniques has shown that our approach is better suited for monitoring under stealthy attacks. Additionally, we demonstrated through attack scenarios that our use of a low-criticality warning as a trigger for proactive collection (P1) is a sound choice.

When it comes to its implementation in practice, we note that our monitoring algorithm relies on a model of the system and its anomaly detector, which is normally available during the control design phase. The stealthy attack model that we focus on encompasses several optimised stealthy attack strategies proposed in previous work. This is also evident by the ability of Algorithm 2 to exhibit high accuracy with respect to a large number of randomised attacks in our accuracy tests (Section 6.2). Finally, the attack model's relative practicality with respect to the attacker makes it a reliable heuristic threat model to consider for a practical design of monitoring algorithms.

In our approach, we require some design parameters to be set. For Algorithm 1, a choice for the size of the partitioning interval Δ_h (Murguia and Ruths (2018)) must be made to guide the grid search. This parameter depends entirely on the desired tightness of the reachable set approximation. As this is a design-time activity, the computational cost of choosing an arbitrarily small Δ_h could be negligible when considering the increased tightness of the resulting approximation. Our experiments have shown that even with a choice of $\Delta_h = 0.01$, we can achieve a high accuracy in terms of safety checks. For Algorithms 2 and 3, the design choice is to set a proper length for the prediction horizon K. We showed that extensive randomised simulations performed in Section 6.2 can help in choosing a value for K that achieves an acceptable trade-off between how "early" we would like to raise warnings vs. the accuracy of Algorithm 2 (Figures 6.2-6.3).

6.4.2.2 Limitations

In our instantiation of the physics-based EWS framework to LTI systems, we relied on adapting existing reachability analysis tools to perform real-time safety monitoring. To alleviate the computational cost incurred by reachability analysis, we relied on the ellipsoidal nature of the approximated reachable sets and applied results in ellipsoid calculus that allowed our approach to be deployed in real-time. We realise nonetheless that such tools are specific to systems that can be approximated using an LTI model. To increase generalisability of our results, in future work we will explore adoption of a different set of tools to instantiate our framework to different types of systems. Finally, from a security perspective, to reduce the risk of a potential adversary subverting the EWS, one possible solution is to implement it on a Shadow Security Unit (SSU) as proposed by Graveto et al. (2019). Such devices are computers designed specifically to remain hidden from potential offenders. Encryption mechanisms can also be added as an extra layer of security, to favour more secure communications at the cost of potentially reduced performance.

6.5 Identification and Selection of Relevant Data

In this section, we evaluate the relevance of the data identified by the *Relevant Data Identifier* ([**COMP2**]), the reduction in the control performance overhead caused by real-time data collection due to the *Relevant Data Selector* ([**COMP3**]) component, and the applicability of our data collection approach in terms of supporting live forensic investigations.

We first evaluate whether [**COMP2**] misses any relevant data, i.e. data that could be lost to physical damage by a stealthy attack. Then, we compare the case where [**COMP2**] and [**COMP3**] are used to identify and select relevant data for collection (Algorithm 4) with the case where all data generated by an ICS is collected at all times. We highlight the improvement in terms of control overhead as a result of our decision-theoretic framework that achieves a trade-off between data collection cost and expected attack impact. Finally, we show a use case of our approach to proactive data collection in the area of live forensics. To perform this evaluation, we simulated the process of data collection in the TEP testbed. We begin this section with a brief description of this implementation, followed by the evaluation of components [**COMP2-3**].

6.5.1 Simulation of Real-time Data Collection

To simulate the process of data collection, we augment the controllers in the TEP simulation with logging capabilities. When Algorithm 4 identifies the set $D_{col}(k)$, the corresponding controllers generate a log in a `json` format. In particular, we assume that each of the controllers designed by Ricker (1996) are implemented on a PLC. Therefore, we simulate the collection of data from these devices by adopting a log structure that mimics the logging of PLC ladder logic configurations (e.g. as in the work of Yau et al. (2018)). Namely, each log entry consists of a timestamp, sensor measurements received, and details about the controller configuration, in addition to the output of the controller (actuation signal). Since each controller in the architecture proposed by Ricker (1996) is a discrete-time Proportional-Integral (PI) controller, the controller configuration is represented by the values of the coefficients of the proportional and integral terms in addition to the sampling time. For sensor measurements, the corresponding sensors simply generate a `json` log entry that includes the timestamp, the value, and the name of the sensor measurement (Table 6.1).

6.5.2 Relevance of Collected Data and Effect on Control Performance

To assess the relevance of the data collected according to Algorithm 4 along with the control performance overhead, we adopt a large number of randomised (Monte Carlo) simulations. In each simulation, we randomise the operating levels of the TEP (i.e. set-points for pressure, temperature, reactor level, etc.), and we choose attacked sensors randomly. For relevance, we check in each simulation whether any data that was lost during the attack was missed by the set identified by the *Relevant Data Identifier* component [**COMP2**] ($D_{col}(k)$).

Since by design our algorithm only collects data when the collection cost does not exceed the expected reduction in revenue, we do not evaluate this part of our approach by comparing the revenue before and after collection. Instead, we showcase how, as a result of the *Relevant Data Selector* ([**COMP3**]), there is a reduced effect on the system's performance. We compare the average execution time of controllers under forensic logging enabled by Algorithm 4, with the case where forensic logging is enabled at all times from all controllers. We also present the average reduction in terms of the number of log entries between the two scenarios. Although our logging mechanism is a simplified version of data acquisition tools, the comparison of average controller execution times between the aforementioned scenarios serves to give an idea of the potential performance improvement brought by our approach with actual data acquisition tools.

For each of the testing scenarios previously described, we ran a total of 500 simulations. Results are shown in Table 6.4.

Discussion. The low percentage of data missed by the proposed algorithm as shown in Table 6.4 demonstrates the capability of our approach to identify relevant data (P2, [**COMP2**]), thus limiting the loss of potential evidence due to damage by a stealthy attack. Additionally, the use of Algorithm 4 with the proposed decision-theoretic framework (P3, [**COMP3**]) results in reducing the average controllers' execution time by around 35%, compared with the case where

Table 6.4: Simulation results for the *Relevant Data Identifier* ([**COMP2**]) and *Relevant Data Selector* ([**COMP3**]) components, Algorithm 4.

Test	**Result**
Relevance of Identified Data ([**COMP2**]) (percentage of cases where data at risk of being damaged is missed)	5.8 %
[**COMP3**]: Average controllers' execution time with Algorithm 4	1.5×10^{-3} secs
[**COMP3**]: Average controllers' execution time without Algorithm 4, "full" logging enabled)	2.3×10^{-3} secs
[**COMP3**]: Average reduction in log size when using Algorithm 4	30.8 %

logging was performed from all controllers at all times. This is consistent with the 31% reduction in the size of logs due to Algorithm 4. These results show that the proposed decision-theoretic framework enables the collection of data in a way that reduces the interference with control operations.

6.5.3 Use Case: Supporting PLC Log-based Live Forensics

To demonstrate the applicability of our approach to a potential use case, we consider tools that can perform live forensics in ICS, i.e. investigating potential attacks while the system is running. Due to the safety-criticality of ICS and the difficulty associated with shutting them down, methods for live forensics activities (Al-Sharif et al. (2018)) have been proposed to avoid such difficulties. One important requirement for such methods is that they need to minimise interference with the safety-critical control operations (Eden et al. (2016)). In the previous section, we showed how our approach reduces the number of PLC logs that need to be collected, hence reducing the effect of logging mechanisms on the controllers' execution time. In this section, we further show how our approach can be used to improve the performance of live forensics operations, due to the fact that we only consider relevant data ([**COMP2**]) and we account for the system's performance while guiding logging activities ([**COMP3**]).

A class of ICS live forensics methods (Yau and Chow (2015)) consists of investigating PLC logs to check for any malicious code modifications to their code. To showcase the increase in performance of such methods, we revisit the Stuxnet-inspired attack scenario introduced in Section 3.2. In this section, we consider the case where in the second phase of the attack, the attacker modifies the function codes in the PLC controlling the reactor pressure in a way that a slowly growing bias is applied to pressure measurements received by the PLC. We run a simulation of such an attack by introducing malicious code to the controller handling the reactor pressure. We augment the implementation of the Algorithm 4 and the logging mechanism described previously with an implementation of the technique proposed by Yau and Chow (2015) based on the log structure that we use in our simulations. Whenever controller logs are generated, we simply compare the logged control output with the one expected by the originally designed control law (as designed by Ricker (1996)). If the two values differ substantially, i.e. more than a pre-defined threshold, the code checking detects the presence of a code modification (Figure 6.11).

Figure 6.11 shows a plot of the received and actual (real) pressure measurements from the reactor under an integrity attack on pressure sensor, along with the anomaly detector's detection metric. In addition, we show a plot of the results from the code modification checks performed to the pressure controller logs collected according to our approach. We plot the results from the code checking vs. the timestamps of the collected logs. We also analyse the performance of PLC code checking under two cases: (1) the checking is performed on logs generated according to Algorithm 4 and (2) the checking has to consider logs from all controllers at each time instant. The results are shown in Table 6.5 and were averaged over a $\approx$ 30-hour simulation, equivalent to around 60000 real-time checks given the system's 1.8 second sampling time.

Discussion. Figure 6.11 shows that our approach enabled the collection of logs from the reactor's pressure controller, where code checking was able to detect modification of the PLC code well before any alarm was raised by the existing

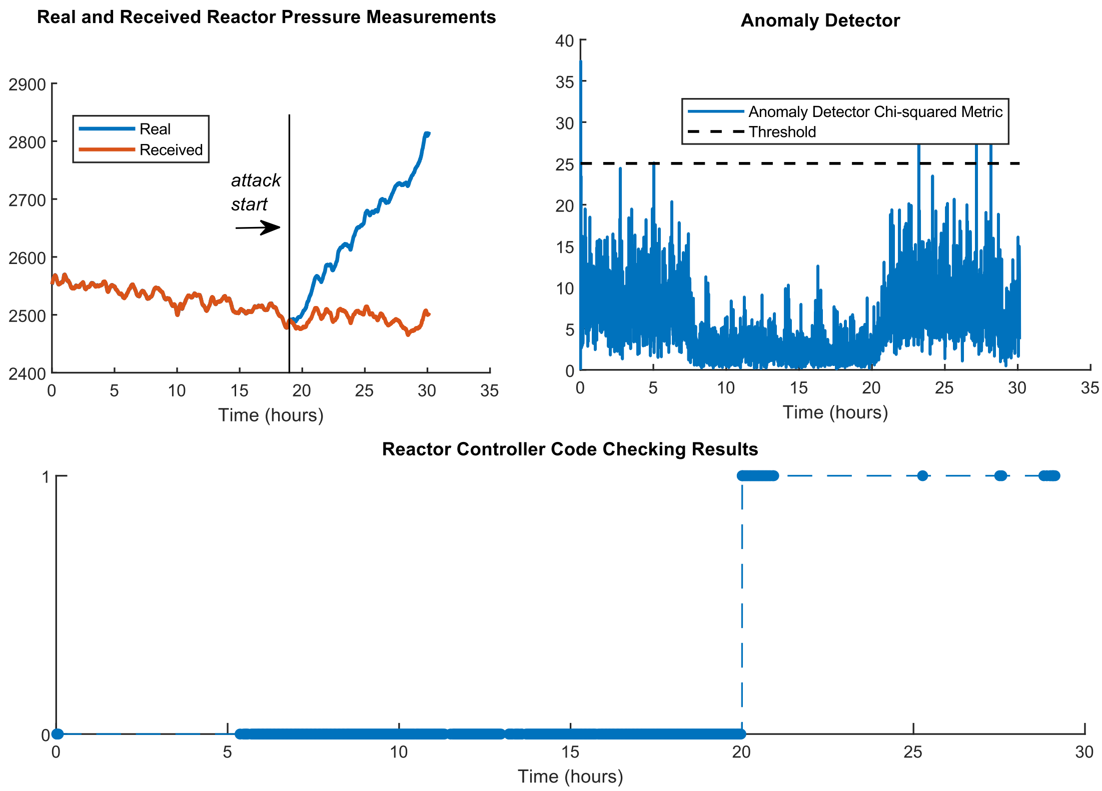


Figure 6.11: Real and received sensor measurements, anomaly detector metric, and results from the real-time Programmable Logic Controller (PLC) code checking applied with the proposed approach (1 = detected code modification, 0 = otherwise).

Table 6.5: Programmable Logic Controller (PLC) code checking performance.

Test	**Average Real-time Execution (seconds)**
With Algorithm 4	0.0013
Without Algorithm 4	0.0277

anomaly detector. While our approach only collected logs from the pressure controller at the start of the attack, it enabled the detection of code modifications much earlier than the existing anomaly detector. In the case where logs are needed for the duration of attacks, Algorithm 4 can be modified such that it can be overridden to ensure the collection of logs as long as the code checking returns with the result that the code is indeed modified.

The benefit of performing PLC code checking under the proposed approach is highlighted with the results shown in Table 6.5. Under this approach, the average real-time execution time of code checking was reduced by more than an order of magnitude. In ICS, live forensics could be more useful than post-mortem forensics (Eden et al. (2016)) as it eliminates the high-costs of shutting down the process. The performance enhancement brought about by the proposed approach has the potential of reducing the risk of interfering with the real-time performance constraints of safety-critical ICS when live forensics is performed, especially considering that devices such as PLC's feature low computational resources.

Finally, it is worth noting that live forensics through PLC code checking and network forensics tools may serve as a heuristic to determine whether proactively collected data is to be preserved or discarded. For instance, in the scenario discussed previously, we note that our approach has triggered the collection of logs for a significant amount of time before the attack started. As no code modification was detected through these checks, they could be discarded in order to save storage resources.

6.6 Summary

We presented in this chapter a thorough evaluation of our approach to engineer Forensic Readiness (FR) in ICS faced with stealthy attacks. First, we focused on the *Physics-based EWS* for stealthy attacks on Linear Time-Invariant (LTI) ICS that forms the first component ([**COMP1**]) of the overall approach. We evaluated the accuracy of our online safety monitoring algorithm and compared it with existing work, where results justified our reliance on reachability analysis tools. We then showcased through attack scenarios that our choice of a low-criticality warning is a suitable trigger for proactive data collection. We ended our evaluation of this component with an assessment of its real-time performance where results showed its feasibility of deployment in practice within large-scale systems. We also discussed some potential limitations, namely in terms of the effort required to tune the monitoring algorithm and instantiate the EWS, and in the latter's generalisability to systems that cannot be modelled as LTI.

Second, we evaluated the relevance of the data identified by the *Relevant Data Identifier* ([**COMP2**]) and the reduction in the control performance overhead caused by real-time data collection due to the *Relevant Data Selector* ([**COMP3**]). We showed first that our proactive collection approach has a low probability of missing data that is actually at risk of being lost to damage ([**COMP2**]), i.e. relevant data. Then, we showed that our approach to data collection results in a reduced overhead on the control performance ([**COMP3**]) when compared with an agnostic approach to data collection (i.e. all data is collected at all times). Finally, we showcased the applicability of our approach in improving the efficiency of live forensics tools. In particular, we presented a use case scenario where our proactive data collection approach significantly improved the efficiency of live PLC log analysis (Yau and Chow (2015)).

Chapter 7

Conclusions and Future Directions

In this thesis, we tackled the problem of engineering Forensic Readiness (FR) in Industrial Control Systems (ICS) faced with stealthy attacks. To this end, we divided our problem into three parts: (1) providing a timely trigger for proactive data collection; (2) identifying what data is relevant to forensic investigations; and (3) selecting a subset of this data to be collected such that potential control performance overhead is reduced. To address these challenges, we proposed an approach whose novelty stems from the combination of its three components.

The first component of the present work is a framework for *Physics-based Early Warning Systems (EWS)* to provide early warnings about damage from potential stealthy attacks targeting physical processes in an ICS. This model-based framework defines two preliminary indicators, or grounds for suspicion, of a stealthy attack: (i) the feasibility of an attack, defined as its ability to drive the system into unsafe states given the system's dynamics and the existing anomaly detector; and (ii) the proximity of the system to a predefined set of unsafe states. Model-based measures of these indicators compose what we call a *suspicion metric*, which in turn measures the probability that *if* the system is subjected to a stealthy attack, then the latter will cause damage without getting detected. As such indicators are model-based, and given the variety of modelling frameworks for control systems, we proposed soundness principles to guide the instantiation of the EWS framework to a particular class of systems. EWS designed according to our framework are not meant to replace existing IDS/ADS, rather to complement them by providing early predictions and advice.

To showcase the applicability of the EWS framework, we instantiated it to a widely studied class of control systems — Linear Time-Invariant (LTI) systems. We also considered a practical and widely studied stealthy attack model consisting of modifying sensor measurements. To provide sound measures of feasibility and proximity, we designed an efficient online safety monitoring algorithm. We followed an approach that consists of pre-computing symbolic reachable sets offline by considering how the estimation error would evolve under attacks (Murguia and Ruths (2018)). In real-time, given a current state estimate, the symbolic reachable sets are instantiated online and emptiness checks of their intersection with the set of unsafe states are performed. This approach and the ellipsoidal nature of reachable set approximations have allowed us to perform the most expensive computations offline, while reducing online safety checks to a mere checking of the sign of a distance measure. Additionally, we derived a suspicion metric composed of measures of feasibility and proximity whose efficient computation takes advantage of existing results in ellipsoid calculus (Boyd et al. (2004); Dai and Pelckmans (2012); Kurzhanskiy and Varaiya (2006)). Based on these measures, we proposed two thresholds for warnings: the first, a low-criticality warning, implies a low-likelihood for damage due to a stealthy attack taking place; the second, a high-criticality warning, implies that a potential attacker could cause damage in a small period of time. We proposed that the first type of warning is a suitable trigger for proactive data collection, as it suggests that there is sufficient time for such activities to take place. The second type of warning implies the need for more immediate, potentially more expensive measures that may stop a potential intrusion. We focused on the low-critical warning and the subsequent data collection in this thesis.

The second main component of the work in this thesis, the *Relevant Data Identifier*, is an approach to identify *relevant* data that can support forensic investigations. After triggering data collection activities through a low-criticality warning, we proposed an automated technique to identify data that is likely to be lost due to physical damage, which we consider to be relevant. We take advantage of the safe Process Plant Layout (PPL) (Quiroz-Pérez et al. (2021)) to map potentially violated safety constraints (as determined by our EWS) to plant areas that will be damaged due to this (uncertain) predicted violation. We then set our data acquisition tools to consider data originating from control devices located in these areas.

The third main component of the work, the *Relevant Data Selector*, consists of a framework to select a subset of relevant data to collect. This selection is performed such that a trade-off between attack impact and potential overhead caused by real-time data collection is achieved. This framework essentially computes the value of collecting more data in terms of the expected impact of the suspected attack. Data is then collected only if its collection cost does not exceed its estimated value. This component results in a reduction the control performance overhead that would be caused by real-time data collection.

We evaluated our approach by considering a case study based on the benchmark Tennessee-Eastman Process (TEP) (Downs and Vogel (1993)). We used a Matlab/Simulink simulation of this realistic chemical process to test different aspects of our work. Our initial tests were focused on the accuracy and real-time performance of the online safety monitoring which forms the core of our *Physics-based EWS*. We showed through extensive Monte Carlo simulations that the algorithm exhibits a high degree of accuracy, performs well in real-time, and scales well with the complexity of unsafe sets and the length of the prediction horizon. We then showcased through certain scenarios the advantage of using the proposed technique when compared with existing work that relies only on raw sensor measurements to assess the system's proximity to unsafe states. We also showed through additional scenarios the soundness of using the proposed low-criticality warning as a trigger for proactive data collection.

We then evaluated the relevance of the data identified by our *Relevant Data Identifier* and the improvement in control performance overhead due to our *Relevant Data Selector*. Through additional sets of Monte Carlo simulations, we showed that our automated technique for relevant data identification rarely misses any data that is at risk of being damaged. Furthermore, we demonstrated the ability of our framework for relevant data selection to incur a reduced controller performance overhead when compared to the case where "agnostic" all-data collection is performed. Finally, we illustrated the applicability of our proactive data collection approach to a use case in ICS live forensics. We showed that using our approach can significantly improve the performance of Programmable Logic Controller (PLC) log-based live forensics tools (Yau and Chow (2015)) in terms of execution time.

The work described in this thesis lays the foundations for a novel approach to the investigation of advanced attacks that can avoid detection through knowledge about existing Anomaly/Intrusion Detection Systems (ADS/IDS). EWS can assist operators by providing early, uncertain predictions of potential harm. Such these predictions, by themselves, do not necessarily imply that the system is subjected to an attack. However, their advantage lies in their ability to guide certain actions that may reveal potential advanced threats that could otherwise evade existing IDS/ADS. In ICS, the framework for physics-based early warnings that we proposed in this thesis can fit within a more comprehensive EWS that can aggregate "weak evidence" from a variety of other sources, e.g. internet databases and cyber intelligence (Apel et al. (2010); Theilmann (2010)), to provide better predictions of potential harm. Given a prediction of harm returned by such an EWS, our proposed approach for proactive data collection can decide on a subset of relevant data to proactively collect. With attacks on ICS that can cause physical damage by manipulating data in control devices, our approach considered data that is at risk of being lost to such damage to be relevant. Other definitions of relevance can be formulated in different contexts. For example, the relevance of data to investigations of insider threats (Chivers et al. (2009); Kalutarage et al. (2012)) can be associated with specific network nodes and segments that are suspicious. Furthermore, our framework for relevant data selection can be used to decide whether it is worth collecting more data based on a trade-off between potential attack impact and a measure of collection cost — measures that can be defined depending on the context.

Future Work. Our work can be extended in several different ways. First, the applicability of our framework for physics-based EWS to other types of control systems can be investigated. In this thesis, we focused on the widely-used LTI modelling formalism. Designing efficient and scalable online safety monitoring algorithms under physics-based attacks for other systems can be however challenging. Using approaches such as offline computation of symbolic reachable sets with a low-cost real-time instantiation (Althoff et al. (2015); Chen et al. (2017)), making use of efficient short-time reachability analysis tools (Tran et al. (2019)), or employing statistical modelling tools to perform safety checks (Allen et al. (2014); Chou et al. (2020); Zhu and Başar (2011)) can provide a potential starting point for developing efficient real-time predictive monitoring. Statistical modelling techniques may also help in approximating feasibility and proximity measures when the computation of the geometric features of reachable sets is intractable. In addition to investigating other types of control systems, the applicability of our framework can be investigated for other threat models for stealthy attacks. In our work, we considered False Data Injection Attacks (FDIA's) on sensors only. Other threat models could include, for example, FDIA's where all sensors and actuators are manipulated simultaneously. Such stealthy attacks, referred to as covert attacks (Smith (2015)), may be difficult to implement in practice. However, there may be certain ICS where such attacks may be possible, thus requiring an extension of our approach to include the appropriate threat models.

Second, our evaluation has revealed that in several instances, warnings may be constantly raised to operators even if no attack is present. While this behaviour is expected since warnings from an EWS do not necessarily imply that an attack is present, a high number of warnings in a security context can cognitively overload operators, as shown in a recent survey by Bryant and Saiedian (2020). Thus, effort from the Human-Computer Interaction (HCI) community is needed to research tools and techniques that can relay such warnings in an efficient manner to operators. Another way to potentially reduce the number of such warnings is to combine them with alarms or warnings from other sources — resulting in a more accurate depiction of the current security/safety situation (Di Pietro et al. (2015)). Such a comprehensive EWS could rely, for instance, on techniques from sensor fusion (Cruz et al. (2016); Di Pietro et al. (2015); Zhang et al. (2019)) to compute a better probability of harm (suspicion metric).

Third, in the development of our framework for relevant data selection, we assumed that all kinds of relevant data hold the same value, i.e., are equally likely to constitute evidence of a potential attack. Hence, our Value of Information (VoI) analysis only focused on the cost of collection as a quantity to trade-off with the potential impact. In future work, this assumption could be dropped, since different kinds of data may hold different value for forensic investigations. For example, if it was known that there is a higher likelihood for a PLC configuration to be modified than a certain sensor value, than it may be more worthwhile to collect data about the PLC than to collect a network packet involved in communicating the sensor measurement — even if the latter has a lower collection cost. In this work, we restricted our attention to PLC configurations and attacks that involve PLC code modifications. Future work could include an assessment of the usefulness of our approach in terms of using other types of proactively collected data to analyse how other kinds of attacks may have occurred. For example, instead of PLC code modifications, other kinds of attacks could include instances where different control devices, such as engineering workstations, are subverted. In addition, such future work would investigate how different types of proactively collected data can be used to understand when such collection should terminate.

Finally, although we focused in this thesis on data collection as a proactive measure, we recognise the need to consider certain preemptive actions that can stop a potential intrusion if the early warning raised by the EWS is high enough. Similarly to data collection, such measures can involve additional costs, possibly exceeding the potential cost of data collection. For example, switching to a safe and less efficient controller (Chen and Sankaranarayanan (2017)) can translate into a reduction in performance and increase in operational costs. The framework that we proposed for the selection of relevant data to collect can be further extended to include the selection of different preemptive actions with associated measures of cost and value. In the future, we seek to extend this framework with more possibilities of preemptive actions (e.g. switching to backup controllers, activating certain fail-safe systems), and with more precise measures of value and cost associated with each action. We will also include a comprehensive economic model (Peters et al. (2003)) of the ICS that can be used to compute more accurate predictions of expected attack impacts and operational costs associated with data collection or preemptive measures.

References

Ab Rahman, N. H., Glisson, W. B., Yang, Y. and Choo, K.-K. R. (2016), 'Forensic-by-design framework for cyber-physical cloud systems', *IEEE Cloud Computing* **3**(1), 50–59. 29

Abbaszadeh, M., Mestha, L. K. and Yan, W. (2018), Forecasting and early warning for adversarial targeting in industrial control systems, *in* '2018 IEEE Conference on Decision and Control (CDC)', pp. 7200–7205. 44, 124

Ahmed, I., Obermeier, S., Naedele, M. and Richard III, G. G. (2012), 'Scada systems: Challenges for forensic investigators', *Computer* **45**(12), 44–51. 28

Ahmed, I., Obermeier, S., Sudhakaran, S. and Roussev, V. (2017), 'Programmable logic controller forensics', *IEEE Security and Privacy* **15**(6), 18–24. 2, 60, 61, 63, 98

Akowuah, F. and Kong, F. (2021), Real-time adaptive sensor attack detection in autonomous cyber-physical systems, *in* '2021 IEEE 27th Real-Time and Embedded Technology and Applications Symposium (RTAS)', IEEE, pp. 237–250. 39, 80

Al-Sharif, Z. A., Al-Saleh, M. I., Alawneh, L. M., Jararweh, Y. I. and Gupta, B. (2018), 'Live forensics of software attacks on cyber–physical systems', *Future Generation Computer Systems* . 28, 29, 134

Albright, D., Brannan, P. and Walrond, C. (2011), 'Stuxnet malware and natanz: Update of isis december 22, 2010 report', *Institute for Science and International Security* **15**, 739883–3. 1, 32, 60

Alenezi, A., Hussein, R. K., Walters, R. J. and Wills, G. B. (2017), A framework for cloud forensic readiness in organizations, *in* '2017 5th IEEE International Conference on Mobile Cloud Computing, Services, and Engineering (MobileCloud)', IEEE, pp. 199–204. 26

Alguliyev, R., Imamverdiyev, Y. and Sukhostat, L. (2018), 'Cyber-physical systems and their security issues', *Computers in Industry* **100**, 212–223.
URL: *http://www.sciencedirect.com/science/article/pii/S0166361517304244* 21

Allen, R. E., Clark, A. A., Starek, J. A. and Pavone, M. (2014), A machine learning approach for real-time reachability analysis, *in* '2014 IEEE/RSJ International Conference on Intelligent Robots and Systems', IEEE, pp. 2202–2208. 48, 143

Alrajeh, D., Pasquale, L. and Nuseibeh, B. (2017), On evidence preservation requirements for forensic-ready systems, *in* 'Proceedings of the 2017 11th Joint Meeting on Foundations of Software Engineering', ACM, pp. 559–569. 2, 25, 26

Alrimawi, F., Pasquale, L., Mehta, D., Yoshioka, N. and Nuseibeh, B. (2019), 'Incidents are meant for learning, not repeating: Sharing knowledge about security incidents in cyber-physical systems', *arXiv preprint arXiv:1907.00199* . 29

Alrimawi, F., Pasquale, L. and Nuseibeh, B. (2017), Software engineering challenges for investigating cyber-physical incidents, *in* 'Proceedings of the 3rd International Workshop on Software Engineering for Smart Cyber-Physical Systems', IEEE Press, pp. 34–40. 25, 27

Althoff, D., Althoff, M. and Scherer, S. (2015), Online safety verification of trajectories for unmanned flight with offline computed robust invariant sets, *in* '2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)', IEEE, pp. 3470–3477. 48, 143

Altschaffel, R., Hildebrandt, M., Kiltz, S. and Dittmann, J. (2019), Digital forensics in industrial control systems, *in* 'International Conference on Computer Safety, Reliability, and Security', Springer, pp. 128–136. 2, 28, 63

Alur, R. (2015), *Principles of cyber-physical systems*, MIT Press. 1, 21, 46

Amin, B. M. R., Anwar, A. and Hossain, M. J. (2019), Distinguishing between cyber injection and faults using machine learning algorithms, *in* '2018 IEEE Region Ten Symposium (Tensymp)', IEEE, pp. 19–24. 40

Amin, S., Cárdenas, A. A. and Sastry, S. S. (2009), Safe and secure networked control systems under denial-of-service attacks, *in* 'International Workshop on Hybrid Systems: Computation and Control', Springer, pp. 31–45. 39

Amin, S., Litrico, X., Sastry, S. and Bayen, A. M. (2012*a*), 'Cyber security of water scada systems—part i: Analysis and experimentation of stealthy deception attacks', *IEEE Transactions on control systems Technology* **21**(5), 1963–1970. 36

Amin, S., Litrico, X., Sastry, S. S. and Bayen, A. M. (2012*b*), 'Cyber security of water scada systems—part ii: Attack detection using enhanced hydrodynamic models', *IEEE Transactions on Control Systems Technology* **21**(5), 1679–1693. 36

Anderson, R. (2020), *Security engineering: a guide to building dependable distributed systems*, John Wiley & Sons. 23

Aoudi, W., Iturbe, M. and Almgren, M. (2018), Truth will out: Departure-based process-level detection of stealthy attacks on control systems, *in* 'Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security', ACM, pp. 817–831. 40, 62

Apel, M., Biskup, J., Flegel, U. and Meier, M. (2009), Towards early warning systems–challenges, technologies and architecture, *in* 'International Workshop on Critical Information Infrastructures Security', Springer, pp. 151–164. 43, 124

Apel, M., Biskup, J., Flegel, U. and Meier, M. (2010), *Early Warning System on a National Level: Project AMSEL*, Universitätsbibliothek Dortmund. 43, 44, 124, 142

Awad, R. A., Beztchi, S., Smith, J. M., Lyles, B. and Prowell, S. (2018), Tools, techniques, and methodologies: A survey of digital forensics for scada systems, *in* 'Proceedings of the 4th Annual Industrial Control System Security Workshop', ACM, pp. 1–8. 27

Ayas, M. S. and Djouadi, S. M. (2016), Undetectable sensor and actuator attacks for observer based controlled cyber-physical systems, *in* '2016 IEEE Symposium Series on Computational Intelligence (SSCI)', IEEE, pp. 1–7. 37

Bai, C.-Z., Pasqualetti, F. and Gupta, V. (2015), Security in stochastic control systems: Fundamental limitations and performance bounds, *in* 'American Control Conference (ACC), 2015', IEEE, pp. 195–200. 3, 29, 36, 37

Bathelt, A., Ricker, N. L. and Jelali, M. (2015), 'Revision of the tennessee eastman process model', *IFAC-PapersOnLine* **48**(8), 309–314. 9, 61, 111, 113

Bernieri, G., Conti, M. and Pozzan, G. (2019), Amon: an automaton monitor for industrial cyber-physical security, *in* 'Proceedings of the 14th International Conference on Availability, Reliability and Security', ACM, p. 33. 32

Bonakdarpour, B., Deshmukh, J. V. and Pajic, M. (2018), Opportunities and challenges in monitoring cyber-physical systems security, *in* 'International Symposium on Leveraging Applications of Formal Methods', Springer, pp. 9–18. 39

Boyd, S., Boyd, S. P. and Vandenberghe, L. (2004), *Convex optimization*, Cambridge university press. 75, 87, 140

Bradford, P. G., Brown, M., Perdue, J. and Self, B. (2004), Towards proactive computer-system forensics, *in* 'International Conference on Information Technology: Coding and Computing, 2004. Proceedings. ITCC 2004.', Vol. 2, IEEE, pp. 648–652. 64

Brahimi, B., Rondeau, E. and Aubrun, C. (2006), 'Comparison between networked control system behaviour based on can and switched ethernet networks', *arXiv preprint cs/0611149* . 112

Brignoli, M., Mazzaro, S., Fortunato, G., Corà, A., Matta, W., Romano, S., Ruggiero, B. and Coscia, V. (2020), Combining exposure indicators and predictive analytics for threats detection in real industrial iot sensor networks, *in* '2020 IEEE International Workshop on Metrology for Industry 4.0 & IoT', IEEE, pp. 423–428. 44, 124

Bryant, B. and Saiedian, H. (2020), 'Improving siem alert metadata aggregation with a novel kill-chain based classification model', *Computers and Security* p. 101817. 143

Bu, L., Xing, S., Ren, X., Yang, Y., Wang, Q. and Li, X. (2019), Incremental online verification of dynamic cyber-physical systems, *in* '2019 Design, Automation and Test in Europe Conference and Exhibition (DATE)', IEEE, pp. 782–787. 48

Carcano, A., Coletta, A., Guglielmi, M., Masera, M., Fovino, I. N. and Trombetta, A. (2011), 'A multidimensional critical state analysis for detecting intrusions in scada systems', *IEEE Transactions on Industrial Informatics* **7**(2), 179–186. 9, 44, 49, 50, 63, 75

Castellanos, J. H. and Zhou, J. (2019), A modular hybrid learning approach for black-box security testing of cps, *in* 'International Conference on Applied Cryptography and Network Security', Springer, pp. 196–216. 9, 44, 49, 50, 63, 75, 76

Cervin, A., Henriksson, D., Lincoln, B., Eker, J. and Arzen, K.-E. (2003), 'How does control timing affect performance? analysis and simulation of timing using jitterbug and truetime', *IEEE control systems magazine* **23**(3), 16–30. 10, 112

Chan, C.-F., Chow, K.-P., Yiu, S.-M. and Yau, K. (2018), Enhancing the security and forensic capabilities of programmable logic controllers, *in* 'IFIP International Conference on Digital Forensics', Springer, pp. 351–367. 28

Chan, E., Venkataraman, S., David, F., Chaugule, A. and Campbell, R. (2010), Forenscope: A framework for live forensics, *in* 'Proceedings of the 26th Annual Computer Security Applications Conference', pp. 307–316. 28

Cheminod, M., Durante, L., Seno, L. and Valenzano, A. (2017), 'Detection of attacks based on known vulnerabilities in industrial networked systems', *journal of information security and applications* **34**, 153–165. 32

Chen, X. and Sankaranarayanan, S. (2017), Model predictive real-time monitoring of linear systems, *in* '2017 IEEE Real-Time Systems Symposium (RTSS)', IEEE, pp. 297–306. 40, 47, 48, 90, 124, 144

Chen, Y., Kar, S. and Moura, J. M. F. (2017), 'Dynamic attack detection in cyber-physical systems with side initial state information', *IEEE Transactions on Automatic Control* **62**(9), 4618–4624. 143

Cheung, S., Dutertre, B., Fong, M., Lindqvist, U., Skinner, K. and Valdes, A. (2007), Using model-based intrusion detection for scada networks, *in* 'Proceedings of the SCADA security scientific symposium', Vol. 46, Citeseer, pp. 1–12. 32

Chivers, H., Nobles, P., Shaikh, S. A., Clark, J. A. and Chen, H. (2009), Accumulating evidence of insider attacks, *in* 'Proceedings of the 1st International Workshop on Managing Insider Security Threats (MIST-2009)', CEUR, pp. 34–50. 43, 64, 66, 67, 142

Chou, Y., Yoon, H. and Sankaranarayanan, S. (2020), Predictive runtime monitoring of vehicle models using bayesian estimation and reachability analysis, *in* 'Intl. Conference on Intelligent Robots and Systems (IROS)'. 48, 115, 143

Coletta, A. (2018), Predictive detection of known security criticalities in cyber physical systems with unobservable variables, *in* '11th international conference on security and its applications (cnsa)', pp. 61–77. 4, 5, 9, 44, 49, 50, 63, 75

Conti, M., Donadel, D. and Turrin, F. (2021), 'A survey on industrial control system testbeds and datasets for security research', *arXiv preprint arXiv:2102.05631* . 1, 3, 23, 33

Cruz, T., Rosa, L., Proença, J., Maglaras, L., Aubigny, M., Lev, L., Jiang, J. and Simoes, P. (2016), 'A cybersecurity detection framework for supervisory control and data acquisition systems', *IEEE Transactions on Industrial Informatics* **12**(6), 2236–2246. 143

Cárdenas, A. A., Amin, S., Lin, Z.-S., Huang, Y.-L., Huang, C.-Y. and Sastry, S. (2011), Attacks against process control systems: risk assessment, detection, and response, *in* 'Proceedings of the 6th ACM symposium on information, computer and communications security', ACM, pp. 355–366. 9, 36, 37, 59, 111

Cárdenas, A. A., Amin, S. and Sastry, S. (2008), Research challenges for the security of control systems, *in* 'HotSec'. 21, 22, 23, 32

Dai, L. and Pelckmans, K. (2012), An ellipsoid based, two-stage screening test for bpdn, *in* '2012 Proceedings of the 20th European Signal Processing Conference (EUSIPCO)', IEEE, pp. 654–658. 8, 75, 87, 91, 140

Di Pietro, A., Panzieri, S. and Gasparri, A. (2015), Situational awareness using distributed data fusion with evidence discounting, *in* 'International Conference on Critical Infrastructure Protection', Springer, pp. 281–296. 143

Di Pinto, A., Dragoni, Y. and Carcano, A. (2018), Triton: The first ics cyber attack on safety instrument systems, *in* 'Proc. Black Hat USA', Vol. 2018, pp. 1–26. 1

Downs, J. J. and Vogel, E. F. (1993), 'A plant-wide industrial process control problem', *Computers and chemical engineering* **17**(3), 245–255. ix, 9, 59, 101, 103, 111, 114, 141

Eden, P., Blyth, A., Burnap, P., Cherdantseva, Y., Jones, K., Soulsby, H. and Stoddart, K. (2016), Forensic readiness for scada/ics incident response, *in* 'Proceedings of the 4th International Symposium for ICS and SCADA Cyber Security Research 2016', BCS Learning and Development Ltd., pp. 1–9. ix, 3, 27, 29, 97, 99, 126, 134, 137

Elhoseny, M., Abbas, H., Hassanien, A. E., Muhammad, K. and Sangaiah, A. K. (2017), 'Secure automated forensic investigation for sustainable critical infrastructures compliant with green computing requirements', *IEEE Transactions on Sustainable Computing* **5**(2), 174–191. 28

Elyas, M., Ahmad, A., Maynard, S. B. and Lonie, A. (2015), 'Digital forensic readiness: Expert perspectives on a theoretical framework', *Computers & Security* **52**, 70–89. 26

Erba, A. and Tippenhauer, N. O. (2020), 'No need to know physics: Resilience of process-based model-free anomaly detection for industrial control systems', *arXiv preprint arXiv:2012.03586* . 6, 40, 63

Esfahani, P. M., Vrakopoulou, M., Margellos, K., Lygeros, J. and Andersson, G. (2010), Cyber attack in a two-area power system: Impact identification using reachability, *in* 'Proceedings of the 2010 American Control Conference', IEEE, pp. 962–967. 40, 49, 86

Etigowni, S., Hossain-McKenzie, S., Kazerooni, M., Davis, K. and Zonouz, S. (2018), Crystal (ball) i look at physics and predict control flow! just-ahead-of-time controller recovery, *in* 'Proceedings of the 34th Annual Computer Security Applications Conference', pp. 553–565. 9, 44, 49, 63, 75

Fang, S., Johansson, K. H., Skoglund, M., Sandberg, H. and Ishii, H. (2019), 'Two-way coding in control systems under injection attacks: From attack detection to attack correction', *arXiv preprint arXiv:1901.05420* . 38

Farooqui, A. A., Zaidi, S. S. H., Memon, A. Y. and Qazi, S. (2014), Cyber security backdrop: A scada testbed, *in* '2014 IEEE Computers, Communications and IT Applications Conference', IEEE, pp. 98–103. 112

Fawzi, H., Tabuada, P. and Diggavi, S. (2014), 'Secure estimation and control for cyber-physical systems under adversarial attacks', *IEEE Transactions on Automatic control* **59**(6), 1454–1467. 40

Geng, Y., Wang, Y., Liu, W., Wei, Q., Liu, K. and Wu, H. (2019), A survey of industrial control system testbeds, *in* 'IOP Conference Series: Materials Science and Engineering', Vol. 569, IOP Publishing, p. 042030. 9, 59, 111

Genge, B., Rusu, D. A. and Haller, P. (2014), A connection pattern-based approach to detect network traffic anomalies in critical infrastructures, *in* 'Proceedings of the Seventh European Workshop on System Security', ACM, p. 1. 32

Genge, B. and Siaterlis, C. (2014), 'Physical process resilience-aware network design for scada systems', *Computers and Electrical Engineering* **40**(1), 142–157. 9, 32, 59, 111

Giraldo, J., Sarkar, E., Cardenas, A. A., Maniatakos, M. and Kantarcioglu, M. (2017), 'Security and privacy in cyber-physical systems: A survey of surveys', *IEEE Design and Test* **34**(4), 7–17. 30

Giraldo, J., Urbina, D., Cardenas, A., Valente, J., Faisal, M., Ruths, J., Tippenhauer, N. O., Sandberg, H. and Candell, R. (2018), 'A survey of physics-based attack detection in cyber-physical systems', *ACM Computing Surveys (CSUR)* **51**(4), 76. 3, 5, 33, 35, 39, 123

Graveto, V., Rosa, L., Cruz, T. and Simões, P. (2019), 'A stealth monitoring mechanism for cyber-physical systems', *International Journal of Critical Infrastructure Protection* **24**, 126–143. 131

Griffioen, P., Weerakkody, S. and Sinopoli, B. (2019), 'A moving target defense for securing cyber-physical systems', *arXiv preprint arXiv:1902.01423* . 38

Griffioen, P., Weerakkody, S., Sinopoli, B., Ozel, O. and Mo, Y. (2019), A tutorial on detecting security attacks on cyber-physical systems, *in* '2019 18th European Control Conference (ECC)', IEEE, pp. 979–984. 38

Grispos, G., Glisson, W. B. and Choo, K.-K. R. (2017), Medical cyber-physical systems development: A forensics-driven approach, *in* '2017 IEEE/ACM International Conference on Connected Health: Applications, Systems and Engineering Technologies (CHASE)', IEEE, pp. 108–113. 29

Grobauer, B., Mehlau, J. I. and Sander, J. (2006), 'Carmentis: A co-operative approach towards situation awareness and early warning for the internet', *IT-Incident Management and IT-Forensics-IMF 2006* . 43

Guo, Z., Shi, D., Johansson, K. H. and Shi, L. (2018), 'Worst-case stealthy innovation-based linear attack on remote state estimation', *Automatica* **89**, 117–124. 81

Gupta, A., Langbort, C. and Başar, T. (2010), Optimal control in the presence of an intelligent jammer with limited actions, *in* '49th IEEE Conference on Decision and Control (CDC)', IEEE, pp. 1096–1101. 39

Gupta, R. A. and Chow, M.-Y. (2009), 'Networked control system: Overview and research trends', *IEEE transactions on industrial electronics* **57**(7), 2527–2535. 113

Hadžiosmanović, D., Simionato, L., Bolzoni, D., Zambon, E. and Etalle, S. (2012), N-gram against the machine: On the feasibility of the n-gram network analysis for binary protocols, *in* 'International Workshop on Recent Advances in Intrusion Detection', Springer, pp. 354–373. 6, 32

Hadžiosmanović, D., Bolzoni, D. and Hartel, P. H. (2012), 'A log mining approach for process monitoring in scada', *International Journal of Information Security* **11**(4), 231–251. 28

Hammad, E. M., Farraj, A. K. and Kundur, D. (2015), A resilient feedback linearization control scheme for smart grids under cyber-physical disturbances, *in* '2015 IEEE Power and Energy Society Innovative Smart Grid Technologies Conference (ISGT)', IEEE, pp. 1–5. 39

Hashemi, N., Murguia, C. and Ruths, J. (2018), A comparison of stealthy sensor attacks on control systems, *in* '2018 Annual American Control Conference (ACC)', IEEE, pp. 973–979. 40, 49, 59, 80, 81, 82

Hendrickx, J. M., Johansson, K. H., Jungers, R. M., Sandberg, H. and Sou, K. C. (2014), 'Efficient computations of a security index for false data attacks in power networks', *IEEE Transactions on Automatic Control* **59**(12), 3194–3208. 41

Hoehn, A. and Zhang, P. (2016), Detection of covert attacks and zero dynamics attacks in cyber-physical systems, *in* '2016 American Control Conference (ACC)', IEEE, pp. 302–307. 38

Hu, H., Fazlyab, M., Morari, M. and Pappas, G. J. (2020), 'Reach-sdp: Reachability analysis of closed-loop systems with neural network controllers via semidefinite programming', *arXiv preprint arXiv:2004.07876* . 75

Huang, Y.-L., Cárdenas, A. A., Amin, S., Lin, Z.-S., Tsai, H.-Y. and Sastry, S. (2009), 'Understanding the physical and economic consequences of attacks on control systems', *International Journal of Critical Infrastructure Protection* **2**(3), 73–83. 69

Husák, M., Komárková, J., Bou-Harb, E. and Čeleda, P. (2018), 'Survey of attack projection, prediction, and forecasting in cyber security', *IEEE Communications Surveys & Tutorials* **21**(1), 640–660. 124

Hwang, I., Kim, S., Kim, Y. and Seah, C. E. (2009), 'A survey of fault detection, isolation, and reconfiguration methods', *IEEE transactions on control systems technology* **18**(3), 636–653. 35

Janicke, H., Nicholson, A., Webber, S. and Cau, A. (2015), 'Runtime-monitoring for industrial control systems', *Electronics* **4**(4), 995–1017. 2, 5, 28, 63, 98, 99

Johnson, T. T., Bak, S., Caccamo, M. and Sha, L. (2016), 'Real-time reachability for verified simplex design', *ACM Transactions on Embedded Computing Systems (TECS)* **15**(2), 26. 48

Junejo, K. N. and Goh, J. (2016), Behaviour-based attack detection and classification in cyber physical systems using machine learning, *in* 'Proceedings of the 2nd ACM International Workshop on Cyber-Physical System Security', ACM, pp. 34–43. 40

Kalaivani, C. and Kalaiarasi, N. (2019), 'Earliest deadline first scheduling technique for different networks in network control system', *Neural Computing and Applications* **31**(1), 223–232. 112

Kalutarage, H. K., Lee, C., Shaikh, S. A. and Sung, F. L. B. (2015), Towards an early warning system for network attacks using bayesian inference, *in* '2015 IEEE 2nd International Conference on Cyber Security and Cloud Computing', IEEE, pp. 399–404. 43, 67, 124, 126

Kalutarage, H. K., Shaikh, S. A., Zhou, Q. and James, A. E. (2012), Sensing for suspicion at scale: A bayesian approach for cyber conflict attribution and reasoning, *in* '2012 4th International Conference on Cyber Conflict (CYCON 2012)', IEEE, pp. 1–19. 44, 66, 124, 142

Kalutarage, H. K., Shaikh, S. A., Zhou, Q. and James, A. E. (2013), How do we effectively monitor for slow suspicious activities?, *in* 'ESSoS Doctoral Symposium 2013', Citeseer, p. 36. 43, 126

Kalutarage, H., Shaikh, S., Lee, B.-S., Lee, C. and Kiat, Y. C. (2015), Early warning systems for cyber defence, *in* 'International Workshop on Open Problems in Network Security', Springer, pp. 29–42. 4

Kaouk, M., Flaus, J.-M., Potet, M.-L. and Groz, R. (2019), A review of intrusion detection systems for industrial control systems, *in* '2019 6th International Conference on Control, Decision and Information Technologies (CoDIT)', IEEE, pp. 1699–1704. 3, 21, 22, 24

Keller, J.-Y. and Sauter, D. (2013), Monitoring of stealthy attack in networked control systems, *in* '2013 Conference on Control and Fault-Tolerant Systems (SysTol)', IEEE, pp. 462–467. 38

Kilpatrick, T., Gonzalez, J., Chandia, R., Papa, M. and Shenoi, S. (2006), An architecture for scada network forensics, *in* 'IFIP International Conference on Digital Forensics', Springer, pp. 273–285. 28

Kiss, I., Genge, B., Haller, P. and Sebestyén, G. (2014), Data clustering-based anomaly detection in industrial control systems, *in* '2014 IEEE 10th International Conference on Intelligent Computer Communication and Processing (ICCP)', IEEE, pp. 275–281. 2, 30, 96

Kravchik, M. and Shabtai, A. (2019), 'Efficient cyber attacks detection in industrial control systems using lightweight neural networks', *arXiv preprint arXiv:1907.01216* . 40

Krotofil, M. and Cárdenas, A. A. (2013), Resilience of process control systems to cyber-physical attacks, *in* 'Nordic Conference on Secure IT Systems', Springer, pp. 166–182. 9, 59, 111

Krotofil, M., Larsen, J. and Gollmann, D. (2015), The process matters: Ensuring data veracity in cyber-physical systems, *in* 'Proceedings of the 10th ACM Symposium on Information, Computer and Communications Security', pp. 133–144. 62

Kurzhanski, A. B. and Varaiya, P. (2000), Ellipsoidal techniques for reachability analysis, *in* 'International Workshop on Hybrid Systems: Computation and Control', Springer, pp. 202–214. 8, 47, 75

Kurzhanskiy, A. A. and Varaiya, P. (2006), Ellipsoidal toolbox (et), *in* 'Proceedings of the 45th IEEE Conference on Decision and Control', IEEE, pp. 1498–1503. 85, 87, 113, 140

Kwon, C. and Hwang, I. (2016*a*), 'Cyber attack mitigation for cyber–physical systems: hybrid system approach to controller design', *IET Control Theory and Applications* **10**(7), 731–741. 49

Kwon, C. and Hwang, I. (2016*b*), Recursive reachable set computation for on-line safety assessment of the cyber-physical system against stealthy cyber attacks, *in* '2016 54th Annual Allerton Conference on Communication, Control, and Computing (Allerton)', IEEE, pp. 1123–1128. 49

Kwon, C. and Hwang, I. (2018), 'Reachability analysis for safety assurance of cyber-physical systems against cyber attacks', *IEEE Transactions on Automatic Control* **63**(7), 2272–2279. 4, 5, 49, 69

Kwon, C., Liu, W. and Hwang, I. (2013), Security analysis for cyber-physical systems against stealthy deception attacks, *in* ‘American Control Conference (ACC), 2013’, IEEE, pp. 3344–3349. 49, 79

Lakshminarayana, S., Karachiwala, J. S., Teng, T. Z., Tan, R. and Yau, D. K. Y. (2019), ‘Performance and resilience of cyber-physical control systems with reactive attack mitigation’, *IEEE Transactions on Smart Grid* . 37

Le Guernic, C. (2009), Reachability analysis of hybrid systems with linear continuous dynamics, PhD thesis. 46, 75

Le Guernic, C. and Girard, A. (2009), Reachability analysis of hybrid systems using support functions, *in* ‘International Conference on Computer Aided Verification’, Springer, pp. 540–554. 47

Lee, R., Assante, M. and Conway, T. (2014), *SANS ICS Defense Use Case (DUC) Dec 30 2014: ICS CP/PE case study paper-German Steel Mill Cyber Attack*, SANS ICS. 1, 60, 61

Li, D. and Martínez, S. (2020), ‘High-confidence attack detection via wasserstein-metric computations’, *arXiv preprint arXiv:2003.07880* . 36

Li, F., Mikulski, D., Wagner, J. R. and Wang, Y. (2019), ‘Trust-based control and scheduling for ugv platoon under cyber attacks’, *SAE Technical Paper* pp. 01–1077. 39

Li, T., Tan, F., Wang, Q., Bu, L., Cao, J.-N. and Liu, X. (2013), ‘From offline toward real time: A hybrid systems model checking and cps codesign approach for medical device plug-and-play collaborations’, *IEEE Transactions on Parallel and Distributed Systems* **25**(3), 642–652. 47, 48

Li, Y.-G. and Yang, G.-H. (2019), ‘Optimal stealthy false data injection attacks in cyber-physical systems’, *Information Sciences* . 37

Liu, H., Niu, B. and Qin, J. (2021), ‘Reachability analysis for linear discrete-time systems under stealthy cyber attacks’, *IEEE Transactions on Automatic Control* . 40, 49

Liu, J., Yin, L., Hu, Y., Lv, S. and Sun, L. (2018), A novel intrusion detection algorithm for industrial control systems based on cnn and process state transition, *in* ‘2018

IEEE 37th International Performance Computing and Communications Conference (IPCCC)', IEEE, pp. 1–8. 32

Liu, Y., Ning, P. and Reiter, M. K. (2011), 'False data injection attacks against state estimation in electric power grids', *ACM Transactions on Information and System Security (TISSEC)* **14**(1), 13. 37

Loukas, G. (2015), *Cyber-physical attacks: A growing invisible threat*, Butterworth-Heinemann. 1

Lun, Y. Z., D'Innocenzo, A., Smarra, F., Malavolta, I. and Di Benedetto, M. D. (2019), 'State of the art of cyber-physical systems security: An automatic control perspective', *Journal of Systems and Software* **149**, 174–216. 3, 6, 33, 71

Mahmoud, M. S., Hamdan, M. M. and Barudi, U. A. (2019), 'Modeling and control of cyber-physical systems subject to cyber attacks: A survey of recent advances and challenges', *Neurocomputing* . 33

Marquis, V., Ho, R., Rainey, W., Kimpel, M., Ghiorzi, J., Cricchi, W. and Bezzo, N. (2018), Toward attack-resilient state estimation and control of autonomous cyber-physical systems, *in* '2018 Systems and Information Engineering Design Symposium (SIEDS)', IEEE, pp. 70–75. 38

Mashima, D. and Cárdenas, A. A. (2012), Evaluating electricity theft detectors in smart grid networks, *in* 'International Workshop on Recent Advances in Intrusion Detection', Springer, pp. 210–229. 40

McEvoy, T. R. and Wolthusen, S. D. (2010), A formal adversary capability model for scada environments, *in* 'International Workshop on Critical Information Infrastructures Security', Springer, pp. 93–103. 32

McEvoy, T. R. and Wolthusen, S. D. (2011*a*), 'Defeating network node subversion on scada systems using probabilistic packet observation', *International Journal of Critical Infrastructures* **9**, 32–51. 32

McEvoy, T. and Wolthusen, S. (2011*b*), A plant-wide industrial process control security problem, *in* 'International Conference on Critical Infrastructure Protection', Springer, pp. 47–56. 9, 59, 111

Mercaldo, F., Martinelli, F. and Santone, A. (2019), Real-time scada attack detection by means of formal methods, *in* '2019 IEEE 28th International Conference on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE)', IEEE, pp. 231–236. 32

Milosevic, J., Sandberg, H. and Johansson, K. H. (2018), 'A security index for actuators based on perfect undetectability: Properties and approximation', *arXiv preprint arXiv:1807.04069* . 41

Milošević, J., Umsonst, D., Sandberg, H. and Johansson, K. H. (2018), Quantifying the impact of cyber-attack strategies for control systems equipped with an anomaly detector, *in* '2018 European Control Conference (ECC)', IEEE, pp. 331–337. 40, 49

Mishra, S. (2019), *Forensic Investigation Framework for Complex Cyber Attack on Cyber Physical System by Using Goals/Sub-goals of an Attack and Epidemics of Malware in a System*, Springer, pp. 491–504. 28

Mitchell, R. and Chen, I.-R. (2014), 'A survey of intrusion detection techniques for cyber-physical systems', *ACM Comput. Surv.* **46**(4), 55:1–55:29.
URL: *http://doi.acm.org/10.1145/2542049* 32

Mo, Y., Chabukswar, R. and Sinopoli, B. (2014), 'Detecting integrity attacks on scada systems', *IEEE Transactions on Control Systems Technology* **22**(4), 1396–1407. 36, 38

Mo, Y. and Sinopoli, B. (2009), Secure control against replay attacks, *in* '2009 47th Annual Allerton Conference on Communication, Control, and Computing (Allerton)', IEEE, pp. 911–918. 35

Mo, Y., Weerakkody, S. and Sinopoli, B. (2015), 'Physical authentication of control systems: Designing watermarked control inputs to detect counterfeit sensor outputs', *IEEE Control Systems Magazine* **35**(1), 93–109. 38

Mohamed, N., Al-Jaroodi, J. and Jawhar, I. (2020), 'Cyber–physical systems forensics: Today and tomorrow', *Journal of Sensor and Actuator Networks* **9**(3), 37. 3, 27, 29, 63

Mousavinejad, E., Ge, X., Han, Q.-L., Yang, F. and Vlacic, L. (2019), 'Resilient tracking control of networked control systems under cyber attacks', *IEEE transactions on cybernetics* . 34, 39

Murguia, C. and Ruths, J. (2016), Cusum and chi-squared attack detection of compromised sensors, *in* '2016 IEEE Conference on Control Applications (CCA)', IEEE, pp. 474–480. 36

Murguia, C. and Ruths, J. (2018), On reachable sets of hidden cps sensor attacks, *in* '2018 Annual American Control Conference (ACC)', IEEE, pp. 178–184. 8, 38, 49, 61, 69, 79, 82, 83, 131, 140

Murguia, C. and Ruths, J. (2019), 'On model-based detectors for linear time-invariant stochastic systems under sensor attacks', *IET Control Theory and Applications* . 59, 81

Murguia, C., Shames, I., Ruths, J. and Nesic, D. (2018), 'Security metrics of networked control systems under sensor attacks (extended preprint)', *arXiv preprint arXiv:1809.01808* . 3, 38, 49, 86

Murguia, C., van de Wouw, N. and Ruths, J. (2017), 'Reachable sets of hidden cps sensor attacks: Analysis and synthesis tools', *IFAC-PapersOnLine* **50**(1), 2088–2094. 38, 40, 49

Myers, D., Radke, K., Suriadi, S. and Foo, E. (2017), Process discovery for industrial control system cyber attack detection, *in* 'IFIP International Conference on ICT Systems Security and Privacy Protection', Springer, pp. 61–75. 28

Niu, H., Bhowmick, C. and Jagannathan, S. (2019), 'Attack detection and approximation in nonlinear networked control systems using neural networks', *IEEE transactions on neural networks and learning systems* . 32

Palleti, V. R., Tan, Y. C. and Samavedham, L. (2018), 'A mechanistic fault detection and isolation approach using kalman filter to improve the security of cyber physical systems', *Journal of Process Control* **68**, 160–170. 37

Pan, K., Palensky, P. and Esfahani, P. M. (2019), 'From static to dynamic anomaly detection with application to power system cyber security', *arXiv preprint arXiv:1904.09137* . 37

Park, G., Lee, C., Shim, H., Eun, Y. and Johansson, K. H. (2019), 'Stealthy adversaries against uncertain cyber-physical systems: Threat of robust zero-dynamics attack', *IEEE Transactions on Automatic Control* . 37

Park, J., Ivanov, R., Weimer, J., Pajic, M., Son, S. H. and Lee, I. (2017), 'Security of cyber-physical systems in the presence of transient sensor faults', *ACM Transactions on Cyber-Physical Systems* **1**(3), 1–23. 39, 76

Pasquale, L., Alrajeh, D., Peersman, C., Tun, T., Nuseibeh, B. and Rashid, A. (2018), Towards forensic-ready software systems, *in* '2018 IEEE/ACM 40th International Conference on Software Engineering: New Ideas and Emerging Technologies Results (ICSE-NIER)', IEEE, pp. 9–12. 3, 25, 26, 30, 58, 96, 97

Pasquale, L., Hanvey, S., Mcgloin, M. and Nuseibeh, B. (2016), 'Adaptive evidence collection in the cloud using attack scenarios', *Computers & Security* **59**, 236–254. 26

Pasquale, L., Yu, Y., Salehie, M., Cavallaro, L., Tun, T. T. and Nuseibeh, B. (2013), Requirements-driven adaptive digital forensics, *in* 'Requirements Engineering Conference (RE), 2013 21st IEEE International', IEEE, pp. 340–341. 25

Pasqualetti, F., Dorfler, F. and Bullo, F. (2015), 'Control-theoretic methods for cyberphysical security: Geometric principles for optimal cross-layer resilient control systems', *IEEE Control Systems* **35**(1), 110–127. 3, 37, 38

Pasqualetti, F., Dörfler, F. and Bullo, F. (2013), 'Attack detection and identification in cyber-physical systems', *IEEE Transactions on Automatic Control* **58**(11), 2715–2729. 29, 37, 38

Penteado, F. D. and Ciric, A. R. (1996), 'An minlp approach for safe process plant layout', *Industrial & Engineering Chemistry Research* **35**(4), 1354–1361.
URL: *https://doi.org/10.1021/ie9502547* 99

Peters, M. S., Timmerhaus, K. D., West, R. E. et al. (2003), *Plant design and economics for chemical engineers*, Vol. 4, McGraw-Hill New York. 106, 144

Polisetty, V. G., Varanasi, S. K. and Jampana, P. (2019), 'Error bounds for identification of a class of continuous lti systems', *IFAC-PapersOnLine* **52**(1), 418 – 423. 12th IFAC Symposium on Dynamics and Control of Process Systems, including Biosystems DYCOPS 2019.
URL: *http://www.sciencedirect.com/science/article/pii/S2405896319301843* 72

Qadeer, R., Murguia, C., Ahmed, C. M. and Ruths, J. (2017), *Multistage downstream attack detection in a cyber physical system*, Springer, pp. 177–185. 37

Qasim, S. A., Smith, J. M. and Ahmed, I. (2020), 'Control logic forensics framework using built-in decompiler of engineering software in industrial control systems', *Forensic Science International: Digital Investigation* **33**, 301013. 28

Quiroz-Pérez, E., de Lira-Flores, J. A., Gutiérrez-Antonio, C. and Vazquez-Román, R. (2021), 'A new multiple-risk map approach to solve process plant layout considering safety and economic aspects', *Journal of Loss Prevention in the Process Industries* **72**, 104524. 8, 57, 96, 99, 100, 140

Ramaki, A. A. and Atani, R. E. (2016), 'A survey of it early warning systems: architectures, challenges, and solutions', *Security and Communication Networks* **9**(17), 4751–4776. 4, 43, 44

Rashidinejad, A., Wetzels, B., Reniers, M., Lin, L., Zhu, Y. and Su, R. (2019), Supervisory control of discrete-event systems under attacks: an overview and outlook, *in* '2019 18th European Control Conference (ECC)', IEEE, pp. 1732–1739. 33

Ricker, N. L. (1993), 'Model predictive control of a continuous, nonlinear, two-phase reactor', *Journal of Process Control* **3**(2), 109–123. 111

Ricker, N. L. (1996), 'Decentralized control of the tennessee eastman challenge process', *Journal of Process Control* **6**(4), 205–221. 9, 61, 103, 111, 132, 135

Rowlingson, R. et al. (2004), 'A Ten Step Process for Forensic Readiness', *International Journal of Digital Evidence* **2**(3), 1–28. 2, 25, 26

Salvador, J. R., Munoz de la Pena, D., Ramirez, D. and Alamo, T. (2020), 'Predictive control of a water distribution system based on process historian data', *Optimal Control Applications and Methods* **41**(2), 571–586. 63

Sandberg, H., Teixeira, A. and Johansson, K. H. (2010), On security indices for state estimators in power networks, *in* 'First Workshop on Secure Control Systems (SCS), Stockholm, 2010'. 41

Sargolzaei, A., Yazdani, K., Abbaspour, A. R., Crane, C. D. and Dixon, W. (2019), 'Detection and mitigation of false data injection attacks in networked control systems', *IEEE Transactions on Industrial Informatics* . 37, 40

Satchidanandan, B. and Kumar, P. R. (2017), 'Dynamic watermarking: Active defense of networked cyber–physical systems', *Proceedings of the IEEE* **105**(2), 219–240. 38

Sayegh, N., Elhajj, I. H., Kayssi, A. and Chehab, A. (2014), Scada intrusion detection system based on temporal behavior of frequent patterns, *in* 'MELECON 2014-2014 17th IEEE Mediterranean Electrotechnical Conference', IEEE, pp. 432–438. 32

Shields, C., Frieder, O. and Maloof, M. (2011), 'A system for the proactive, continuous, and efficient collection of digital forensic evidence', *Digital investigation* **8**, S3–S13. 26

Slay, J. and Miller, M. (2007), Lessons learned from the maroochy water breach, *in* 'International Conference on Critical Infrastructure Protection', Springer, pp. 73–82. 1

Smith, R. S. (2015), 'Covert misappropriation of networked control systems: Presenting a feedback structure', *IEEE Control Systems Magazine* **35**(1), 82–92. 37, 79, 143

Subbe, C. P., Kruger, M., Rutherford, P. and Gemmel, L. (2001), 'Validation of a modified early warning score in medical admissions', *Qjm* **94**(10), 521–526. 42

Sun, Y.-C. and Yang, G.-H. (2019), 'Robust event-triggered model predictive control for cyber-physical systems under denial-of-service attacks', *International Journal of Robust and Nonlinear Control* . 39

Symantec, I. (2014), 'Internet security threat report', *Mountain View, CA:[sn]* . 44

Sánchez, H. S., Rotondo, D., Escobet, T., Puig, V. and Quevedo, J. (2019), 'Bibliographical review on cyber attacks from a control oriented perspective', *Annual Reviews in Control* . 3, 33

Taveras, P. (2013), 'Scada live forensics: real time data acquisition process to detect, prevent or evaluate critical situations', *European Scientific Journal* **9**(21). 3, 5, 27, 28, 29, 56, 63

Teixeira, A., Sandberg, H. and Johansson, K. H. (2015), Strategic stealthy attacks: the output-to-output l2-gain, *in* '2015 54th IEEE Conference on Decision and Control (CDC)', IEEE, pp. 2582–2587. 37

Teixeira, A., Shames, I., Sandberg, H. and Johansson, K. H. (2012), Revealing stealthy attacks in control systems, *in* '50th Annual Allerton Conference on Communication, Control, and Computing, Allerton, IL, USA, October 01-05, 2012', IEEE conference proceedings, pp. 1806–1813. 38

Teixeira, A., Shames, I., Sandberg, H. and Johansson, K. H. (2015), 'A secure control framework for resource-limited adversaries', *Automatica* **51**, 135–148. 35, 36, 69, 79, 80

Teixeira, A., Sou, K. C., Sandberg, H. and Johansson, K. H. (2013), *Quantifying cyber-security for networked control systems*, Springer, pp. 123–142. 41

Teixeira, A., Sou, K. C., Sandberg, H. and Johansson, K. H. (2015), 'Secure control systems: A quantitative risk management approach', *IEEE Control Systems Magazine* **35**(1), 24–45. 41

Theilmann, A. (2010), *Beyond centralism: The herold approach to sensor networks and early warning systems*, Universitätsbibliothek Dortmund. 44, 142

Tian, J., Tan, R., Guan, X., Xu, Z. and Liu, T. (2019), 'Moving target defense approach to detecting stuxnet-like attacks', *IEEE Transactions on Smart Grid* . 38

Tran, H.-D., Nguyen, L. V., Musau, P., Xiang, W. and Johnson, T. T. (2019), Decentralized real-time safety verification for distributed cyber-physical systems, *in* 'International Conference on Formal Techniques for Distributed Objects, Components, and Systems', Springer, pp. 261–277. 48, 143

Umsonst, D. and Sandberg, H. (2018), Anomaly detector metrics for sensor data attacks in control systems, *in* '2018 Annual American Control Conference (ACC)', IEEE, pp. 153–158. 36

Urbina, D. I., Giraldo, J. A., Cardenas, A. A., Tippenhauer, N. O., Valente, J., Faisal, M., Ruths, J., Candell, R. and Sandberg, H. (2016), Limiting the impact of stealthy attacks on industrial control systems, *in* 'Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security', ACM, pp. 1092–1105. 3, 35, 37, 39, 79

van der Knijff, R. M. (2014), 'Control systems/scada forensics, what's the difference?', *Digital Investigation* **11**(3), 160–174. ix, 2, 3, 27, 29, 97, 99

van Vliet, P., Kechadi, M. T. and Le-Khac, N.-A. (2015), *Forensics in industrial control system: a case study*, Springer, pp. 147–156. 2, 27, 28, 29, 63

Wang, Z., Xu, Q. and Ho, T. C. (2014), 'Emission source characterization during an ethylene plant shutdown', *Chemical Engineering & Technology* **37**(7), 1170–1180. 27

Weerakkody, S., Mo, Y. and Sinopoli, B. (2014), Detecting integrity attacks on control systems using robust physical watermarking, *in* '53rd IEEE Conference on Decision and Control', IEEE, pp. 3757–3764. 38

Weerakkody, S., Ozel, O., Griffioen, P. and Sinopoli, B. (2017), Active detection for exposing intelligent attacks in control systems, *in* '2017 IEEE Conference on Control Technology and Applications (CCTA)', IEEE, pp. 1306–1312. 38

Weerakkody, S. and Sinopoli, B. (2015), Detecting integrity attacks on control systems using a moving target approach, *in* '2015 54th IEEE Conference on Decision and Control (CDC)', IEEE, pp. 5820–5826. 38

Wu, T., Disso, J. F. P., Jones, K. and Campos, A. (2013), Towards a scada forensics architecture, *in* 'Proceedings of the 1st International Symposium for ICS and SCADA Cyber Security Research', Vol. 12. 28

Yau, K. and Chow, K.-P. (2015), 'Plc forensics based on control program logic change detection', *Journal of Digital Forensics, Security and Law* **10**(4), 5. 2, 11, 98, 99, 135, 138, 142

Yau, K., Chow, K.-P. and Yiu, S.-M. (2018), A forensic logging system for siemens programmable logic controllers, *in* 'IFIP International Conference on Digital Forensics', Springer, pp. 331–349. 28, 58, 98, 132

Zhang, F., Kodituwakku, H. A. D. E., Hines, W. and Coble, J. B. (2019), 'Multi-layer data-driven cyber-attack detection system for industrial control systems based on network, system and process data', *IEEE Transactions on Industrial Informatics* . 143

Zhang, Q., Liu, K., Pang, Z., Xia, Y. and Liu, T. (2020), 'Reachability analysis of cyber-physical systems under stealthy attacks', *IEEE Transactions on Cybernetics* . 49

Zhang, X.-G. and Yang, G. (2021), 'Kullback-leibler divergence-based attacks against remote state estimation in cyber-physical systems', *IEEE Transactions on Industrial Electronics* . 81

Zhang, X.-G., Yang, G.-H. and Wasly, S. (2021), 'Man-in-the-middle attack against cyber-physical systems under random access protocol', *Information Sciences* **576**, 708–724. 81

Zhou, H., Taal, A., Koulouzis, S., Wang, J., Hu, Y., Suciu, G., Poenaru, V., de Laat, C. and Zhao, Z. (2018), Dynamic real-time infrastructure planning and deployment for disaster early warning systems, *in* 'International conference on computational Science', Springer, pp. 644–654. 42

Zhu, B., Joseph, A. and Sastry, S. (2011), A taxonomy of cyber attacks on scada systems, *in* '2011 International conference on internet of things and 4th international conference on cyber, physical and social computing', IEEE, pp. 380–388. 21, 22, 23

Zhu, M. and Martinez, S. (2014), 'On the performance analysis of resilient networked control systems under replay attacks', *IEEE Transactions on Automatic Control* **59**(3), 804–808. 40

Zhu, Q. and Başar, T. (2011), Robust and resilient control design for cyber-physical systems with an application to power systems, *in* '2011 50th IEEE Conference on Decision and Control and European Control Conference', IEEE, pp. 4066–4071. 22, 24, 37, 40, 48, 143