Digital Twin-Based Cyber-Attack Detection Framework for Cyber-Physical Manufacturing Systems

Efe C. Balta, Michael Pease, James Moyne, Kira Barton, Dawn M. Tilbury

Prensented by: 조병현 (Byung Hyun Jo)

Seoul National University of Science and Technology, Seoul, Korea

UCSLAB
Ubiquitous Computing & Security Laboratory

CONTENTS

1. INTRODUCTION

2. PRELIMINARIES AND PROBLEM STATEMENT

3. PROPOSED DT-Based METHODOLOGY

4. The CYBERSECURITY DT

Introduction

- SMART manufacturing (SM) is an increasingly important paradigm that promotes the use of run-time and historical data collected via onboard and additional Internet of Things (IoT) sensing in the manufacturing system to derive decisions for the plant floor [1], [2], [3], [4].
- The decisions are implemented, often in run-time, on the resources in the manufacturing system to minimize disruptions, by integrating cyber and physical systems in modern manufacturing resources, allowing them to be reconfigurable and robust in response to disturbances.
- This framework of data-enabled decision making coupled with cyber-physical manufacturing resources is commonly referred to in the industry as Cyber-Physical Manufacturing Systems (CPMS) [5], [6].
- Specifically, in the context of this work, a DT is a software replica of a physical counterpart (i.e., the physical twin), system, process, or product, and has a purpose of impacting an aspect of the physical twin and its environment in a positive way through utilizing models, data analytics, and subject matter expertise (SME) [2], [14]

Introduction

- Decision-making logic designed for the nominal conditions of a CPMS may underperform or fail to detect certain abnormalities in the system due to complex interdependencies between multiple resources in a manufacturing process [7], [8].
- Another important implication of the cyber-physical nature of CPMS is its vulnerability to cyber-attacks. As cyber components are linked to their physical counterparts, attacks that are initiated in the cyber domain may cause harm and damage to the physical manufacturing resource, product, or even the human workers that are interacting with the CPMS [9].

Introduction

- However, detecting cyber-attacks through traditional IT-based attack detection technology deployed on or in operational technology (OT) devices and environments can sometimes adversely impact OT performance or safety. Therefore, new and effective methods to monitor CPMS and detect cyber-attacks are required.
- Detecting cyber-attacks on CPMS is not a trivial task for several reasons. Systems routinely undergo faults and expected abnormalities, namely, physical degradation, anomalies.
- These anomalies may be hard to distinguish from a carefully targeted cyber-attack (e.g., one with malicious intent) as these attacks often mimic the expected anomalous behavior to deceive the decision-making logic. Furthermore, cyber-attacks may originate from non-malicious intent (e.g., mis-calibration, version mismatch, malfunction, etc.), which also causes difficulties in distinguishing them from anomalies.

Introduction

- A DT implementation consists of one or more compute resources as required to meet scalability, modularity, and maintainability requirements.
- Use of a single DT (i.e., one compute resource) for complex CPMS has been proposed [5], [12], [15] However, scalability, modularity, and maintainability of such solutions often becomes a challenge in practice.
- More recently, a framework of multiple DTs that utilize structured abstractions to improve scalability, flexibility, maintainability, and modularity of DT-based solutions has been proposed [2], [14], [16], which is the DT architecture adopted in this work.
- The DT framework presented here utilizes multiple compute resources to distribute different data collection and analysis tasks supporting the anomaly and cyber-attack detection processes in a flexible, modular, and reconfigurable fashion.

Introduction

- As DTs themselves are software entities, they may also bring along the additional burden of vulnerabilities that could compromise the physical components through cyber-attacks.
- Traditional enterprise cybersecurity control implementations are not always possible or feasible within Industrial Control Systems (ICS) network environments and improper implementations can have unintended consequences [17].
- Typically, passive monitoring capabilities for supporting threat detectionwithin ICS network environments are implemented as risk management strategies within these networks.
- However, countering the growing threats facing ICS environments requires both passive and active monitoring [17].

Introduction

- Many of the afore mentioned methods on CPS and CPMS cybersecurity from the literature are often referred to as physics-based attack detection methods (see [28] for a detailed survey).
- Most notably, the majority of the existing literature considers the cyber-attack detection problem for a CPMS with no anomalies, which is often unrealistic in practical scenarios. Additionally, most existing methods in the literature rely on threshold-checking on the residual signals, which may underperform for controlled processes with setpoint changes during transients. We propose novel approaches to overcome these challenges in our proposed framework and methods.

Introduction

- Recent work provides a methodology to detect and differentiate specific types of cyber-attacks from equipment failure [29]. The method in [29] utilizes specific models and assumptions, which may be difficult to extend and scale for a general CPMS with various types of attacks.
- Therefore, there exists an opportunity to address the afore mentioned short comings and support both manufacturing and cybersecurity automation enhancements by leveraging common technological enablers such as DT and the Industrial Internet of Things (IIoT).
- Previous research, such as [30], demonstrates techniques to utilize Industry 4.0 technologies and methodologies such as IIoT, Industrial Internet of Services (IIoS), and DTs to create smart factories and establish “Knowledge as a Service” manufacturing processes to monitor product or service quality.
- Our research builds on the previous literature and investigates utilizing cybersecurity DT technology to monitor devices and processes for abnormal conditions that could be indicators of cybersecurity events in the context of run-time controller inputs and anomalies.
- These cybersecurity DTs could be implemented to support a passive/active hybrid approach to protect the ICS environment from advanced device-level risks.

Preliminaries and problem statement

- In this section, we first present definitions and background knowledge that will be useful in further discussions. Then, we formally state our problem in the context of the introduced formal concepts.

A. Classification of Abnormality Types

To address the challenge of cyber-attack detection for a CPMS, we first present a classification of anomalies, attacks, and faults in the context of the present work. Figure 1 presents various types of attacks and anomalies for a CPMS resource.

Preliminaries and problem statement

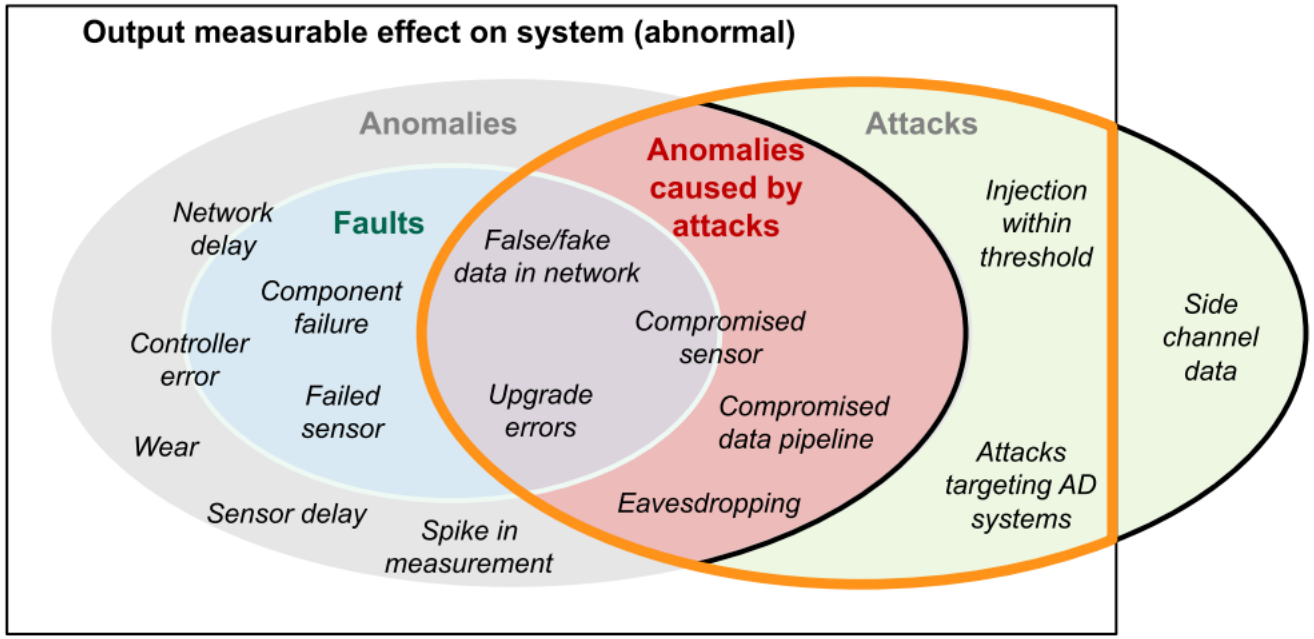

Fig. 1. Illustration of the subspaces for observable abnormalities, anomalies, faults, and attacks considered in this work. The scope of this paper is outlined with orange borders in the figure. AD: Anomaly detection.

Preliminaries and problem statement

B. Problem Statement

- The set of attacks depicted in Fig. 1 has three distinct sub-spaces; attacks that are not output measurable (e.g., sidechannel attacks), attacks that are output measurable but do not necessarily cause anomalies, and attacks that are output measurable and cause anomalies.
- Within the scope of this work, we are focusing on attacks that have output measurable effects on the system. Thus, the goal of our proposed DT is to detect the aforementioned output measurable attacks.

Preliminaries and problem statement

- *Remark 1: Within the context of cyber-attacks on CPMS, we do not necessarily require malicious intent. For example, we consider a miscalibrated sensor as a non-malicious attack.*
- Additionally, we note that the physical system is a controlled CPMS resource, thus the operational characteristics of the system may be modified by a controller. This results in transient behavior and multiple setpoint references that must be analyzed in run-time to mitigate false-positives in attack detection.
- *Remark 2: We note that our work differs from the past literature as we do not rely on a specific system model or analysis tool to provide our results. Instead, we present a general-purpose DT framework where data-driven and physics-based information about the CPMS may be utilized efficiently to detect cyber-attacks in an extensible and systematic manner.*

PROPOSED DT-Based METHODOLOGY

- In this section, we present the proposed methodology to utilize DTs for attack detection in the context of anomalies and controllers in the system.. We discuss how related methodologies from the literature can be implemented by the proposed DTs in our architecture.

A. Framwork Architecture

- Figure 2 illustrates the architecture of the controlled CPS framework with the proposed DTs considered in this work.

- To avoid confusion of terminology, we use the term process instead of system in this section (e.g., a physical system is a physical process).

Proposed DT-based methobology

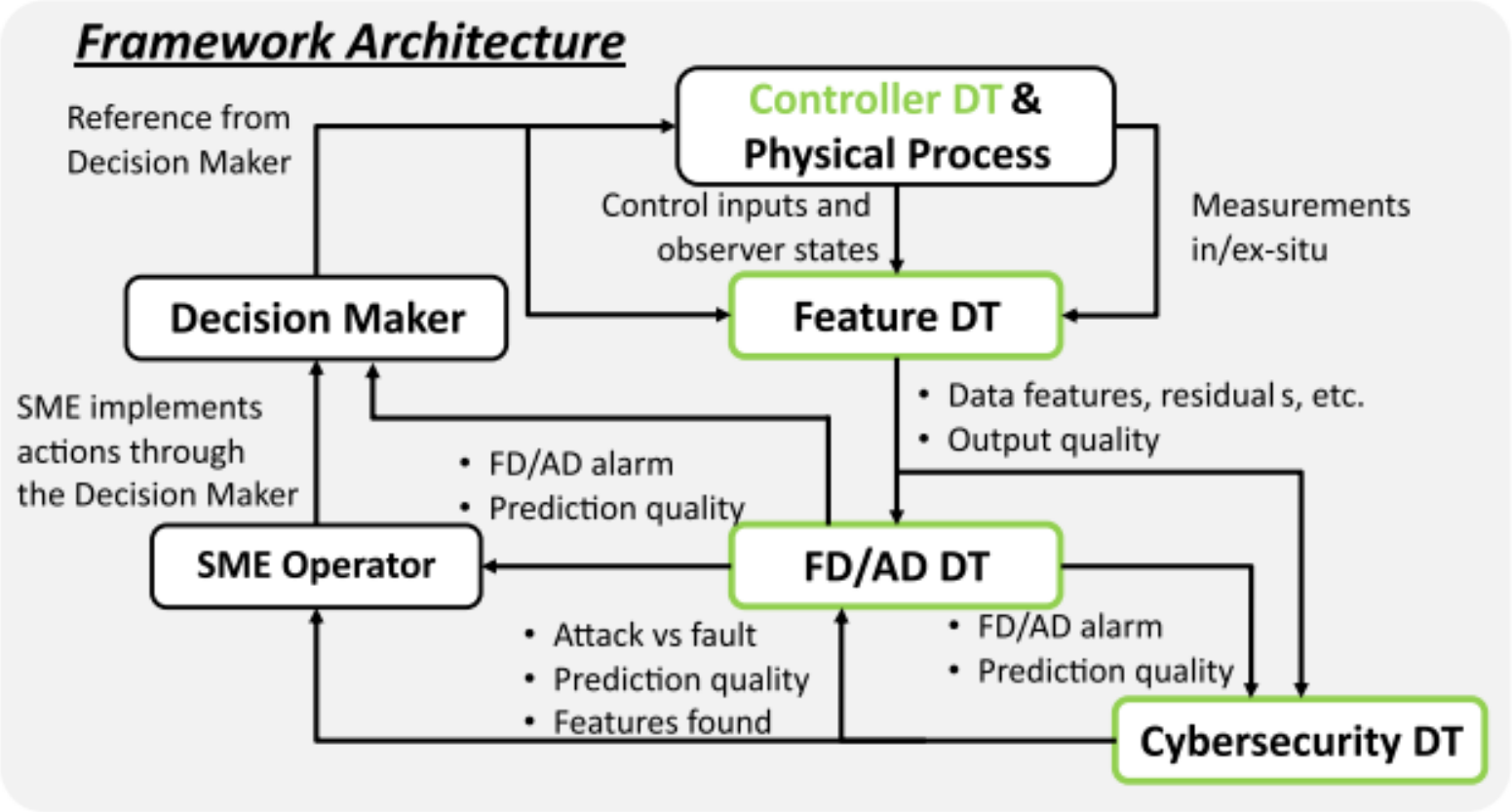

Fig. 2. The framework architecture including all the DTs and physical components. The architecture provides a basis for further extensions based on the needs on a certain physical process. The decision-maker in the architecture may be autonomous or purely advisory depending on the application domain. The color green indicates the DTs in the framework.

Proposed DT-based methobology

- The physical process may be discrete or continuous depending on the application domain. The execution of discrete manufacturing processes is often considered in terms of runs where a single unit (or batch) is manufactured. We consider the data collected during the run as in-situ and the data collected after a run is completed as ex-situ (e.g., for post-process quality control). The framework architecture presented in Fig. 2 is largely based on augmenting existing feature-based anomaly and fault detection systems in the literature (e.g., [16], [19], [32], [33], [34]).

- *Remark 3: It is important to note that the abstraction of what constitutes a DT is a design decision. We adopt the DT framework methodology instead of considering a single DT entity, e.g. [15]. However, there is no loss of generality since the proposed DTs in this work may be encapsulated by a single DT. As previously discussed, we utilize the framework approach, following [14], [34], [35] to improve flexibility, interoperability, and maintainability of the proposed method. Therefore our approach is complementary to recent standards, such as [15], and proposes a new DT-based approach for the cyber-attack detection problem, without loss of generality in the context of the existing works*

Proposed DT-based methobology

- 1) Physical Process: We assume that the physical process (referred to as process for the rest of the paper) is a manufacturing process that has sensors in place to collect in- and ex-situ data and the measurements are available to the DTs in the framework for data analysis purposes in run-time as well as in the form of historical data through a database. A discretetime representation of the process is then given in a general form as

$$\boldsymbol{x}(t+1) = f(\boldsymbol{x}(t), \boldsymbol{u}(t), \boldsymbol{w}(t)) \tag{1a}$$

$$\boldsymbol{y}(t) = g(\boldsymbol{x}(t), \boldsymbol{v}(t)), \tag{1b}$$

Proposed DT-based methobology

- An SME monitors the process through the DTs in the framework as illustrated in Fig. 2, and implements reconfigurations or changes to the process through a decision-maker.
- We assume that a decision-maker exists in the framework without loss of generality. The decision-maker provides setpoint references r(t) for the process to track (in the sense that $||r(t) - y(t)||$ is as small as possible in a suitable norm).
- While we assume that the process has the form in (1) for our further discussions and developments, systems of various forms and dynamics can be considered here (e.g., discrete event systems). Additionally, the process itself can be modeled as a separate DT to perform simulation-based analysis on the process.

Proposed DT-based methobology

- 2) Controller DT: The Controller DT houses the run-time controller with the control logic, as well as observers, process models, and simulation tools. The Controller DT employs various control methods and logic (e.g., feedback, feedforward, rule-based, hybrid, etc.) to regulate the process measurements y(t) toward the reference setpoints r(t) provided by the decision-maker.
- To perform state-based control, the Controller DT may incorporate various types of filters and estimators to estimate the current and future states of the process by using the measurements and information such as historical data, or model adaptation information provided by other DTs in the framework (e.g., models of the noises v(t) and w(t)).
- Control inputs $u(t) \in U$ are implemented on the process, where U denotes an input constraint set. In practical implementations, there may be additional safety control loops that bypass the control input implementation (e.g., emergency stop switch for a robotic manipulator).

Proposed DT-based methobology

- 3) Feature DT: The Feature DT provides uniform data streams to the DTs in the framework to improve the interoperability of the framework.
- Existing run-time anomaly detectionmethods often rely on residual analysis to provide thresholdbased decisions [7], [32].
- We assume that an SME defines the desired residual signals with specific features, and implements them as part of the Feature DT so that the residual information is shared with other DTs for further data analysis. Another important task of the Feature DT is to evaluate key process indicators (KPIs) for the process.
- Various types of KPIs include health indicators, performance indicators, and efficiency indicators [34], [36]. Similarly, the Feature DT may be tasked to pre-process or partition large scale or high sampling-rate measurement data for another DT that performs statistical learning on the measurement data.

Proposed DT-based methobology

- 4) FD/AD DT: The FD/AD DT performs fault and anomaly detection on run-time data streams. Preliminary detection capabilities are included in most CPS for reliable run-time performance.
- Such detection mechanisms are considered as part of the FD/AD DT here. The FD/AD DT is usually built to perform threshold-based limit-checking on the physical process.
- The FD/AD DT may include safety monitoring and performance monitoring systems to detect anomalies and faults.

Proposed DT-based methobology

- 5) The Cybersecurity DT: The Cybersecurity DT provides predictions about attacks on the system in the context of anomalies and transient response of the controlled process. We assume that the Cybersecurity DT is designed by an SME knowledgeable on the cybersecurity of the process and we focus on attacks with output measurable effects as stated earlier.
- In the absence of such prior knowledge, historical data may be used to understand the normal system behavior initially. In this context, abnormalities can be recorded during operation and labeled as normal, anomalous, or attack data by an SME. If an SME or enough historical data is not available to initialize the framework, the proposed approaches may not be applicable. The Cybersecurity DT is a novel contribution of this work to distinguish cyber-attacks from expected anomalies for a controlled process, and we provide a detailed analysis of the Cybersecurity DT in later sections.
- 6) SME Operator: The operator monitors the outputs of the FD/AD DT and the Cybersecurity DT to further analyze if the physical process has an anomaly or is under a cyber-attack. For this purpose, the DTs report their prediction quality and the features found in the data so that a human SME may further investigate any abnormalities.

Proposed DT-based methobology

- 7) Decision Maker: The role of the decision-maker is to provide an interface between the SME and the plant floor. Many CPMS in practice utilize a supervisory control and data acquisition (SCADA) layer as a decision-maker.
- The decision-maker may have a supervisory role where it takes actions on the plant floor by making autonomous decisions. If the decision-maker is purely advisory, the SME may implement actions and prescribe references directly to the controlled plant, bypassing the decision-maker.
- In our context, the decision-maker provides details and updates on the reference signal r(t) for the process.

The Cybersecurity DT

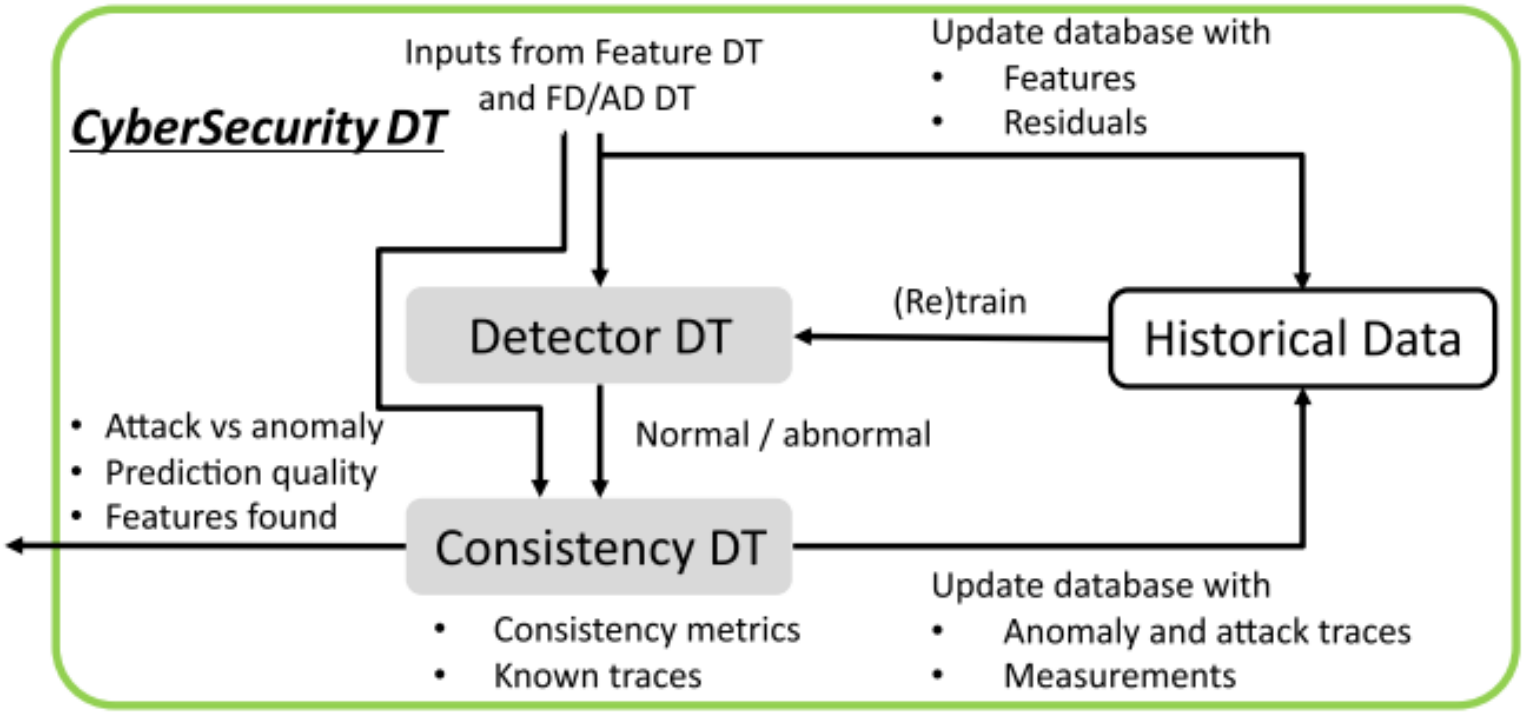

Fig. 3. The architecture of the Cybersecurity DT. The Detector DT and the Consistency DT are used for detecting abnormalities and attacks on the physical process. The historical data is stored in a database for model training as well as knowledge storage and SME data mining of the types of expected anomalies, attacks, etc.

The Cybersecurity DT

A. Cybersecurity DT Architecture

- We propose a Cybersecurity DT that utilizes two DTs to perform abnormality detection and attack detection, so that it can predict the presence of a cyberattack on the physical process. The Cybersecurity DT also has a database that includes historical data that is used for model training, data mining, and data analysis. Note that while the Cybersecurity DT uses run-time data to predict attacks, the DT may or may not run synchronously with the physical twin.
- We assume that the data streams within the framework are time-stamped such that asynchronous DT predictions that indicate predicted time-instance of an attack onset are possible. The actual time frame of the DT versus the physical process for a practical implementation depends on the application domain of the process.

The Cybersecurity DT

- 1) Detector DT: Noting that our goal is to detect attacks that have output measurable effects, we first need to identify if a measurement is abnormal. The Detector DT is tasked with performing abnormality detection on the process data by leveraging the anomaly prediction from the FD/AD DT.
- A key problem with anomaly and attack detection is the scarcity of abnormal process data versus the abundance of normal process data, leading to an unbalanced data set. For this purpose, machine learning models such as one-class discriminators [38], [40], [41], [42] and auto-encoders [43], [44], [45] are often utilized in the literature for abnormality detection to represent the normal data sufficiently well in a projected space such that abnormal data can be detected efficiently (see Section IV-B1).
- Data-driven models may be utilized in the Detector DT and with the assumption on the availability of sufficient normal process data to train data-driven models. Additionally, the FD/AD DT predictions may be utilized to improve the abnormality detection in the Detector DT.

The Cybersecurity DT

- 2) Consistency DT: If a measurement is labeled as abnormal by the Detector DT, or alternatively if an anomaly is detected by the AD-DT, further analysis is performed by the Consistency DT.
- To understand if an abnormal measurement is due to an attack or an anomaly, we utilize the notion of consistency metrics on the process data. A consistency metric for the physical process (1) characterizes the expected behavior of the system during expected anomalies, e.g., how a measurement signal changes due to expected mechanical wear.

The Cybersecurity DT

- 3) Historical Data: The historical database stores process data, the expected anomaly features, and data traces, as well as historical outputs from the Feature DT and the FD/AD DT. The database is updated with the outputs of the DTs for further analysis. Additionally, the SME updates the labels of the historical data to account for new expected anomalies encountered on the process. This procedure may be initialized with sufficient historical data to build consistency metrics. The SME can monitor abnormal data identified by the DTs to build a library of expected anomalies and better consistency metrics over time.

B. Proposed Illustrative Methods for Attack Detection

- We present the theoretical background for the proposed illustrative detection methods used in the Detector DT and the Consistency DT for abnormality detection and consistency metrics based attack detection, respectively, in this section.

The Cybersecurity DT

- 1) Abnormality Detection: To implement abnormality detection, the Detector DT is trained on the historical process data $D = \{ y(t), x(t), u(t), \eta(t) \mid t = t0, t0 + 1, \ldots, t0 + nw\}$, where $\eta(t) \in \{0, 1\}$ is a label for abnormality of a data point, to recognize features of normal process data.
- Note that in practice for an unbalanced dataset, we utilize only a single class label and define everything else as abnormal (i.e., η becomes trivial as all data in D corresponds to a single class).
- We denote the normal data boundaries trained by the Detector DT using the data D as $B(D) \subset F$, where F is a possibly nonlinear feature space where the Detector DT operates.

The Cybersecurity DT

- The Detector DT utilizes its trained model B(D) to monitor run-time data provided by the Feature DT and FD/AD DT and detect if current measurements of the physical process are normal, i.e., if $\psi(y(t')) \in B(D)$, where $\psi : Y \rightarrow F$ is a map from the measurement space Y to the feature space F of the Detector DT. Based on this analysis the Detector DT outputs its prediction as a label $\eta(\hat{} t') \in \{0, 1\}$ of normal versus abnormal.
- Additionally, probabilistic predictions and prediction quality measures may be provided. To this end, statistical learning methods such as the ones provided in [26] may be utilized. Alternatively, physics-based methods [28] may be utilized to detect abnormalities.
- The training for the Detector DT utilizes historical process data of the steady-state operation at a predefined setpoint reference, e.g., $r(t) = \bar{r}$, to train B(D). However, the setpoint of the process may be altered either by a decision-maker or by a closed-loop controller on the physical process. The setpoint changes result in transient dynamic behavior on the system (1),which may cause false positives by the Detector DT.

The Cybersecurity DT

- To mitigate false positives of the Detector DT during transients, we utilize the solution map of the process (1), $\phi : X \times U_\infty \times Z_+ \rightarrow X$, where X is the state space of (1) and U_∞ is the space of sequential control inputs on (1). Given an initial state x(t0) and a control sequence $u \in U_\infty$ over a time interval including the interval [t0, tc], we have

where x(tc) is the state at time tc (i.e., the current state).

- Our motivation for the proposed abnormality detection method is to utilize the trained data boundaries B(D) during transient response. Roughly speaking, as B(D) is trained for the process at a given setpoint, we define a projection using ϕ to estimate state of the process at a previous setpoint given the transient observations (i.e., as the process moves away from the said setpoint) and the control inputs. If the process is normal, (i.e., no attacks or anomalies), the projected state should be within B(D).

The Cybersecurity DT

- Remark 4: Forward projections of the set B(D) for the transient control inputs can also be used for abnormality detection. However defining such projections may in general be computationally expensive as B(D) may be control and state dependent, and new computations are needed at each control step. Therefore, we focus on the proposed projection type method for abnormality detection in this work.
- Formally, the goal of the Detector DT during transients is to estimate the initial state $x^{-}(t0)$ of the process at time t0 based on the observed sequence of states and control input u until the current time tc. Let us denote the model of the state progression as

$$\Phi(\boldsymbol{x}(t_0)) = \begin{bmatrix} \phi(\boldsymbol{x}(t_0), \boldsymbol{u}; t_0) \\ \phi(\boldsymbol{x}(t_0), \boldsymbol{u}; t_0 + 1) \\ \vdots \\ \phi(\boldsymbol{x}(t_0), \boldsymbol{u}; t_c) \end{bmatrix}. \qquad (3)$$

The Cybersecurity DT

- Additionally, let x denote the sequence of estimated states of the process between the times [t0, tc]. Then, the Detector DT solves the following minimization to estimate the initial state $\bar{x}(t0)$ by using the control input u and the state sequence x.

$$\bar{\boldsymbol{x}}(t_0) = \underset{z}{\operatorname{argmin}}\{||\Phi(z) - \boldsymbol{x}||\}, \qquad (4)$$

- where z is an intermediate variable for the notation. For a normal process (i.e., process outputs with $\psi(y(t')) \in B(D)$), the solution of (4) is close (in the normed distance sense) to the actual initial state x(t0).
- . Therefore, the Detector DT evaluates the abnormality of the projected state $\bar{x}(t0)$ to evaluate the label $\eta(\hat{} tc)$ for the current state x(tc). Namely, if $\bar{x}(t0) \in B(D)$, then the current state x(tc) is predicted as normal by the Detector DT.

The Cybersecurity DT

- We omit a detailed background on STL and refer interested readers to [47]. An STL formula π is formed by the following syntax:

$$\pi \triangleq \quad \top \quad | \quad p \quad | \quad \neg\pi \quad | \quad \pi_i \wedge \pi_j \quad | \quad \pi_i \, \mathcal{U}_{[a,b]} \pi_j \quad (5)$$

- where, ⊤ is logical true, p is a predicate, ¬π is the logical negation of the proposition π, πi ∧πj is the logical conjunction of two propositions, and πi U[a,b]πj is the until operator defined as the proposition πi being true at least until the proposition πj is true in the time interval [t +a, t +b], where t is the current time.

- A signal s(t) at time t is satisfied by a predicate p if ℓ(s(t)) > 0 for some function ℓ (i.e., s(t) |H p ⇐⇒ ℓ(s(t)) > 0). Here the operator |H is used to indicate that the condition on the left side satisfies the condition on the right side. Additionally, ⊥ = ¬⊤ is the logical false, the eventually operator is ◊[a,b]π ≜ ⊤ U[a,b]π, and the always operator is □[a,b]π ≜ ¬(◊[a,b]¬π).