Anomaly detection using invariant rules in Industrial Control Systems

Contents lists available at ScienceDirect

Control Engineering Practice

journal homepage: www.elsevier.com/locate/conengprac

Anomaly detection using invariant rules in Industrial Control Systems

Qilin Zhu [a,b], Yulong Ding [a,b], Jie Jiang [c], Shuang-Hua Yang [a,d,*]

[a] Shenzhen Key Laboratory of Safety and Security for Next Generation of Industrial Internet, Southern University of Science and Technology, Shenzhen, 518055, China
[b] Department of Computer Science and Engineering, Southern University of Science and Technology, Shenzhen, 518055, China
[c] College of Artificial Intelligence, University of Petroleum (Beijing), Beijing, 102249, China
[d] Department of Computer Science, University of Reading, Reading, RG6 6UR, UK

ARTICLE INFO

Keywords:
Industrial Control System
Anomaly detection
Invariant rule
Association rule mining

ABSTRACT

Industrial Control Systems (ICS) are intelligent control systems that integrate computing, physical processes, and communication to manage critical infrastructures such as power grids, oil and gas processing facilities, and water treatment plants. In recent years, ICS have been increasingly targeted by malicious attacks, causing severe consequences. Anomaly detection systems utilized in ICS are crucial in safeguarding ICS from potential threats by sending out an alert upon detecting any network attacks. However, existing methods for ICS anomaly detection often suffer from limitations. Supervised machine learning methods encounter the issue of imbalanced positive and negative samples, while residual-based anomaly detection methods face challenges in detecting stealthy attacks. This paper presents an unsupervised anomaly detection method for ICS using association rule mining techniques. Utilizing the proposed variation-driven predicate generation strategy, the method incorporates temporal features of sensor readings into the generated predicates, achieving the mining of invariant rules that take into account the temporal dependencies among physical variables. This approach allows for a more comprehensive exploration of the invariant patterns maintained in the dynamic processes of systems. Through experiments conducted on two public datasets, the method demonstrates high detection efficiency, meeting the real-time demands of online detection. Experimental results showcase its notable efficacy in anomaly detection, with a substantial enhancement in the recall rate. Furthermore, the method's capability to promptly issue warnings enables it to detect multiple attacks with low latency.

1. Introduction

ICS are integrated systems that comprise multiple automation control components and process control components designed to collect, process, and analyze real-time data to ensure the automatic operation of various industrial infrastructures. ICS typically consist of distributed components that interact with physical processes through sensors and actuators. These components are interconnected via the network, which renders the system vulnerable to cyberattacks. Any attack on ICS could severely damage critical infrastructures and have far-reaching negative impacts. In recent years, frequent ICS attack incidents have posed significant threats to the economic development and public safety. In 2009, the Stuxnet attack on Iran's nuclear facilities caused widespread concern (Falliere, Murchu, & Chien, 2011); the 2015 attack on Ukraine's power grid resulted in massive blackouts (Case, 2016); and the 2018 virus attack on TSMC led to the shutdown of crucial production facilities across multiple important bases (Kumar, 2018).

Anomaly detection systems monitor activities by analyzing data logs or network traffic to identify potential or ongoing cyberattacks. However, the conventional anomaly detection methods that have been extensively studied and applied in traditional Information Technology (IT) systems may not be entirely applicable to ICS. ICS security requires a holistic approach that considers both information systems and physical processes. Furthermore, the uninterrupted nature of ICS operations is crucial, making software patches and frequent updates unsuitable for ICS. Therefore, ICS security solutions should incorporate self-awareness, adaptive decision-making, and real-time response capabilities. Anomaly detection holds significant importance in ensuring the security of ICS (Wang, Zhou, Chen, & Wang, 2020). The effectiveness of anomaly detection algorithms is directly correlated with the security and efficiency of ICS operations (Zhou et al., 2015).

Anomaly detection based on physical processes is an effective defense mechanism against attacks on ICS, as such attacks often cause changes to the physical states of the system (Zheng, Julien, Kim, & Khurshid, 2015). Available anomaly detection methods based on physical processes mostly rely on traditional machine learning techniques or

[*] Corresponding author at: Department of Computer Science, University of Reading, Reading, RG6 6UR, UK.
E-mail address: shuang-hua.yang@reading.ac.uk (S.-H. Yang).

https://doi.org/10.1016/j.conengprac.2024.106164
Received 16 May 2024; Received in revised form 28 October 2024; Accepted 4 November 2024
Available online 19 November 2024

deep neural networks. Supervised methods necessitate sufficient normal and abnormal data to train a classifier. However, anomalous data is scarce in practice, and the positive and negative samples exhibit a considerable imbalance (Yuan, Yu, Yang, Duan, & Li, 2023). Many unsupervised methods (Chen et al., 2021; Li et al., 2019; Zeng et al., 2023) rely on residuals to detect anomalies by comparing the current state of the system with a predicted normal range or reconstructed value (Tian & Zhao, 2023). However, these methods are limited by unclear control boundaries and may fail to detect indirect and stealthy attacks (Feng, Li, Zhu, & Chana, 2017). Since ICS physical process data typically consist of continuous and categorical variables, some methods propose mixed modeling methods for concurrent analysis of continuous and categorical data (Chen, Zhao, & Ding, 2023; Eiras-Franco, Martinez-Rego, Guijarro-Berdinas, Alonso-Betanzos, & Bahamonde, 2019; Wang, Sheng, Zhou, & Chen, 2022; Wang, Zhou, & Chen, 2023; Wang et al., 2020). Since these methods rely on probabilistic modeling, they often require assumptions and priors, making the design of effective anomaly detection methods challenging.

ICS exhibit inherent deterministic characteristics, characterized by a relatively fixed network topology, a small and static set of application functions, and regular and predictable communication patterns (Zhou et al., 2015). As a result, some methods rely on invariant rules to collect information about the relationships among sensors and actuators in the system under normal operating conditions (Feng, Palleti, Mathur, & Chana, 2019). In the event of an attack, the physical processes of the system may be affected, causing associated sensors and actuators to exhibit structural changes in their normal behavior. These changes violate the invariant rules that the system maintains, making it possible to detect attacks by identifying such structural changes. Traditionally, these invariant rules were manually derived by system experts using their prior knowledge, which was a time-consuming, costly, error-prone, and non-portable process. In the literature, researchers have tried to utilize data-driven techniques for mining invariant rules, making the widespread application of such methods possible (Feng et al., 2019; Momtazpour, Zhang, Rahman, Sharma, & Ramakrishnan, 2015).

This paper proposes an anomaly detection method based on invariant rules generated from the physical process data of ICS. The method captures the normal behaviors of the system by mining the invariant rules that must always be fulfilled during its regular operations and flags non-compliant data as anomalous. By focusing on situations where the trend of sensor reading segments changes, the method incorporates temporal features of sensor readings into the generated predicates. The mined invariant rules capture the temporal dependencies among system physical variables, enabling a more comprehensive exploration of the typical normal patterns maintained in the dynamic processes of ICS.

Specifically, our method employs a data-driven association rule mining algorithm (Agrawal, Imieliński, & Swami, 1993) to generate invariant rules automatically. During the predicate generation phase, the sensor readings collected at discrete time steps are divided into smaller segments. These segments are then analyzed to generate meaningful predicates. The predicate for each data segment is expressed as a collection of data attributes from the previous and current segments, where each attribute is defined as the slope trend or mean level of each segment and highlights the changing processes as sensor reading evolves.

To evaluate our method, we conduct experiments on the publicly available Secure Water Treatment (SWaT) (Goh, Adepu, Junejo, & Mathur, 2017) dataset and the Water Distribution System (WADI) (Ahmed, Palleti, & Mathur, 2017) dataset. The experimental results demonstrate that our method achieves high detection efficiency and meets the real-time requirements of online detection. It can detect a wide range of attacks with low latency in attack identification. Compared to the existing anomaly detection method (Feng et al., 2019) based on invariant rules in ICS, our method improves the detection performance and effectively enhances the detection recall rate. The contributions of our method can be summarized as follows:

- It introduces an innovative variation-driven predicate generation strategy that utilizes (a, b) tuples to connect different changing trends between segments, extending the predicate generation perspective from current readings to historical processes.
- It focuses on trend changes in sensor reading segments and incorporates temporal features into predicate generation, enhancing the detection of anomalies by considering temporal dependencies among physical variables.
- It can detect stealthy attacks (attacks that make slight modifications to the sensor readings at each time step) since accumulated sensor deviation has a tendency to violate certain invariant rules at specific time points.
- It generates invariant rules that reflect certain underlying mechanisms of the system. This offers insights into the relationships among critical factors of ICS and possesses interpretability.
- It is unsupervised and utilizes data-driven techniques to mine meaningful invariant rules without relying on system models, providing high portability and applicability to similar IoT scenarios.

The remainder of this paper is structured as follows. Section 2 presents related works on ICS anomaly detection. In Section 3, the background and notations are provided. Section 4 presents the proposed anomaly detection method in this paper. In Section 5, the experimental settings and results are presented. Section 6 analyzes and discusses the experimental results. Finally, Section 7 presents the conclusions of this study.

2. Related works

Much research has utilized the physical process information of the system to develop anomaly detection methods, as cyberattacks on ICS often alter system physical states (Zheng et al., 2015).

Methods that utilize system physical process information for anomaly detection typically include residual discrimination methods and invariant rule-based methods. Aoudi, Iturbe, and Almgren (2018) and Maurya, Agarwal, Kumar, and Shukla (2022) propose residual discrimination anomaly detection methods that project measured raw physical process data onto a low-dimensional signal subspace using Singular Spectrum Analysis (SSA). They then create spherical (Aoudi et al., 2018) or ellipsoidal (Maurya et al., 2022) decision boundaries to partition the normal mode region. Using SSA for denoising enables the methods to detect subtle structural changes hidden within the noise range of the physical process variables. Neural network-based residual discrimination methods typically consider time-series features to detect anomalies. Let $r(t)$ denote the difference between the predicted or reconstructed values and the actual values of sensor readings at time t, and let J_{th} represent the constant threshold for anomaly detection. Assuming a false data injection (FDI) attack (injecting attack data into the original reading/signal) is initiated at time t_0. The detection of an attack is determined using the evaluation function $J(t) = \|r(t)\|_{\text{RMS}}$, where $\|r(t)\|_{\text{RMS}}$ represents the Root Mean Square (RMS) of the variable $r(t)$. An attack is confirmed if, at a certain time $t_d > t_0$, the evaluation function $J(t)$ surpasses the predefined threshold J_{th}, denoted as:

$$J\left(t_d\right) > J_{\text{th}}. \tag{1}$$

Chen et al. (2021), Li et al. (2019) and Zeng et al. (2023) consider sensor readings as multivariate time series and use adversarial learning for high-precision prediction of the target value, fully utilizing the temporal correlations between physical process variables, achieving outstanding anomaly detection performance. However, residual determination methods are almost incapable of detecting stealthy attacks, where attackers can hide their operations within the range of noise, eventually leading to system control failure by accumulating injected tiny biases that induce cascading effects (Dán & Sandberg, 2010; Feng et al., 2017; Liu, Ning, & Reiter, 2011).

An attack is stealthy when it can inject false data without being detected with respect to Eq. (1). More precisely, let r_n denote the sensor reading residual in normal conditions, and r_a represent the deviation of r due to an attack, *i.e.*, $r_a = r - r_n$. An FDI attack is stealthy for the anomaly detector if both the following conditions are satisfied (Zhang, Keliris, Parisini, & Polycarpou, 2021):

$$\begin{aligned} &\left\| r_a(t) \right\| \to 0 \text{ as } t \to +\infty, \\ &\| r(t) \|_{\text{RMS}} \le J_{\text{th}} - \delta \text{ for } t \ge t_0, \end{aligned} \tag{2}$$

where $\delta > 0$ is a predefined scalar such that $J_{\text{th}} - \delta > 0$. Consequently, a stealthy attack is considered undetectable for this residual-based anomaly detector, wherein the detection probability equals the probability of false alarms rate (Ghaderi, Gheitasi, & Lucia, 2020).

Mixed modeling methods make full use of the complementary information between continuous and categorical variables to model the data distribution that should be satisfied during normal system operation. Chen et al. (2023) proposes a mixed anomaly detection model that can handle both non-Gaussian and non-Bernoulli variables simultaneously. They introduce a finite mixed model to describe data for each class. Each component includes a multivariate Gaussian distribution and several categorical distributions, relying on the assumption of conditional independence to simplify parameter estimation. To estimate the parameters, they employ Variational Inference (VI) to determine the appropriate number of components for each class automatically. Eiras-Franco et al. (2019) presents an anomaly detection model that can handle large-scale and high-dimensional data. They split the mixed probability distribution into two components: the continuous variables' marginal density and the categorical variables' conditional probability based on the feature vector's continuous portion. However, the accuracy of anomaly detection with such methods heavily depends on robust and accurate probabilistic models, and even in simple industrial control processes, it is not easy to determine the model distribution.

The invariant rule-based methods aim to discover the unchanging relationships between physical process variables that must be maintained during normal operations. Traditionally, these invariant rules are often manually derived by experts who have a deep understanding of the system.

Adepu and Mathur (2016) utilizes the Process and Instrumentation Diagram (P&ID), which describes various equipment and control devices, to deduce plant design (for instance, when analyzing the SWaT testbed described in Section 5.1, factors such as valve opening or closing times, flow rates along pipelines, and chemical dosing rates can be considered), thereby uncovering system invariants. Yoong, Palleti, Maiti, Silva, and Poskitt (2021) generates invariant rules necessary for the normal operation of the system by iteratively decomposing the design logic of the ICS based on the principles of axiomatic design. Mishra, Palleti, and Mathur (2019) proposes a framework that allows for modeling the entire critical infrastructures and the interconnections within or among them. It leverages system architecture knowledge to model each subsystem individually, thereby uncovering the invariant relationships among these systems. Mehmood, Baig, and Syed (2024) uses association rule mining techniques to generate attack rules and invariant rules that must be maintained during the system's normal operation, dividing sensor readings into two intervals: "high" and "low".

However, such methods have a high transplantation cost and are difficult to mine a complete set of rules. In contrast, data-driven invariant rule mining methods, which do not rely on expert knowledge, have received extensive research attention. Das, Adepu, and Zhou (2020) propose a supervised classification method based on a partially defined Boolean function and Logical Analysis of Data to mine invariant rules from historical sensor readings. This method can detect abnormal behaviors in near real-time using laptop-class computing power. Compared to unsupervised methods, supervised methods have relatively poor performance detecting zero-day attacks and face challenges due to extremely imbalanced data caused by the lack of abnormal samples. Momtazpour et al. (2015) quantify the relationships between time series using the AutoRegressive models with eXogenous inputs (Ljung, 1998), and if any relationship does not change over time, it is considered invariant. The authors use latent variables obtained from factor analysis to supplement the indirect relationships among time series in physical process data and represent all direct or indirect relationships of variables as an invariant graph of the system. However, this method has a high computational complexity and is not suitable for real-time online detection in ICS.

When applying invariant rule-based methods for anomaly detection in ICS, we face several fundamental challenges: How can continuous real-valued data be incorporated into invariant rules, considering that industrial control processes often involve both continuous and categorical variables? How can we discretize continuous data to ensure the discrete representation effectively captures its characteristics? How can we use data-driven methods to discover meaningful invariant rules without relying on prior knowledge? These challenges point to a critical and effective solution strategy: discretize continuous data into a finite set of representative predicates and use association rule mining techniques (Agrawal et al., 1993) to automatically discover the invariant relationships among system physical process variables. Such methods have the advantages of being data-driven, unsupervised, and real-time responsive, and they generally involve three main steps: predicate generation based on physical process data, closed frequent itemset mining, and invariant rule generation.

Predicate generation is the crucial and challenging step. It defines a set of mapping relationships that primarily map continuous sensor readings into finite and discrete descriptions that can reflect the states of the sensors. For example, a common approach is to divide the range of sensor readings into multiple intervals and use the interval index to which the reading belongs as the predicate for that reading. However, since predicate generation involves discretizing continuous variables, using simple interval partitioning to generate predicates could result in significant information loss (*e.g.*, we might completely fail to distinguish between overlapping intervals of two different distributions). Nonetheless, some level of discretization helps to mitigate the impact of noise and sensitivity to changes in the data distribution, thereby assisting in identifying the main dependencies among system variables. Therefore, it is necessary to design a predicate generation strategy that effectively captures the primary characteristics of the sensors. Following predicate generation, the next steps are closed frequent itemset mining and invariant rule generation, which can be achieved using mature association rule mining techniques to generate if-$then$ statements composed of predicates.

Feng et al. (2019) propose two predicate generation strategies consistent with ICS's control dynamics: the distribution-driven strategy and the event-driven strategy. This paper refers to their method as DDEA (Distribution-Driven and Event-Driven). The distribution-driven strategy exploits the influence of the ICS's control states on the updates of sensor readings. For any given sensor, this strategy classifies the updates in sensor readings at different time steps into multiple Gaussian distributions, where each distribution corresponds to a predicate.

The event-driven strategy leverages the observation that sensor readings reaching critical values generally prompt changes in actuator states. Therefore, this strategy aims to identify the critical values that sensor readings must satisfy when each actuator state alters. The strategy then classifies the readings of each sensor at any other time steps into two discrete predicates: "not yet reached the critical values" or "already exceeded the critical values".

Our proposed method is an invariant rule-based anomaly detection method that utilizes association rule mining techniques. The main difference from DDEA lies in predicate generation. While DDEA only considers the impact of the current control state on the system, it lacks perception of historical processes and thus cannot capture the temporal correlations of sensor readings. Consequently, it cannot adequately consider the most crucial temporal features of time-series data. Additionally, in industrial control processes, noises inevitably affect

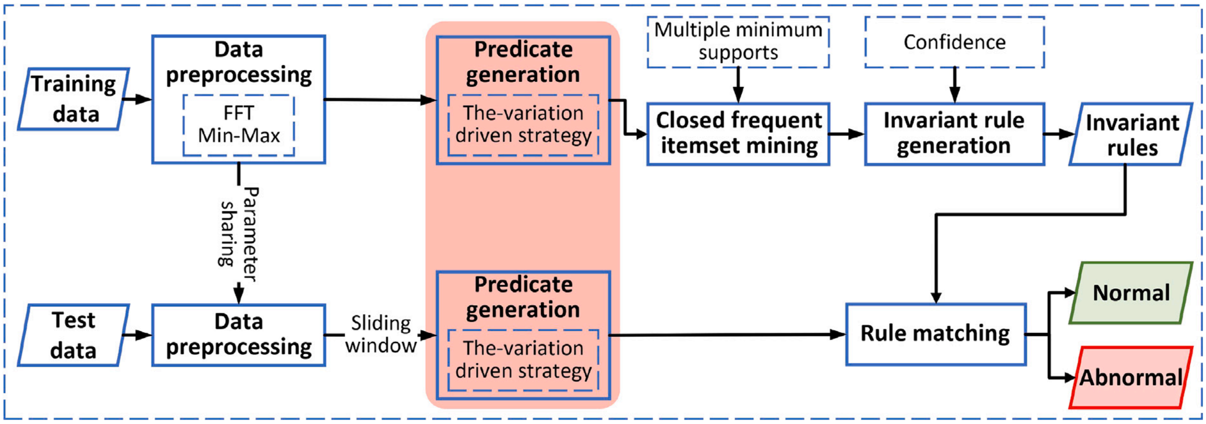

Fig. 1. Proposed framework for anomaly detection in ICS.

sensor readings, leading to tiny local fluctuations. Thus, considering only the current moment makes it prone to misjudging the overall changing trend of readings.

Existing invariant rule-based anomaly detection methods generally emphasize predicate generation based on the current moment for continuous real-valued data, overlooking the critical temporal features of sensor readings. Many methods directly divide the range of sensor readings into several non-overlapping intervals, with the predicate at each moment determined by the interval the reading falls into. For example, they generate predicates such as "in the low range" or "in the high range" (Maiti, Yoong, Palleti, Silva, & Poskitt, 2023; Mehmood et al., 2024; Mishra et al., 2019).

Due to the stable overall variation patterns of industrial processes and the general periodic characteristics of the dynamic processes, it is essential to fully utilize the temporal correlations. To this end, we use the variation-driven strategy for predicate generation. Sensor readings within a sliding time window are divided into segments, and the temporal features are incorporated into the generated predicates based on the changes in the slope trend or mean level of each segment. This method allows for the discovery of invariant relationships among physical variables in ICS dynamic processes.

3. Background and notations

ICS acquire process information through sensors and control processes with actuators. For an ICS equipped with m sensors and n actuators, let $D\{1:T\} = \{\left\{S_1\{1:m\}, A_1\{1:n\}\right\}, \left\{S_2\{1:m\}, A_2\{1:n\}\right\}, \ldots, \left\{S_T\{1:m\}, A_T\{1:n\}\right\}\}$ be the time-series data log collected under T discrete time steps in the normal working state of the ICS, where $S_t\{1:m\}$ represents the values of m sensors at time step t, while $A_t\{1:n\}$ represents the states of n actuators.

Predicates represent discrete entities possessing statistical significance, serving as indicators of sensor values and actuator states at each time step. The main work of the predicate generation step is to discretize and limit the continuous sensor readings into finite and well-defined predicates. As an illustration, let us consider a liquid temperature sensor denoted as k, monitoring temperature values within the range $[10, 60]$. Let k_{val} represent the temperature value measured by sensor k. In this context, predicates for sensor k could potentially be formulated as $\{10 \leqslant k_{val} < 30, 30 \leqslant k_{val} \leqslant 60\}$, among other possible formulations. Subsequently, each item $\left\{S_t\{1:m\}, A_t\{1:n\}\right\}$ in the time series data log $D\{1:T\}$ can be represented as a predicate set $P_t = \left\{p_1, p_2, \ldots, p_h\right\}$, where each term p_i, $i \in (1, 2, \ldots, h)$, represents a predicate. For instance, consider a predicate set $P_1 = \{\text{P502} = 1, 10 \leqslant k_{val} < 30\}$, which conveys that at time t=1, actuator P502 is in state 1, and sensor k records temperature value within the interval $[10, 30)$.

The predicate set P_t that represents the system's states at any given time t is referred to as a transaction, and the transactions collected under T times are called a transaction set D_{trans}. Association rule mining aims to identify rules of the form $A \longrightarrow B$ in the transaction set generated by $D\{1:T\}$, where A and B are two disjoint sets of predicates. These rules indicate that whenever a transaction contains the predicate set A, it should also contain the set B. Such rules describe the invariant relationships that must be maintained among different physical variables of ICS. To illustrate, consider the following invariant rule: {P502=1, P601=1} $\longrightarrow$ {P102=1, P401=1, P403=1} which implicates that if the pump actuators P502 and P601 are in state 1 at any given time t, then the actuators P102, P401 and P403 must also be in state 1 at the same time t. Association rule mining involves closed frequent itemset mining and invariant rule generation. In this context, we solely present the definition of closed frequent itemsets and provide a concise overview of the approach employed for generating invariant rules based on these closed frequent itemsets. More details will be elaborated in the next section.

An itemset is a set of items. In ICS, an item is typically interpreted as a predicate, and an itemset can be viewed as a predicate set. The occurrence frequency of an itemset is equal to the frequency of all transactions that contain the itemset, also referred to as the support. Itemset Z is considered a frequent itemset if its support is greater than or equal to a pre-defined minimum support threshold. A closed itemset is defined as an itemset Z for which the support of any of its direct supersets (if itemset Z_1 is a direct superset of itemset Z_2, then Z_2 is a proper subset of Z_1, denoted as $Z_2 \subsetneq Z_1$) is strictly less than the support of itemset Z itself. A closed frequent itemset Z' is a frequent itemset that is also closed.

The generation of invariant rules is based on closed frequent itemsets. Each closed frequent itemset Z' is partitioned into two disjoint subsets, A and $Z' - A$. The generation of invariant rules aims to identify relationships $A \rightarrow B$ (where $B = Z' - A$) that are consistently maintained in the training set.

To detect a sample, we first generate the predicate set for this sample. This predicate set is then compared to the invariant rules generated from the training data $D\{1:T\}$. Any test sample with a predicate set that fails to conform to any invariant rule is considered anomalous. Regarding the invariant rule mentioned in the example above, if a given test sample satisfies the conditions of both actuators P502 and P601 being in state 1 while actuator P102 is in state 0 at the same time, then it is classified as anomalous.

4. Proposed framework

The proposed framework for anomaly detection in ICS is presented in Fig. 1. This framework includes preprocessing the training data, predicate generation, closed frequent itemset mining, and invariant rule generation. When processing the test data, the retained parameters from the training data preprocessing stage are used for preprocessing. Then, the predicate generation step is performed, and the generated predicate set is compared to the invariant rules generated from the training data. Data samples that conform to all the invariant rules are classified as normal, while those that fail to satisfy any invariant rule are identified as anomalous.

4.1. Data preprocessing

The data preprocessing module consists of two steps: selecting sensor readings using the Fast Fourier Transform (FFT) and normalizing the selected readings using Min-Max normalization.

The sensor readings collected at discrete and equidistant time steps constitute multiple time series. Discrete Fourier Transform (DFT) converts discrete signals from the time domain to the frequency domain, making it suitable for analyzing sensor readings. FFT is a fast algorithm that computes the DFT by reducing repeated calculations, making it practical for large datasets (Brigham, 1988; Cooley & Tukey, 1965).

For the time series collected by any sensor, after applying FFT, we can identify the three sine waves with the highest amplitudes that contribute the most to the sensor series and determine their corresponding periods. The amplitude of each component in the spectrogram illustrates its contribution to the time series. Thus, a larger amplitude of a component indicates that the corresponding sine wave period is a better representation of the overall period of the time series. If the minimum period among these three dominant periods exceeds a predefined threshold L, it indicates that the periodicity of the sensor time series is not significant. This could impede the mining of meaningful invariant rules. Therefore, this sensor time series is removed from further analysis.

After selecting the sensors to be used, the next step is normalizing the relevant sensor readings. We use Min-Max normalization to normalize the time series of each sensor. Min-Max normalization performs a linear transformation on the original data, scaling the data to the range of $[0, 1]$. The equation is presented below:

$$X_{\text{nom}} = \frac{X - X_{\min}}{X_{\max} - X_{\min}}, \tag{3}$$

where X is the original time series, $X_{\max}$ and $X_{\min}$ are the maximum and minimum values of X, respectively, and X_{nom} is the normalized series.

The time series of each sensor in the test set is normalized based on the maximum and minimum values observed in the corresponding sensor data from the training set. This offers a reliable way of capturing the overall deviation of the test set data from the training set data and facilitates online detection.

4.2. Predicate generation

In order to represent the discrete and limited states of an actuator, we can assign a predicate to each possible state. Assume that an actuator denoted by u has a set of possible states $\{v_1, v_2, \ldots, v_z\}$ as observed in the data log. Then, we can generate the following predicate set to represent the possible states: $\{u = v_1, u = v_2, \ldots, u = v_z\}$. However, for sensors, since the readings obtained are continuous variables that span an infinite range, it is necessary to partition the continuous range into discrete and finite predicates.

We propose the variation-driven strategy for generating sensor reading predicates, which considers the dynamic processes of the physical variables of the system and complements the temporal correlations among these variables, as compared to DDEA.

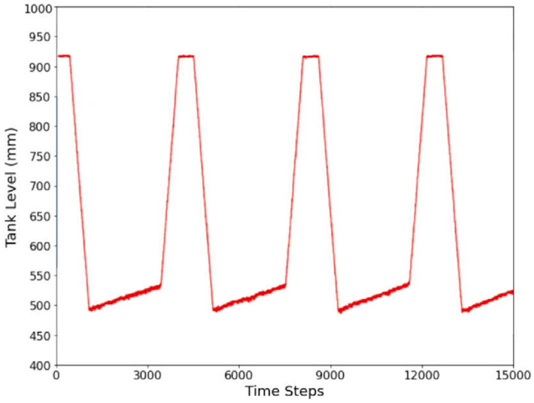

Fig. 2. Trend chart of sensor readings for sensor LIT101 under 15,000 time steps.

4.2.1. The variation-driven strategy

The variation-driven strategy focuses on scenarios where the trend of sensor reading segments changes, with a particular emphasis on the slope trend or mean level of the sensor reading before and after the change. The variation-driven strategy is based on the experience that the state of a system in the current moment is influenced by its previous state, which is consistent with the control dynamics of general ICS. The strategy aims to identify invariant rules that can uncover consistent patterns within the dynamic processes of the system when it is in the normal state.

The variation-driven strategy initially partitions each sensor time series into continuous segments that exhibit the same changing trend. We employ the Bottom-Up (BU) time series segmentation algorithm proposed by Keogh, Chu, Hart, and Pazzani (2001). This algorithm approximates the univariate time series X, composed of sensor readings collected at T successive time steps ($X = \{x_0, x_1, x_2, \ldots, x_{T-1}\}$, with each x_t representing the sensor reading at time step t), through utilizing multiple linear lines. The BU algorithm initially partitions time series X into segments composed of adjacent pairs of points, resulting in $T/2$ segments overall. Subsequently, neighboring segments are paired together and collectively treated as a unit. The Least Squares Method (LSM) is then employed to fit a straight line through all the data points within each pair of segments, and the fitting error is calculated. Afterward, the two segments exhibiting the minimum fitting error while satisfying an error threshold are merged into a new segment. This process is iteratively repeated until no adjacent segments can be further merged. The detailed algorithm is presented in Algorithm 1.

After segmentation, attributes are assigned to each segment based on its characteristics. Each segment is classified as either a flat or changing segment, depending on whether the slope exceeds a predefined threshold K. Flat segments are represented by their mean values, representing the overall level maintained by them. Changing segments are quantified by their slopes, reflecting the overall changing trend within each segment. To generate more representative and statistically significant predicates, we perform coarse-grained processing on both the mean value of each flat segment and the slope of each changing segment.

The coarse-grained process involves categorizing the mean values of the flat segments into several ranges, such as “low” ($\leqslant 0.4$), “medium” (> 0.4 and < 0.6), and “high” ($\geqslant 0.6$).

Algorithm 1 Bottom-Up Time Series Segmentation

```
Input: time series $X$, error threshold $max\_error$
Output: time series segments $Seg$
for $i = 0, 2, 4, \ldots$ do
    Initialize $Seg(i)$ as a segment consisting of adjacent points $\{x_i, x_{i+1}\}$ in $X$.
end for
for $i = 0, 1, 2, \ldots$ do
    Employ LSM to determine a linear fit containing all the data points of the segments $Seg(i)$ and $Seg(i+1)$.
        $x = \beta_0 + \beta_1 t$
        $\beta_0 = \frac{\sum_{v=i-n+1}^{i} t_v^2 \sum_{v=i-n+1}^{i} x_v - \sum_{v=i-n+1}^{i} t_v \sum_{v=i-n+1}^{i} t_v x_v}{n \sum_{v=i-n+1}^{i} t_v^2 - \left(\sum_{v=i-n+1}^{i} t_v\right)^2}$
        $\beta_1 = \frac{n \sum_{v=i-n+1}^{i} t_v x_v - \sum_{v=i-n+1}^{i} t_v \sum_{v=i-n+1}^{i} x_v}{n \sum_{v=i-n+1}^{i} t_v^2 - \left(\sum_{v=i-n+1}^{i} t_v\right)^2}$
    Assuming that these two segments contain $n$ data points in total, $t_i$ represents the time, and $x_i$ represents the sensor value at that time.
    Let $fitting\_cost(i)$ be the Sum of Squared Errors (SSE) between the observed values and the corresponding estimated values of the $n$ data points from the linear fit.
        $fitting\_cost(i) = \sum_{v=i-n+1}^{i} \left(x_v - \left(\beta_0 + \beta_1 t_v\right)\right)^2$
end for
while the minimum value of $fitting\_cost < max\_error$ do
    Identify the two segments referred to by the minimum value in $fitting\_cost$:
        $\mathrm{argmin_j}\ fitting\_cost(j)$
    Merge $Seg(j)$ and $Seg(j+1)$ to generate a new $Seg(j)$.
    Update the indices of all segments after $Seg(j+1)$.
    Recalculate $fitting\_cost(j-1)$ and $fitting\_cost(j)$.
    Update the indices of all $fitting\_cost$ after $fitting\_cost(j+1)$.
end while
```

Moreover, the slopes of the changing segments are clustered using Variational Bayesian Gaussian Mixture Models (VBGMM) to group similar slopes. VBGMM differs from Gaussian Mixture Models based on the Expectation Maximization algorithm, as it can automatically determine the optimal number of clusters. The attribute of each segment is then represented by the range of the mean value or the slope category.

The final stage is to generate predicates. For each segment, we define a predicate in the format of (a, b), where a represents the preceding attribute that differs from the attribute of the current segment, and b represents the current attribute. Attributes reflect the overall changing trend or overall numerical level of each segment, and the combination of a distinct attribute from the preceding segment, along with the current attribute, can depict the historical trend changes of the present data. Under normal operating conditions, ICS typically exhibit recurrent sensor reading patterns. Consequently, it is possible to effectively depict the consistent evolving patterns in normal conditions by using predicates expressed in the format of (a, b).

The application of the variation-driven strategy in generating predicates can be exemplified as follows:

Fig. 2 illustrates the sensor readings over a period of 15,000 discrete time steps collected from the liquid level sensor LIT101 in the SWaT dataset described in Section 5.1. The variation-driven strategy classifies the sensor readings depicted in Fig. 2 into four distinct changing trend categories: slow rising, sharp rising, flat, and sharp falling. Supposing that the mean level of the flat segment is "high", the resultant complete attributes generated contain "slow rising", "sharp rising", "high", "sharp falling". Consequently, the corresponding predicate set can be formulated as follows: {LIT101=("slow rising", "sharp rising"), LIT101=("sharp rising", "high"), LIT101=("high", "sharp falling"), LIT101=("sharp falling", "slow rising")}

4.2.2. Online predicate generation

The BU algorithm requires one segmentation operation for each sensor time series in the training set. However, for the test data, multiple segmentation operations are needed for every sensor time series X since the measurements arrive successively. To avoid excessive computational overhead, we employed sliding window for online predicate generation.

We use the sub-time series W_t as the input for the BU algorithm, derived by applying a sliding window approach. Specifically, W_t comprises a sequence of data points of length S that terminates with the current values x_t at time step t. Formally, W_t can be represented as $W_t = \left\{x_{t-S+1}, x_{t-S+2}, \ldots, x_t\right\}$. For example, if the window size S is 128 and the current time step is $t = 128$, then the sliding window will include the data points $\left\{x_1, x_2, \ldots, x_{128}\right\}$. Similarly, at time step $t = 129$, the sliding window will shift to include $\left\{x_2, x_3, \ldots, x_{129}\right\}$. Subsequently, each sub-time series W_t is fed into the BU algorithm for segmentation.

Algorithm 2 Online predicate generation method based on the variation-driven strategy

```
Input: sensor time series $X$, sliding window size $S$, flatness determination slope threshold $K$
Output: predicates of sensor readings at each time step
for $i = S-1, S, S+1, \ldots$ do
    Let the sliding window sub-time series $W_i$ take the continuous values $\left\{x_{i-S+1}, x_{i-S+2}, \ldots, x_i\right\}$ from $X$.
    Feed $W_i$ into Algorithm 1 for time series segmentation.
    Obtain the slope of the linear regression for the last segment after segmentation:
        $x = \beta_0 + \beta_1 t$
        $\beta_1 = \frac{n \sum_{v=i-n+1}^{i} t_v x_v - \sum_{v=i-n+1}^{i} t_v \sum_{v=i-n+1}^{i} x_v}{n \sum_{v=i-n+1}^{i} t_v^2 - \left(\sum_{v=i-n+1}^{i} t_v\right)^2}$
    Assuming that the last segment contains $n$ data points in total, $t_i$ represents the time, and $x_i$ represents the sensor reading at that time.
    if slope $\beta_1 <$ threshold $K$ then
        Determine the range of the mean value for this segment.
        the attribute of $x_i$ = the slope range ("low", "medium" or "high")
    else
        Apply the VBGMM to categorize the slope for this segment.
        the attribute of $x_i$ = the category index ($K_1$, $K_2$, $K_3$ or $K_4$) obtained from clustering
    end if
    if the attributes of $x_i$ and $x_{i-1}$ are different then
        the predicate of $x_i$ = (the attribute of $x_{i-1}$, the attribute of $x_i$)
    else
        the predicate of $x_i$ = the predicate of $x_{i-1}$
    end if
    Record the predicate of $x_i$ as the predicate of sensor reading at time step $i$.
end for
```

Once the linear regression slope of the corresponding segment containing measurements x_t is determined, the subsequent steps remain consistent with the previously described processes. Our online predicate generation method based on the variation-driven strategy is depicted in Algorithm 2.

4.3. Closed frequent itemset mining

As defined in Section 3, a closed frequent itemset is an itemset that is both *frequent* – having a support that meets or exceeds a specified minimum threshold – and *closed*, meaning that none of its supersets has the same support.

The mining of frequent itemsets relies on the support threshold. Our method utilizes multiple minimum supports to ensure the statistical significance of the generated invariant rules.

4.3.1. Multiple minimum supports

Support refers to the frequency with which an itemset appears in the transactions. For instance, if we consider a predicate set like {P502=1, P601=1}, the support is determined by how often these conditions are simultaneously satisfied.

Methods that rely on a uniform minimum support face the rare itemset problem. Specifically, setting a high threshold risks overlooking rare but valuable itemsets, whereas a low threshold may lead to generating a large number of frequent itemsets, increasing the rate of false positives in anomaly detection (Liu, Hsu, & Ma, 1999). To address this issue, our method uses multiple minimum supports as applied in Feng et al. (2019), where each itemset can satisfy a different support threshold depending on the items included.

The support function $\sigma(.)$ is formally defined by Eq. (4), which quantifies the frequency of occurrence of an itemset Z in a given transaction set. The numerator in the equation represents the number of transactions containing the itemset Z, while the denominator represents the total number of transactions in the transaction set.

$$\sigma(Z) = \frac{\left|P_t \in D_{trans}; Z \in P_t\right|}{\left|D_{trans}\right|}, \tag{4}$$

where P_t denotes the predicate set at time step t and is referred to as a transaction. The collection of these transactions over T time steps forms the transaction set D_{trans}.

Suppose Z represents the itemset $\left\{i_1, i_2, \dots, i_n\right\}$. Since the occurrence of multiple items in transactions is not more probable than the occurrence of any individual item, for example, the proportion of $P502 = 1$ (or $P601 = 1$) appearing in the transaction dataset is certainly higher than the proportion of both $P502 = 1$ and $P601 = 1$ appearing simultaneously in the same dataset. Eq. (5) is always satisfied.

$$\sigma(Z) \leq \min\left(\sigma\left(i_1\right), \sigma\left(i_2\right), \dots, \sigma\left(i_n\right)\right). \tag{5}$$

Therefore, the support threshold of an invariant rule should exceed the product of the upper support limit of the itemset Z that makes up that rule with a scaling factor γ, where $\gamma \in (0, 1)$. Additionally, a global threshold θ, $\theta \in (0, \gamma)$ is used to ensure that any discovered itemsets have a statistically significant minimum support, preventing the discovery of coincidental rules. It is crucial that θ is set to be lower than γ, otherwise, $\gamma \cdot \min\left(\sigma\left(i_1\right), \sigma\left(i_2\right), \dots, \sigma\left(i_n\right)\right)$ will fall below θ. This would lead to a uniform minimum support across all rules, which is undesirable. To this end, Eq. (6) provides the formula for multiple minimum supports. An itemset Z is considered frequent only if it meets the criteria defined in Eq. (6).

$$\sigma(Z) > \max\left(\gamma \min\left(\sigma\left(i_1\right), \sigma\left(i_2\right), \dots, \sigma\left(i_n\right)\right), \theta\right). \tag{6}$$

Eq. (6) suggests a method to determine the multiple minimum supports for frequent itemsets. It imposes stricter statistical requirements on more common itemsets while assigning uniform minimum support θ to rare itemsets to prevent a high false positive rate in anomaly detection.

Parameter Tuning. Selecting optimal values of the parameters γ and θ in multiple minimum supports in Eq. (6) is crucial. Lowering these thresholds can generate more invariant rules and increase the possibility of detecting anomalies. However, it can also compromise the statistical significance of the generated rules, resulting in more false positives. As determining the optimal values of γ and θ relies solely on the validation steps, selecting parameters must trade-off between the number of rules generated and the false positive rate on the validation set. We adopt a parameter tuning approach inspired by the method proposed in Feng et al. (2019). This approach involves splitting the normal data log into training and validation datasets and selecting various values for γ and θ. The selected parameter values are then used to generate invariant rules on the training dataset. Subsequently, these rules are evaluated on the validation dataset, and any anomalies detected are false positives.

Let $N(\gamma, \theta)$ denote the number of invariant rules generated under the set parameters, while $\text{Acc}(\gamma, \theta)$ denotes the detection accuracy of the invariant rules when evaluated on the validation dataset, $1-\text{Acc}(\gamma, \theta)$ reflects the false positive rate on the validation set, τ_e be the user-defined acceptable validation error threshold. In order to obtain a maximum number of meaningful invariant rules to increase the possibility of detecting anomalies, the ideal values of γ and θ are obtained using Eq. (7):

$$\begin{aligned} &(\gamma^*, \theta^*) = \arg\max_{\gamma,\theta} N(\gamma, \theta) \\ &\text{subject to } \ \text{Acc}(\gamma, \theta) \geq 1 - \tau_e. \end{aligned} \tag{7}$$

4.3.2. Closed frequent itemset mining algorithm

Our method utilizes the Conditional Frequent Pattern-growth (CFP-growth) algorithm (Hu & Chen, 2006) for mining frequent itemsets from the transaction set. Subsequently, we identify all closed itemsets from the discovered frequent itemsets, ultimately generating the complete closed frequent itemsets. CFP-growth is an algorithm for mining frequent itemsets with multiple minimum supports. It employs the Multiple Item Support Tree to store crucial information about frequent itemsets, requiring just a single scan of the transaction set to extract complete frequent itemsets. The CFP-growth algorithm primarily consists of four steps: FP-tree construction, frequent itemset mining, recursive calls, and merging frequent itemsets. Due to the complexity of the algorithm, we do not elaborate on it and instead encourage interested readers to refer to Hu and Chen (2006) for more details.

4.4. Invariant rule generation

Invariant rules within ICS refer to the unchanging relationships that must be maintained among physical variables during normal operations, including but not limited to pressure, water level, valve opening or closing. Therefore, these rules serve as a certain reflection of the system mechanism, providing insights into the relationships among critical factors. In the event of an attack, alterations in the physical states of ICS may occur, leading to changes in the values of sensors and the states of actuators. These changes can disrupt the dependencies among physical variables.

In association rule mining, confidence is another main variable besides support. Confidence is used to measure the trustworthiness of the rule and is defined as follows:

$$C(A \rightarrow B) = \frac{\sigma(A \cup B)}{\sigma(A)}, \tag{8}$$

where the function $\sigma(.)$ represents the itemset support as defined in Eq. (4).

The generation of invariant rules is based on closed frequent itemsets to ensure statistical significance and prevent the generation of redundant rules.

To initiate the mining process, each closed frequent itemset Z' is partitioned into two disjoint subsets, A and $Z' - A$. The confidence of the rule $A \rightarrow Z' - A$ is then calculated. If this confidence meets a predetermined threshold, the rule $A \rightarrow B$ $\left(\text{ where } B = Z' - A\right)$ is considered to be successfully mined. This procedure is repeated for all closed frequent itemsets derived from the transaction set, resulting in the identification of complete ICS invariant rules (Feng et al., 2019). The method for mining complete invariant rules within closed frequent itemsets is detailed in Algorithm 3.

In ICS, invariant rules generated from association rule mining on physical process data can be considered a manifestation of the mechanism model. These rules may reflect the fundamental workings of the system or other inherent mechanisms, possessing a certain interpretability. Therefore, setting the confidence threshold to 1 is necessary for the rules to effectively capture the invariant relationships within the system, thereby reflecting some inherent nature of ICS.

As a result, the rule $A \rightarrow B$ dictates that whenever predicate set A appears in a transaction, set B must also appear in the same transaction.

4.5. Online anomaly detection

To detect anomalies in test samples, we first generate a predicate set for each sample using the method outlined in Section 4.2.2. The generated predicate set for each test sample is then compared against the invariant rules derived from the training set $D\{1 : T\}$. An anomaly is identified when a test sample's predicate set satisfies the antecedent (the if part of the if-$then$ expression) of an invariant rule but fails to fulfill the consequent (the $then$ part), indicating a deviation from the expected behavior. A test sample is deemed normal only if it adheres to all the invariant rules.

Algorithm 3 Mining complete invariant rules within closed frequent itemsets

```
Input: closed frequent itemsets $CFI$, confidence threshold $min\_conf$
Output: invariant rules
for $Z'$ in $CFI$ do
    Get the itemset support $Supp0$ for the closed frequent itemset $Z'$.

    length $Len$ = the number of predicates in $Z'$
    for $i = 1, 2, \dots, 2^{Len-1} - 1$ do
        Initialize subsets A and B.
        for $j = 0, 1, \dots, Len - 1$ do
            if the rightmost bit of the binary representation of integer $i$ is 1 when right-shifted by $j$ positions then
                Append the $j$-th element of $Z'$ to subset $A$.
            else
                Append the $j$-th element of $Z'$ to subset $B$.
            end if
        end for
        Get the itemset support $SuppA$ for subset $A$.
        Get the itemset support $SuppB$ for subset $B$.
        if $Supp0/SuppA \geqslant min\_conf$ then
            Generate the invariant rule $A \rightarrow B$.
        end if
        if $Supp0/SuppB \geqslant min\_conf$ then
            Generate the invariant rule $B \rightarrow A$.
        end if
    end for
end for
```

5. Experiments

The experiments are conducted using Python 3.7.4 on Windows 10 with an Intel(R) Core(TM) i7-8750H CPU, 16 GB, and an NVIDIA GeForce GTX 1060, 6.0 GB.

5.1. Datasets

We conduct our experiments on the SWaT and WADI datasets, which are widely used for ICS anomaly detection and serve as publicly available benchmark datasets in this area.

SWaT Dataset. The SWaT (Goh et al., 2017) dataset is provided by the iTrust Laboratory at the Singapore University of Technology and Design. The dataset is based on an operational water treatment testbed, a scaled-down version of a modern large-scale plant in major cities. The water purification process in the SWaT testbed consists of six subprocesses. The first process is used for the supply and storage of raw water. The second process conducts a basic assessment of water quality. Unwanted substances are filtered through an ultrafiltration system in the third process. In the fourth process, a dechlorination step eliminates any remaining chlorine. The fifth process uses a reverse osmosis system to reduce inorganic impurities. Finally, the sixth process stores the purified water until it is ready for distribution. 36 attacks with varying purposes and durations were executed on the SWaT testbed, including post-steady-state and continuous attack scenarios.

Table 1
Comparison of the SWaT and WADI datasets and experimental settings.

Parameter	SWaT	WADI
data collection duration (days)	11	16
sampling frequency (sample/sec)	1	1
normal operation duration (days)	1–7	1–14
dimensions (sensors + actuators)	51 (25+26)	124 (65+59)
training set (days)	1–5	1–12
validation set (days)	6–7	13–14
test set (days)	8–11	15–16
removed training samples	initial 16,000	initial 20,000

Table 2
The parameter settings for experiments of our method on the SWaT and WADI datasets.

Parameter	SWaT	WADI
sensor series period threshold L	1e4	1e4
flatness determination slope threshold K	2e−5	1e−6
sliding window size S	128	128
acceptable validation error threshold τ_e	1e−6	1e−5
multiple minimum supports threshold γ	0.95	0.75
multiple minimum supports threshold θ	0.08	0.05

WADI Dataset. The WADI (Ahmed et al., 2017) testbed is equipped with chemical dosing systems, booster pumps, valves, meters, and analyzers, constituting a water treatment, storage, and distribution network. The testbed consists of three processes: the first process is used for raw water collection and storage; the second process distributes water to the storage tank based on the pre-set demand pattern; and the third process serves as the return network, sending water back to the first process. A total of 15 attacks were executed on the WADI testbed, causing disruptions such as water tank leaks and termination of water supply.

This study utilizes the physical properties of the SWaT and WADI datasets. Due to the time required for both testbeds to reach a stable state after operation, the initial samples from the training data are removed to ensure the reliability of the training sets (Goh et al., 2017). Table 1 provides a detailed comparison between the SWaT and WADI datasets, as well as the experimental settings for both datasets in this study.

5.2. Parameters

The parameters γ and θ define multiple minimum supports. To determine the ideal values of these two parameters of our method, we conduct a grid search on the validation set using Eq. (7). Specifically, we set the candidate values of γ to be $[0.6, 0.65, 0.7, 0.75, 0.8, 0.85, 0.9, 0.95, 0.99]$, while the candidate values of θ to be $[0.05, 0.06, 0.07, 0.08, 0.09, 0.1, 0.2, 0.3, 0.4]$. Table 2 displays the final selected ideal results and other parameters for the experiments on the SWaT and WADI datasets of our method.

Simultaneously, to validate our description in Section 4 that the confidence threshold (Eq. (8)) needs to be 1, Table 3 demonstrates the overall anomaly detection performance of our method on the SWaT and WADI datasets when setting the confidence threshold to other values close to 1. In this table, positive detection corresponds to the identification of abnormal samples, and negative detection refers

Table 3
Overall anomaly detection performance on the SWaT and WADI datasets when setting the confidence threshold to other values.

Conf. threshold	SWaT				WADI			
	F1	Precision	Recall	FP	F1	Precision	Recall	FP
1	0.886	0.959	0.823	1783	0.696	0.891	0.571	694
0.999	0.827	0.815	0.839	9644	0.629	0.626	0.632	3771
0.998	0.810	0.777	0.845	12 265	0.596	0.547	0.656	5426
0.997	0.797	0.753	0.846	14 071	0.561	0.490	0.656	6816
0.996	0.235	0.134	0.942	308 278	0.550	0.474	0.656	7271
0.995	0.234	0.133	0.943	310 002	0.539	0.455	0.659	7862

to normal samples. FP represents False Positives. It can be observed that if the confidence is not 1, the discovered relationships become uncertain, indicating potential errors in the rules and reducing the reliability of detection results. We need to ensure that rule $A \rightarrow B$ dictates that whenever predicate set A appears in a transaction, set B must also appear in the same transaction to try to reflect the inherent deterministic nature of ICS.

5.3. Baseline methods

- DDEA (Feng et al., 2019). An invariant rule-based anomaly detection method that utilizes system physical process data. The method proposes two predicate generation strategies: the distribution-driven strategy and the event-driven strategy.
- LSTM-VAE (Park, Hoshi, & Kemp, 2018). A model that combines a Variational Autoencoder (VAE) with a Long Short-Term Memory (LSTM) network. By replacing the feed-forward network of VAE with LSTM, this model introduces temporal dependence of sequential data into VAE.
- MAD-GAN (Li et al., 2019). A multivariate time-series anomaly detection method based on Generative Adversarial Network (GAN). The model uses Long-Short-Term-Memory Recurrent Neural Network (LSTM-RNN) as the basic model in the GAN framework to capture the temporal correlation among multiple time series.
- USAD (Audibert, Michiardi, Guyard, Marti, & Zuluaga, 2020). An encoder–decoder architecture within the framework of adversarial training, combining the advantages of autoencoder and adversarial training. The model incorporates one encoder network and two decoder networks to amplify the reconstruction error of anomalous input data.
- TNC (Tonekaboni, Eytan, & Goldenberg, 2021). A contrastive learning method that exploits the local smoothness of time series. Positive and negative sample pairs are defined through an automated domain determination process, where samples in the same domain are similar, to learn the potential state of non-stationary time series.

For these baseline methods, all experimental settings are based on maximizing adherence to the original papers or open-source codes provided by the authors. Regarding data preprocessing, DDEA does not undergo any processing; LSTM-VAE and USAD use the Min-Max normalization method, while MAD-GAN and TNC employ standardization, transforming the data to have a mean of 0 and a standard deviation of 1.

We select the remaining hyperparameters for deep neural network-based residual discrimination methods based on achieving the optimal F1-Score on the test set apart from the default and author-specified parameter settings. For instance, the threshold J_{th} for residual discrimination is determined following this principle. Table 4 showcases the settings for some hyperparameters of these methods.

Regarding the contrastive learning method TNC, a decoder has been added to the representations learned by the model to reconstruct the input time series. The newly added decoder architecture aligns with the original encoder architecture of TNC, representing an inverse reconstruction process solely in the data dimension.

Table 4
The parameter settings for experiments of deep neural network-based methods on the SWaT and WADI datasets.

Parameter	LSTM-VAE	MAD-GAN	USAD	TNC
batch size	50	128	200	10
window size	128	5	12	4
training iterations	50	100	100	100
optimizer	Adam	AdamW	Adam	Adam
learning rate	1e−3	1e−4	5e−4	1e−3

Table 5
Overall anomaly detection performance comparison results of each baseline method and our approach on the SWaT and WADI datasets.

Method	SWaT			WADI		
	F1-Score	Precision	Recall	F1-Score	Precision	Recall
DDEA	0.857	0.977	0.763	0.629	**0.932**	0.474
LSTM-VAE	0.855	0.946	0.780	0.527	0.801	0.393
MAD-GAN	0.814	**0.994**	0.689	0.320	0.221	**0.577**
USAD	0.821	0.982	0.705	0.489	0.766	0.359
TNC	0.813	0.985	0.692	0.504	0.602	0.434
Our method	**0.886**	0.959	**0.823**	**0.696**	0.891	0.571

DDEA, as a method also utilizing association rule mining techniques, the multiple minimum support thresholds (γ and θ) on the SWaT dataset are 0.9 and 0.32, respectively. While on the WADI dataset, the values for γ and θ are 0.7 and 0.04, respectively. The confidence thresholds for this method are set to 1 on both datasets.

5.4. Evaluation metrics

Our method is evaluated based on three criteria: overall anomaly detection performance, attack identification rate, and algorithm efficiency.

In assessing the overall anomaly detection performance, Precision, Recall, and F1-score metrics are used, where positive detection refers to abnormal samples and negative detection refers to normal samples. Owing to the unique characteristics inherent to ICS, cyberattacks on ICS have the potential to result in severe physical damage, emphasizing the essential importance of the recall metric. In our experimental analysis, we will place particular emphasis on the recall rate.

To evaluate the performance of detecting various attacks, we define two distinct criteria. The first assesses recognition accuracy by setting a threshold P for the recall rate detected by the methods. An attack is considered detected if the detected recall exceeds P. The second criterion evaluates identification latency, defined as the time between attack launch and identification. Here, we set a latency threshold Q, and successful detection occurs if the latency is not greater than Q. The identification rate of different attacks, denoted as $R(P)$ or $R(Q)$, is the ratio of successfully detected attacks to the total number of attacks in both cases. In practice, attacks typically persist, creating a continuous sequence of abnormal data. Therefore, it is acceptable to detect only a partial subset of the continuous abnormal data series (Chen et al., 2021), making the set of threshold P more permissive.

Achieving real-time response is crucial for anomaly detection in ICS. Specifically, to meet the system's real-time performance requirements, the anomaly detection time for each test sample should be less than the data sampling interval of 1 second.

5.5. Experimental results

We derive a total of 137 predicates on the SWaT dataset, leading to 9,006 invariant rules through the mining process. Moreover, the WADI dataset yields 110 predicates, and 10,490 invariant rules are mined.

Overall Anomaly Detection Performance. Table 5 shows the overall results of the baseline methods and the method proposed in this study for anomaly detection on the SWaT and WADI datasets.

The two methods with the highest F1-Score are based on invariant rule mining of physical process data. DDEA achieves the best F1-Score

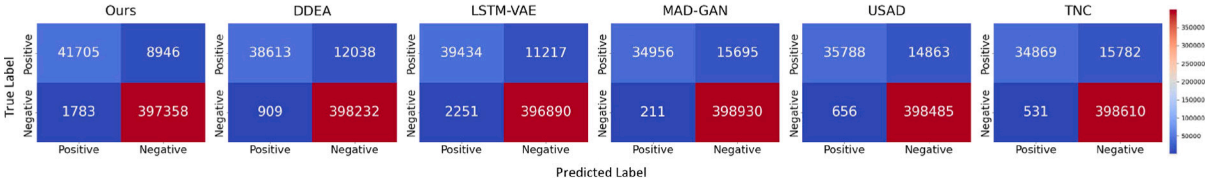

Fig. 3. Confusion matrix of each baseline method and our approach on the SWaT dataset.

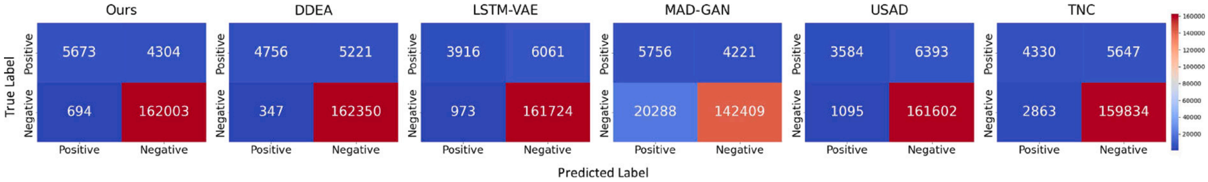

Fig. 4. Confusion matrix of each baseline method and our approach on the WADI dataset.

Table 6
Identification rates of different attacks of each baseline method and our approach on the SWaT and WADI datasets.

Evaluation criteria		$R(.)$ on SWaT						$R(.)$ on WADI					
		Ours	DDEA	LSTM	MAD	USAD	TNC	Ours	DDEA	LSTM	MAD	USAD	TNC
Recall rate threshold P	0.05	**21/36**	**15/36**	**9/36**	**9/36**	**10/36**	**9/36**	**10/15**	**11/15**	**6/15**	**12/15**	**9/15**	**8/15**
	0.1	20/36	15/36	9/36	9/36	10/36	9/36	10/15	10/15	6/15	12/15	9/15	8/15
	0.2	19/36	13/36	9/36	9/36	10/36	9/36	9/15	9/15	6/15	12/15	8/15	8/15
	0.4	16/36	13/36	9/36	9/36	9/36	9/36	8/15	9/15	5/15	9/15	8/15	6/15
	0.6	16/36	12/36	8/36	9/36	9/36	9/36	8/15	7/15	5/15	7/15	6/15	5/15
Recognition delay threshold Q	0(s)	**16/36**	**8/36**	**7/36**	**5/36**	**4/36**	**4/36**	**7/15**	**3/15**	**6/15**	**5/15**	**6/15**	**3/15**
	20(s)	18/36	8/36	7/36	5/36	4/36	4/36	7/15	6/15	6/15	5/15	7/15	5/15
	180(s)	20/36	19/36	8/36	8/36	8/36	8/36	9/15	8/15	6/15	9/15	9/15	6/15

among the baseline methods. Our method outperforms it, with the F1-Score approximately 3% higher on the SWaT dataset and 6.7% higher on the WADI dataset. Despite our method having a lower precision rate, the recall rate improves significantly, reaching 0.823 on the SWaT dataset and 0.571 on the WADI dataset. On the WADI dataset, methods based on invariant rules exhibit a notable superiority in precision, with precision reaching 0.932 for DDEA and 0.891 for our method. In contrast, other baseline methods based on residual discrimination show relatively lower precision due to the generation of many false positives. ICS typically manage extensive industrial processes and require avoiding false alarms that could lead to substantial resource wastage.

In addition to presenting the F1-Score, Precision, and Recall metrics, the performance of the various methods is further evaluated through the confusion matrices on the SWaT and WADI datasets, as depicted in Fig. 3 and Fig. 4. The confusion matrix is comprised of four elements: True Positives (TP) in the top-left, False Negatives (FN) in the top-right, False Positives (FP) in the bottom-left, and True Negatives (TN) in the bottom-right.

On the SWaT dataset, our method demonstrates notable performance in TP, identifying 41,705 anomalous samples, indicating effective anomaly detection. Although the FP count is relatively high at 1,783, the TN and TP suggest its effectiveness in distinguishing normal from abnormal samples. For the WADI dataset, our method identifies 5,673 TP and maintains a low FP count of 694. The TN count of 162,003 indicates that the method performs well in distinguishing normal samples.

Attack identification rate. To further analyze the sensitivity for anomaly detection, Table 6 compares the identification rates of different attacks of the baseline methods and our approach on the SWaT and WADI datasets (note that in this table, LSTM refers to LSTM-VAE, and MAD refers to MAD-GAN).

A total of 36 attacks were conducted on the SWaT testbed. Methods based on invariant rules demonstrate a noticeable superiority in attack identification rate on the SWaT dataset, while other residual-based machine learning methods show similar performance. When the recall rate threshold P is set at 0.05, the baseline methods based on residuals essentially identify the same set of attacks. Compared to methods based on invariant rules, these methods overlook attacks associated with changes in actuator states. In other words, the residual-based methods demonstrate relatively lower sensitivity to changes in actuator states. On the contrary, methods based on invariant rules generate predicates for each category of actuator states, which are then incorporated into the mined rules. As a result, methods based on invariant rules are more prone to detect malicious changes in actuator states.

On the SWaT dataset, when the recall rate threshold P is set at 0.05, our method successfully detects 21 attacks. Among the missed 15 attacks, 12 attacks do not achieve their intended goals or make any real changes to the testbed. The remaining three attacks are related to the states of the water tanks, such as “tank overflow” or “tank underflow”. Our method centers on situations where the trend of sensor reading segments changes and focuses on the slopes of the changing segments, which weakens the perception of the specific flow level of the changing segments.

Notably, our method demonstrates superiority in identifying anomalies with low latency in various attack scenarios. Specifically, on the SWaT dataset, our method can detect 16 attacks with zero delay, which is at least 8 more attacks than what other baseline methods achieve. Moreover, when the latency threshold Q is set to 3 minutes, our method successfully detects a total of 20 attacks. The performance of our method in identifying attacks with low latency shows its capability to provide prompt attack warnings, which helps to secure valuable time for emergency response in ICS.

On the WADI testbed, a total of 15 attacks were launched. Some attacks are covert, and some do not even achieve the attacking intent and expected impact, showing no noticeable effects on sensor readings or actuator states. Consequently, detecting attacks on the WADI dataset is challenging. Despite no significant differences in the identification rates of different attacks among all methods, our approach performs well. Although MAG-GAN shows superiority in the identification recall rate, its low precision diminishes the advantage.

Notably, three attacks on the WADI testbed are stealthy. However, as indicated by Eq. (2), for residual-based anomaly detection methods, it is challenging to describe stealthy attacks without considering the residual threshold J_{th}. In our conducted experiments, we lack information on the error bounds considered for these three stealthy attacks of the WADI dataset. Consequently, we determine the residual threshold J_{th} for these methods solely based on the principle of optimizing the F1-Score on the test set. Therefore, we only compare the performance of detection stealthy attacks for methods based on invariant rules. When the recall rate threshold P is set to 0.05, both the method proposed in this paper and DDEA are able to detect two of the three stealthy attacks. However, our method has a shorter attack identification latency compared to DDEA by two minutes (nearly 120 time steps).

Algorithm Efficiency. The SWaT and WADI datasets have a sampling interval of 1 second. To meet the real-time response requirements of the systems, any anomaly detection method must have a detection time of less than 1 second per sample (Feng et al., 2019). For invariant rule-based anomaly detection methods, the detection time is determined by the time required for predicate generation and rule matching for each sample in the test set. Despite generating a large number of invariant rules – 9,006 on the SWaT dataset and 10,490 on the WADI dataset – our method performs well, with a detection time of 0.002 s per sample. This performance fully meets the real-time detection requirements of the system.

6. Discussion

Residual-based machine learning methods face challenges in detecting stealthy attacks due to noise interference and unclear control boundaries in ICS. Despite the relative stability of sensor reading overall changing patterns, current machine-learning strategies are inadequate in addressing these challenges. Methods based on invariant rules discretize sensor readings and coarse-grain them through the generation of predicates, which can help minimize the problem and reduce the likelihood of false positives.

Invariant rules-based methods often depend on expert knowledge for rule design and extraction. Given the complexity of ICS, even experienced experts may not cover all relevant rules comprehensively, leading to limited coverage. Additionally, many methods discretize sensor readings into non-overlapping intervals, which could miss complex patterns and result in significant information loss.

In contrast, our method uses a novel, data-driven approach that incorporates temporal features of sensor readings. Unlike traditional methods that only consider the current state, our method leverages historical data, providing a more comprehensive view of the system's behavior and improving anomaly detection performance.

The variation-driven strategy takes into account the temporal features between previous and current time segments, making it well-suited for situations where sensor readings in ICS exhibit regular and predictable patterns. Compared to DDEA, our method can capture effective relationships that DDEA may miss.

For example, our method mines out the following rule on the SWaT dataset: {LIT101=(K_1, “high”), MV301=1, P302=2} ⟶ {P401=1, MV101=1, P403=1} which successfully identifies 638 abnormal samples without causing any false positives. However, DDEA fails to detect any of them. The reasons are as follows:

(1) For sensor LIT101, DDEA generates predicates of the form LIT101=K_0 or K_1 or K_2 or K_3, indicating that the clustering category indices of the sensor reading updates Δx_t of LIT101 at different time steps are K_0 or K_1 or K_2 or K_3. Since the method cannot capture the temporal correlations in the dynamic processes of LIT101, DDEA does not mine out any similar rules like:

{LIT101=K_0, MV301=1, P302=2} ⟶ {P401=1, MV101=1, P403=1}

(2) Even if DDEA successfully mines out such similar rules, it remains incapable of identifying all these anomalous samples. This stems from the instabilities in clustering sensor reading updates (Δx_t) under successive time steps, reducing the probability of detecting anomalies. In contrast, the predicates generated by our method exhibit greater stability. Since DDEA only considers the current control state, it is susceptible to noise and measurement errors. Conversely, our method considers sensor readings over segments, thereby reducing the impact of local fluctuations.

However, our method also has shortcomings. The predicates generated by our method are combinations of attributes from two consecutive segments. Consequently, any anomaly in one segment may affect the next, resulting in the generation of two consecutive predicates that are influenced by the attribute of the previous segment. This phenomenon increases the risk of abnormal transmission and leads to a higher false positive rate. Additionally, our method only focuses on the slopes of the changing segments, without considering the specific values within these segments. Including descriptions of the initial and final values of the changing segments might be beneficial.

Table 5 demonstrates that the anomaly detection performance of all the methods on the WADI dataset is worse compared to the SWaT dataset. This result is primarily due to the greater complexity of the WADI dataset. Compared to SWaT, WADI features a larger and more complex system architecture, and the attacks are more covert. Many attack signals accumulate gradually, making their effects less likely to manifest in the system's state over a short period, which further increases the difficulty of detection. Future research could explore techniques for faster detection of attacks in large and complex scenarios. For instance, system partitioning could be considered, with independent invariant rules generated in each zone to enable quicker detection and localization of attacks. Additionally, it would be worth exploring how to hierarchically manage the generated rules by using methods like information gain to prioritize more important rules, aiming to match critical rules first.

7. Conclusion

This paper presents a data-driven, unsupervised ICS anomaly detection method based on invariant rules. Utilizing the proposed variation-driven strategy for predicate generation, we innovatively incorporate temporal features of sensor readings into the generated predicates. Through this approach, we identify invariant rules that take into account the temporal dependencies among physical variables. Additionally, our invariant rule-based method is effective in detecting stealthy attacks and is suitable for ICS that require real-time anomaly detection capabilities. Experimental results on two publicly available datasets demonstrate that our method improves anomaly detection performance compared to other methods.

CRediT authorship contribution statement

Qilin Zhu: Writing – original draft, Software, Methodology, Investigation, Formal analysis. **Yulong Ding:** Writing – review & editing, Supervision, Investigation. **Jie Jiang:** Writing – review & editing, Supervision, Project administration. **Shuang-Hua Yang:** Writing – review & editing, Validation, Funding acquisition, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This research is supported in part by the National Natural Science Foundation of China (Grant No. 92067109, 61873119, 62211530106) and in part by the Shenzhen Science and Technology Program, China (Grant No. ZDSYS20210623092007023, GJHZ20210705141808024).

References

Adepu, S., & Mathur, A. (2016). Distributed detection of single-stage multipoint cyber attacks in a water treatment plant. In *Proceedings of the 11th ACM on Asia conference on computer and communications security* (pp. 449–460).

Agrawal, R., Imieliński, T., & Swami, A. (1993). Mining association rules between sets of items in large databases. In *Proceedings of the 1993 ACM SIGMOD international conference on management of data* (pp. 207–216).

Ahmed, C. M., Palleti, V. R., & Mathur, A. P. (2017). WADI: a water distribution testbed for research in the design of secure cyber physical systems. In *Proceedings of the 3rd international workshop on cyber-physical systems for smart water networks* (pp. 25–28).

Aoudi, W., Iturbe, M., & 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* (pp. 817–831).

Audibert, J., Michiardi, P., Guyard, F., Marti, S., & Zuluaga, M. A. (2020). Usad: Unsupervised anomaly detection on multivariate time series. In *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining* (pp. 3395–3404).

Brigham, E. O. (1988). *The fast Fourier transform and its applications*.

Case, D. U. (2016). Analysis of the cyber attack on the ukrainian power grid. *Electricity Information Sharing and Analysis Center*, *388*, 1–29.

Chen, X., Deng, L., Huang, F., Zhang, C., Zhang, Z., Zhao, Y., et al. (2021). Daemon: Unsupervised anomaly detection and interpretation for multivariate time series. In *2021 IEEE 37th international conference on data engineering* (pp. 2225–2230).

Chen, J., Zhao, C., & Ding, J. (2023). A flexible probabilistic framework with concurrent analysis of continuous and categorical data for industrial fault detection and diagnosis. *IEEE Transactions on Industrial Informatics*, *19*(10), 10578–10590.

Cooley, J. W., & Tukey, J. W. (1965). An algorithm for the machine calculation of complex Fourier series. *Mathematics of Computation*, *19*(90), 297–301.

Dán, G., & Sandberg, H. (2010). Stealth attacks and protection schemes for state estimators in power systems. In *2010 first IEEE international conference on smart grid communications* (pp. 214–219).

Das, T. K., Adepu, S., & Zhou, J. (2020). Anomaly detection in industrial control systems using logical analysis of data. *Computers & Security*, *96*, Article 101935.

Eiras-Franco, C., Martinez-Rego, D., Guijarro-Berdinas, B., Alonso-Betanzos, A., & Bahamonde, A. (2019). Large scale anomaly detection in mixed numerical and categorical input spaces. *Information Sciences*, *487*, 115–127.

Falliere, N., Murchu, L. O., & Chien, E. (2011). W32. stuxnet dossier. *White Paper, Symantec Corp., Security Response*, *5*(6), 29.

Feng, C., Li, T., Zhu, Z., & Chana, D. (2017). A deep learning-based framework for conducting stealthy attacks in industrial control systems. arXiv preprint arXiv:1709.06397.

Feng, C., Palleti, V. R., Mathur, A., & Chana, D. (2019). A systematic framework to generate invariants for anomaly detection in industrial control systems. In *26th annual network and distributed system security symposium*.

Ghaderi, M., Gheitasi, K., & Lucia, W. (2020). A blended active detection strategy for false data injection attacks in cyber-physical systems. *IEEE Transactions on Control of Network Systems*, *8*(1), 168–176.

Goh, J., Adepu, S., Junejo, K. N., & Mathur, A. (2017). A dataset to support research in the design of secure water treatment systems. In *Critical information infrastructures security: 11th international conference, CRITIS 2016, Paris, France, October 10–12, 2016, revised selected papers 11* (pp. 88–99).

Hu, Y., & Chen, Y. (2006). Mining association rules with multiple minimum supports: a new mining algorithm and a support tuning mechanism. *Decision Support Systems*, *42*(1), 1–24.

Keogh, E., Chu, S., Hart, D., & Pazzani, M. (2001). An online algorithm for segmenting time series. In *Proceedings of the 2001 IEEE international conference on data mining* (pp. 289–296).

Kumar, M. (2018). Tsmc chip maker blames wannacry malware for production halt. *The Hacker News*, *7*(8).

Li, D., Chen, D., Jin, B., Shi, L., Goh, J., & Ng, S.-K. (2019). MAD-GAN: Multivariate anomaly detection for time series data with generative adversarial networks. In *Artificial neural networks and machine learning–ICANN 2019: text and time series: 28th international conference on artificial neural networks, munich, Germany, September 17–19, 2019, proceedings, part IV* (pp. 703–716).

Liu, B., Hsu, W., & Ma, Y. (1999). Mining association rules with multiple minimum supports. In *Proceedings of the fifth ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 337–341).

Liu, Y., Ning, P., & Reiter, M. K. (2011). False data injection attacks against state estimation in electric power grids. *ACM Transactions on Information and System Security*, *14*(1), 1–33.

Ljung, L. (1998). *System identification*.

Maiti, R. R., Yoong, C. H., Palleti, V. R., Silva, A., & Poskitt, C. M. (2023). Mitigating adversarial attacks on data-driven invariant checkers for cyber-physical systems. *IEEE Transactions on Dependable and Secure Computing*, *20*(4), 3378–3391.

Maurya, V., Agarwal, R., Kumar, S., & Shukla, S. K. (2022). Epasad: ellipsoid decision boundary based process-aware stealthy attack detector. arXiv preprint arXiv:2204.04154.

Mehmood, M., Baig, Z., & Syed, N. (2024). Securing industrial control systems (ICS) through attack modelling and rule-based learning. In *16th international conference on cOMmunication systems & nETworkS* (pp. 598–602).

Mishra, V. K., Palleti, V. R., & Mathur, A. (2019). A modeling framework for critical infrastructure and its application in detecting cyber-attacks on a water distribution system. *International Journal of Critical Infrastructure Protection*, *26*.

Momtazpour, M., Zhang, J., Rahman, S., Sharma, R., & Ramakrishnan, N. (2015). Analyzing invariants in cyber-physical systems using latent factor regression. In *Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 2009–2018).

Park, D., Hoshi, Y., & Kemp, C. C. (2018). A multimodal anomaly detector for robot-assisted feeding using an lstm-based variational autoencoder. *IEEE Robotics and Automation Letters*, *3*(3), 1544–1551.

Tian, C., & Zhao, C. (2023). Unbiased estimation based multivariate alarm design considering temporal and multimodal process characteristics. *Control Engineering Practice*, *136*, Article 105531.

Tonekaboni, S., Eytan, D., & Goldenberg, A. (2021). Unsupervised representation learning for time series with temporal neighborhood coding. arXiv preprint arXiv:2106.00750.

Wang, M., Sheng, L., Zhou, D., & Chen, M. (2022). A feature weighted mixed naive Bayes model for monitoring anomalies in the fan system of a thermal power plant. *IEEE/CAA Journal of Automatica Sinica*, *9*(4), 719–727.

Wang, M., Zhou, D., & Chen, M. (2023). Hybrid variable monitoring: An unsupervised process monitoring framework with binary and continuous variables. *Automatica*, *147*, Article 110670.

Wang, M., Zhou, D., Chen, M., & Wang, Y. (2020). Anomaly detection in the fan system of a thermal power plant monitored by continuous and two-valued variables. *Control Engineering Practice*, *102*, Article 104522.

Yoong, C. H., Palleti, V. R., Maiti, R. R., Silva, A., & Poskitt, C. M. (2021). Deriving invariant checkers for critical infrastructure using axiomatic design principles. *Cybersecurity*, *4*, 1–24.

Yuan, L., Yu, S., Yang, Z., Duan, M., & Li, K. (2023). A data balancing approach based on generative adversarial network. *Future Generation Computer Systems*, *141*, 768–776.

Zeng, F., Chen, M., Qian, C., Wang, Y., Zhou, Y., & Tang, W. (2023). Multivariate time series anomaly detection with adversarial transformer architecture in the internet of things. *Future Generation Computer Systems*, *144*, 244–255.

Zhang, K., Keliris, C., Parisini, T., & Polycarpou, M. M. (2021). Stealthy integrity attacks for a class of nonlinear cyber-physical systems. *IEEE Transactions on Automatic Control*, *67*(12), 6723–6730.

Zheng, X., Julien, C., Kim, M., & Khurshid, S. (2015). Perceptions on the state of the art in verification and validation in cyber-physical systems. *IEEE Systems Journal*, *11*(4), 2614–2627.

Zhou, C., Huang, S., Xiong, N., Yang, S.-H., Li, H., Qin, Y., et al. (2015). Design and analysis of multimodel-based anomaly intrusion detection systems in industrial process automation. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, *45*(10), 1345–1360.

Qilin Zhu is currently working toward the M.Eng. degree in the Department of Computer Science and Engineering at the Southern University of Science and Technology, Shenzhen, China. She received a B.Eng. degree in cybersecurity from the School of Cyber Science and Engineering at Sichuan University, Chengdu, China. Her current research interests include Cyber–Physical Systems security and physical watermarking.

Yulong Ding received the B.A.Sc. and M.A.Sc. degrees in chemical engineering from Tsinghua University, Beijing, China, in 2005 and 2008, respectively, and the Ph.D. degree in chemical engineering from The University of British Columbia, Vancouver, BC, Canada, in 2012. He is currently a Research Associate Professor with the Shenzhen Key Laboratory of Safety and Security for Next Generation of Industrial Internet and the Department of Computer Science and Engineering, Southern University of Science and Technology, Shenzhen, China. His main interests are Industrial Internet of Things and low-power wide-area networks.

Jie Jiang obtained her Ph.D. in Artificial Intelligence from Delft University of Technology in the Netherlands in 2015. Currently, she holds the position of an Associate Professor in the College of Artificial Intelligence at China University of Petroleum (Beijing). Her research focuses primarily on Internet of Things data mining, time series analysis, and machine learning. She has published more than 20 papers in international conferences and journals.

Shuang-Hua Yang received the Ph.D. degree from Zhejiang University, China, in 1991. He was awarded DSc from Loughborough University in 2014 for his contribution to wireless monitoring research. He is the Head of Department of Computer Science at the University of Reading and the Director of Shenzhen Key Laboratory of Safety and Security for Next Generation of Industrial Internet, Southern University of Science and Technology. His research interests include Cyber–Physical Systems safety and security, Internet of Things. He is a Fellow of IET and InstMC. He is an Associate Editor of the IET Journal Cyber–Physical Systems Theory and Applications.