

ANOMALY DETECTION SYSTEM AND METHOD FOR AN INDUSTRIAL CONTROL SYSTEM

Cross Reference to Related Application(s)

5

The present disclosure claims the benefit of Singapore Patent Application No. 10202101249T filed on 05 February 2021, which is incorporated in its entirety by reference herein.

10 Technical Field

The present disclosure generally relates to an anomaly detection system and method for an industrial control system. More particularly, the present disclosure describes various embodiments of a system and method for detecting anomalies in the industrial
15 control system, such as those due to cyberattacks on the industrial control system.

Background

Modern Industrial Control Systems (ICS) play a vital role in controlling and automating
20 industrial processes in critical public infrastructure, such as water treatment and distribution plants, oil / natural gas plants, power generation plants, and power distribution grids. An ICS typically has three major components – computing systems such as programmable logic controllers (PLCs), communication infrastructure, and physical devices such as sensors and actuators. The communication infrastructure
25 connects the computing systems with the physical devices to perform their functions. A physical process controlled by an ICS, such as water treatment process, evolves over time in accordance with the design of the control algorithms in the PLCs. Cooperating with the sensors and actuators, the PLCs monitor the current system state of the ICS and control the overall system behaviour of the ICS. A combination of
30 PLCs, sensors, and actuators constitutes a distributed control system that operates in real-time, often without interruption and with minimal human involvement. Usually, all components of the ICS are deployed within an air gap, i.e., using an isolated communication network. More recently, advancement in information technology (IT)

has led to the growing openness of ICS towards the corporate network for economic and operational gains.

ICS in public infrastructure have become a target of cyberattacks. The connectivity of
5 ICS to the public network increases the chances of the ICS being subjected to a variety
of cyberattacks. Cyberattacks on the ICS can be defined as intentional actions aimed
at causing undesired effects in the physical world. The primary intention of these
targeted cyberattacks is to disrupt or change the underlying physical process to a state
which does not conform to its design specifications. For example, cyberattacks
10 typically target, directly or indirectly, the physical devices of the ICS by manipulating
the sensor measurements and control signals to the actuators, thereby causing
abnormal changes in the system state and physical process. Such abnormal changes
in the system state and physical process are referred to as anomalies. To mitigate
undesirable consequences in the ICS operation and to maintain ICS safety, such
15 anomalies should be detected and responded to quickly, allowing the cause of the
anomalies to be eliminated before damage to the ICS occurs.

In a real-time cyberattack, an attacker can make use of several strategies to prevent
the early detection by the existing defense mechanisms to achieve their malicious
20 intention. Such sophisticated cyberattacks on the ICS can evade existing defense
mechanisms of the ICS and cause catastrophic damages or failures with significant
impact to the components of the ICS or even lead to service disruption. Timely
detection of such sophisticated cyberattacks and anomalies in the ICS is hence
important to protect the ICS.

25 Many existing cyberattack detection solutions for ICS are extensions of the traditional
network-centric or IT-centric defense system or infrastructure such as Intrusion
Detection Systems (IDS), access control systems, encryption protocols, and firewalls
to prevent intrusions into the ICS. Such solutions are designed for preventing
30 unauthorized access into the ICS, but are inadequate for defending the ICS from
cyberattacks and protecting critical infrastructure against malicious intruders,
especially against insiders.

Some ICS employ anomaly detectors that work in real-time together with the operation of the ICS. There are two types of such anomaly detectors, namely design-centric and data-centric. Design-centric anomaly detectors are based on the design knowledge of the ICS. The design knowledge may include the ICS architecture specifications of each component, computing, and communication infrastructure details. Such design knowledge may be available in operator manuals, vendor specifications and other design documents that are useful during ICS maintenance and upgrade. Additionally, such design knowledge can be coded to ensure the normal operation of the ICS in real-time.

10

One example of a design-centric anomaly detector is the Distributed Attack Detector (DAD), such as described in United States Patent No. 10,911,482. The DAD is built for the SWaT (Secure Water Treatment) testbed plant at the Singapore University of Technology and Design (SUTD). The DAD is generated from plant design based on fundamental laws of physics and chemistry, and manually extracts deterministic rules from the plant design to govern the plant operation. The DAD is able to detect anomalies in real-time when such rules are violated. However, as these rules are manually created, the DAD is only feasible for small plants that contain a few components, such as the SWaT testbed plant. It is not feasible to implement the DAD in large commercial plants, such as city-scale water treatment plants, because of the multitude and variety of heterogeneous components in the plants with complex interrelationships which would be prohibitively complex and time-consuming to model manually.

15

Existing data-centric anomaly detectors combine machine learning algorithms, such as Long Short-Term Memory (LSTM) Network, Recurrent Neural Network (RNN), and Generative Adversarial Network (GAN), etc. with statistical techniques to detect anomalies. However, these data-centric anomaly detectors are based on temporal dependencies across the plant components which limit their applicability to plant components with continuous state variables, such as sensors monitoring the physical and chemical properties of water. These data-centric anomaly detectors fail to learn the behaviours of the complex heterogeneous components in large plants, resulting in an unacceptable number of false alarms.

25

30

Therefore, given the existing vulnerabilities of ICS and threats of cyberattacks to the ICS, and in order to address or alleviate at least one of the aforementioned problems and/or disadvantages, there is a need for an improved anomaly detection system and
5 method for the ICS.

Summary

According to aspects of the present disclosure, there is a system and method for
10 detecting anomalies in an industrial control system. The anomaly detection method comprises: determining state variables of the industrial control system based on a system design of the industrial control system; determining invariants governing the state variables based on the system design; receiving historical measurement data of the state variables of each invariant from the industrial control system; constructing a
15 set of behavioural models for each invariant using a set of machine learning algorithms and the historical measurement data of the respective state variables of the invariant, the behavioural models representing normal behaviour of the respective state variables; predicting measurement data of the state variables of each invariant using the behavioural models and the historical measurement data of the respective state
20 variables; receiving current measurement data of the state variables during operation of the industrial control system; and detecting the anomalies based on deviations between the current measurement data and predicted measurement data of the state variables of each invariant.

25 A system and method for detecting anomalies in an industrial control system according to the present disclosure are thus disclosed herein. Various features, aspects, and advantages of the present disclosure will become more apparent from the following detailed description of the embodiments of the present disclosure, by way of non-limiting examples only, along with the accompanying drawings.

30

Brief Description of the Drawings

Figure 1 is a schematic illustration of an industrial control system.

Figures 2A and 2B are schematic illustrations of a water treatment process.

Figure 3 is an illustration of an attack scenario on a plant performing the water
5 treatment process.

Figures 4A and 4B are schematic illustrations of an anomaly detection system
according to embodiments of the present disclosure.

10 Figure 5 is a table of some invariants and state variables of the water treatment
process.

Figure 6 is an illustration of a transient state of a component of the water treatment
15 process.

Figure 7 is a table of some rule models for the water treatment process.

Figures 8A and 8B are tables of attack scenarios used to evaluate performance of the
anomaly detection system.
20

Figures 9A to 9E are illustrations of evaluation results on the performance of the
anomaly detection system.

Detailed Description

25 For purposes of brevity and clarity, descriptions of embodiments of the present
disclosure are directed to a system and method for detecting anomalies in an industrial
control system, in accordance with the drawings. While aspects of the present
disclosure will be described in conjunction with the embodiments provided herein, it
30 will be understood that they are not intended to limit the present disclosure to these
embodiments. On the contrary, the present disclosure is intended to cover alternatives,
modifications and equivalents to the embodiments described herein, which are
included within the scope of the present disclosure as defined by the appended claims.

Furthermore, in the following detailed description, specific details are set forth in order to provide a thorough understanding of the present disclosure. However, it will be recognized by an individual having ordinary skill in the art, i.e., a skilled person, that the present disclosure may be practiced without specific details, and/or with multiple
5 details arising from combinations of aspects of particular embodiments. In a number of instances, well-known systems, methods, procedures, and components have not been described in detail so as to not unnecessarily obscure aspects of the embodiments of the present disclosure.

10 In embodiments of the present disclosure, depiction of a given element or consideration or use of a particular element number in a particular figure or a reference thereto in corresponding descriptive material can encompass the same, an equivalent, or an analogous element or element number identified in another figure or descriptive material associated therewith.

15

References to “an embodiment / example”, “another embodiment / example”, “some embodiments / examples”, “some other embodiments / examples”, and so on, indicate that the embodiment(s) / example(s) so described may include a particular feature, structure, characteristic, property, element, or limitation, but that not every
20 embodiment / example necessarily includes that particular feature, structure, characteristic, property, element or limitation. Furthermore, repeated use of the phrase “in an embodiment / example” or “in another embodiment / example” does not necessarily refer to the same embodiment / example.

25 The terms “comprising”, “including”, “having”, and the like do not exclude the presence of other features / elements / steps than those listed in an embodiment. Recitation of certain features / elements / steps in mutually different embodiments does not indicate that a combination of these features / elements / steps cannot be used in an embodiment.

30

As used herein, the terms “a” and “an” are defined as one or more than one. The use of “/” in a figure or associated text is understood to mean “and/or” unless otherwise indicated. The term “set” is defined as a non-empty finite organization of elements that

mathematically exhibits a cardinality of at least one (e.g. a set as defined herein can correspond to a unit, singlet, or single-element set, or a multiple-element set), in accordance with known mathematical definitions. The recitation of a particular numerical value or value range herein is understood to include or be a recitation of an
5 approximate numerical value or value range.

Embodiments of the present disclosure describe a system and method for detecting anomalies in an industrial control system (ICS) 100. The general architecture of an ICS 100 is shown in Figure 1. The ICS 100 is application-specific with a layered
10 architecture that integrates computing and communication components with physical devices to control the underlying industrial or physical process 110. The components of the ICS 100 are distributed across three main layers of the ICS 100, namely the physical layer 120, control layer 130, and supervisory control layer 140.

15 The physical layer 120 includes physical devices, such as sensors 122 and actuators 124, that measure and control the physical process 110. The physical layer 120 may optionally include remote devices such as input/output (IO) devices. The control layer 130 includes programmable logic controllers (PLCs) and/or remote terminal units (RTUs) (collectively referred to as PLCs 132). It will be appreciated that the PLCs 132
20 are generally programmable in a variety of suitable programming languages such as ladder logic, structured text, and functional blocks. The supervisory control layer 140 includes a Supervisory Control and Data Acquisition (SCADA) system 142 and workstation 144 for overall monitoring and control of the physical process 110, data historian 146 for recording inputs to and outputs from the PLCs 132, and Human-
25 Machine Interfaces (HMIs) 148.

The communication infrastructure in the ICS 100 is divided into two levels – Level 0 and Level 1. Level 0 refers to the communication network between the physical layer 120 and control layer 130. Specifically, the Level 0 communication network uses a ring
30 network topology to enable communications among the sensors 122, actuators 124, and PLCs 132. Level 1 refers to the communication network between the control layer 130 and supervisory control layer 140. Specifically, the Level 1 communication network uses a star network topology to enable communications among the PLCs 132,

SCADA system 142 and workstation 144, data historian 146, and HMIs 148. Communications in the Level 0 and Level 1 communication networks are carried out through the secure communication channels with appropriate security protocols, as will be readily understood by the skilled person.

5

The sensors 122 and actuators 124 in the physical layer 120 report and modify the current state of the physical process 110 based on the control signals transmitted and received from the PLCs 132 in the control layer 130. In an example of the ICS 100 being a water treatment plant, the sensors 122 communicate to the PLCs 132 the physical and chemical properties of water, such as water level in a tank, water flow rate, pH value, and the chlorine content. Based on the received data from the sensors 122, the PLCs 132 issue control signals or commands to the actuators 124, such as to open or close a valve, and to run or stop a pump. The sensors 122 and actuators 124 represent state variables of the ICS 100. A state variable measures one state attribute of one component of the ICS 100, such as the water level in a tank. The supervisory control layer 140 enables the operators of the ICS 100 to monitor the physical process 110 and control actions taken by the PLCs 132.

The communications infrastructure of an ICS 100 may be connected to an external communications network, but such connections render the ICS 100 susceptible to cyberattacks. Such cyberattacks may compromise the Level 0 and/or Level 1 communication networks. Each communication link is considered as an attack point in the ICS 100 and once a communication link is compromised, an attacker can, for example, send false sensor data to the PLCs 132 or bypass the PLCs 132 and directly control the actuators 124. Examples of the ICS 100 and attack scenarios are described below.

An exemplary ICS 100 is a water treatment plant to perform a water treatment process. One example of a water treatment plant is the SWaT testbed plant at SUTD for performing the water treatment process 200, as shown in Figure 2A. The SWaT testbed plant is a fully operational and scaled down water treatment plant that produces 5 gallons of filtered water per minute. Its architecture replicates that of large modern water treatment plants found in cities.

- As shown in Figure 2A, the water treatment process 200 can be divided into 6 process stages – Stage 1 (Raw Water Input 210), Stage 2 (Chemical Dosing 220), Stage 3 (Ultrafiltration 230), Stage 4 (Dechlorination 240), Stage 5 (Reverse Osmosis 250), and Stage 6 (Backwash 260). Each of the 6 process stages is equipped with a set of sensors 122 and a set of actuators 124 that are controlled by a PLC 132. The sensors 122 include water level sensors (labelled “LIT”), chemical property sensors (labelled “AIT”), flow rate indicators or flow sensors (labelled “FIT”), and differential pressure indicators (labelled “DPIT”). The actuators 124 include motorized valves (labelled “MV”) and pumps (labelled “P”). The PLCs 132 across the 6 process stages are communicative with each other to share information about the state information of their corresponding sensors 122 and actuators 124 for executing the water treatment process 200 in coordination.
- Stage 1 controls the inflow of raw water to be treated by opening or closing a motorized valve MV101 that connects an external water source to the inlet of a raw water tank T101. Water from the tank T101 is pumped via a chemical dosing station in Stage 2 to an ultrafiltration (UF) unit in Stage 3 for the removal of solid materials in the water. In Stage 3, using an UF feed pump, the filtered water is transferred to a tank T401 in Stage 4 to remove excess chlorine, which is used to disinfect the water. In Stage 4, the dechlorination unit, such as an ultraviolet (UV) unit, treats the water through an UV chlorination process to remove the excess chlorine from the water. The dechlorinated water is then transferred to the reverse osmosis (RO) filtration unit in Stage 5 for a two-stage RO process to remove inorganic impurities in the water. In Stage 5, the dechlorinated water is passed through the two-stage RO unit and the filtered water is then stored in tanks T601 and T602 in Stage 6. The water in the tank T602 is utilized for cleaning the filter membranes in the UF unit in Stage 3 through a backwash process in Stage 6. The water in the tank T601 is recycled to the tank T101 in Stage 1.
- The communication infrastructure of the SWaT plant is not connected to any other external network. However, there is a wireless network through which the sensors 122, actuators 124, and PLCs 132, and other devices communicate with each other. An attacker, such as of the insider type, can exploit vulnerabilities in the network protocols

or PLC firmware to compromise the communication links, such as between the PLC 132 and sensors 122, PLC 132 and actuators 124, and PLC 132 and SCADA system 142. Through the compromised communication links, the attacker can inject false information or spoof the values of measurement data transferred from one or more
 5 sensors 122 to one or more PLCs 132 to alter the process flow of the water treatment process 200 and/or to damage the physical devices.

An exemplary simple attack scenario on the SWaT plant to disrupt the water treatment process 200 is described as follows. The specific process stage or operation in Stage
 10 1 (Raw Water Input 210) is shown in Figure 2B. Stage 1 has a set of physical devices including the corresponding sensors 122, actuators 124, and PLCs 132 that define a set of state variables n , as shown in Equation 1. n_1 , n_2 , and n_3 are continuous state variables, and n_4 and n_5 are discrete state variables. The state space of each component in n , i.e. n_1 to n_5 , can be expressed as V_1 to V_5 , respectively, as shown in
 15 Equations 2 to 6.

Equation 1

$$n = \{n_1: LIT101, n_2: FIT101, n_3: FIT201, n_4: P101, n_5: MV101\}$$

20 Equation 2

$$V_1 = \{x_1 \in \mathbb{R}\}$$

Equation 3

$$V_2 = \{x_2 \in \mathbb{R}\}$$

25

Equation 4

$$V_3 = \{x_3 \in \mathbb{R}\}$$

Equation 5

30 $V_4 = \{ON, OFF\}$

Equation 6

$$V_5 = \{OPEN, CLOSED\}$$

The motorized valve MV101 controls the flow of water to the raw water tank T101. The rate of inflow is measured the at flow sensor FIT101 and the water level in the tank T101 is measured by the level sensor LIT101. The pump P101 pumps water from the tank T101 to the next stage (Stage 2 or Chemical Dosing 220). The overall operation of Stage 1 is monitored and controlled by PLC1. The tank T101 is labelled with four markers, namely LL, L, H, and HH, as shown in Figure 2B. Each marker represents a particular value of the water level in the tank T101 and is used by the PLC1 to control the state of the motorized valve MV101 and pump P101. During normal plant operation, the motorized valve is MV101 is closed when the water level is above H, and MV101 is open when the water level falls below L. The pump P101 is switched on when the water level is above L to transfer water to Stage 2, and P101 is switched off when the water level is below L.

15

If an attacker wants to underflow the tank T101, MV101 should be kept closed even though water level in T101 is below L. In this attack scenario as shown in Figure 3, the attack is launched against the level sensor LIT101 in the state where LIT101 is at 550 mm and MV101 is closed. During normal operation, the water level measurement from LIT101 falls gradually as water is pumped out by P101 from T101 to Stage 2. MV101 remains closed until LIT101 reaches L which is 500 mm. Due to the attack against LIT101 at the 550 mm level, whenever the measurement from LIT101 falls below 550 mm, it is increased by 10 mm and the manipulated value is reported to PLC1. Thus, when the actual water level reaches L (500 mm), the manipulated value of 510 mm is sent to PLC1. In response, PLC1 does not send the control signal to open MV101. The water in T101 is continuously pumped out to Stage 2 and over time, T101 will not have enough water for the subsequent stages, thereby affecting the entire water treatment process 200. This simplified attack scenario illustrates how an attacker can potentially affect the SWaT plant and cause anomalies in the the water treatment process 200 through cyberattacks by spoofing the measurement data of the sensors

30

122.

In representative or exemplary embodiments of the present disclosure with reference to Figures 4A and 4B, there is an anomaly detection system 300 and computer-implemented or computerized anomaly detection method for detecting anomalies in an ICS 100. The system 300 may be referred to as "AICrit". The method can be performed on a computer system having one or more processors and various steps of the computerized method are performed in response to non-transitory instructions operative or executed by the processors. The non-transitory instructions are stored on a memory of the computer system and may be referred to as computer-readable storage media and/or non-transitory computer-readable media. Non-transitory computer-readable media include all computer-readable media, with the sole exception being a transitory propagating signal per se.

The anomaly detection system 300 is installed as part of the supervisory control layer 140 of the ICS 100 and is communicative with the data historian 146 for ensuring that the physical process 110 is under control and runs normally. The system 300 ensures the control behaviour integrity of the ICS 100 during operation through the interaction among its physical components including the sensors 122 and actuators 124. The system 300 relies on measurement data obtained from the data historian 146 for the anomaly detection process and does not require architectural changes or modifications in the control code of the PLCs 132. In many embodiments, the ICS 100 is a large commercial water treatment plant which is built and operated based on the standard design principles and guidelines of the United States National Institute of Standards and Technology (NIST).

Following the above example of the SWaT plant, the ICS 100 can be mathematically modelled as a multivariate and distributed system with n components, where the state of each component at time t is represented by the state variable $x_i(t)$, $1 \leq i \leq n$. The n -dimensional state vector that denotes the state of the ICS 100 can be expressed as $X(t)$ as shown in Equations 7 and 8.

30

Equation 7

$$X(t) = [x_1(t), x_2(t), \dots, x_i(t), \dots, x_n(t), i = 1, 2, \dots, n]$$

Equation 8

$$X(t) = X_c(t) \cup X_d(t)$$

Given that the ICS 100 has continuous and discrete state variables, $X(t)$ can be expressed as Equation 8, where $X_c(t)$ and $X_d(t)$ denote, respectively, the observable state of the continuous and discrete state variables. For example, the state space of a motorized valve (e.g. MV101) is discrete whereas that of the flow rate of water (e.g. FIT101) is continuous. The anomaly detection system 300 is designed based on this mathematical model of the ICS 100 and the spatiotemporal dependencies among the state variables. More specifically, the system 300 includes a set of behavioural models that can represent the behaviour of the state variables including the physical interactions among them, as described further below.

As shown in Figures 4A and 4B, the system 300 includes a data pre-processor module 310 for determining the state variables of the ICS 100 based on a system design 312 of the ICS 100, and for determining invariants governing the state variables based on the system design 312. The system design 312 may include a piping and instrumentation diagram (P&ID) of the ICS 100. The invariants are designed based on the relationships among the physical and/or chemical properties of the underlying physical process 110 operative in the ICS 100. An invariant defines a condition that must hold in a given state of the ICS 100. This condition describes the expected behaviour of the state variables of the invariant under normal operation of the ICS 100. An anomaly in the physical process 110 can be detected based on deviation from the invariant or expected behaviour of the corresponding state variables.

25

An exemplary set of invariants and corresponding state variables of Stage 1 of the water treatment process 200 operative in the SWaT plant is shown in Figure 5. These invariants describe and model the expected or normal behaviour of the respective state variables, and they can be categorized into two types. The first type of invariants captures the behaviour of only continuous state variables, particularly the functional relationship or dependency across them, such as INV 1.1, INV 1.2, INV 1.4, and INV 1.5. The second type of invariants captures the behaviour of both continuous and

30

discrete state variables, particularly the functional relationship or dependency across them, such as INV 1.3.

Further, the data pre-processor module 310 is configured for receiving historical measurement data of the state variables of each invariant from the ICS 100. As shown in Figure 4A, the data pre-processor module 310 is communicative with the data historian 146 for receiving the historical measurement data of the continuous and discrete state variables. Additionally, the data pre-processor module 310 is configured for receiving current measurement data of the state variables during operation of the ICS 100, wherein the current measurement data can be used to detect the anomalies. The communication link between the data historian 146 and data pre-processor module 310 is one-way such that the data pre-processor module 310 can only receive measurement data from the data historian 146 but cannot return or alter any data in the data historian 146.

The data pre-processor module 310 may receive the measurement data at predefined intervals, such as once every second. The data pre-processor module 310 may convert the measurement data into a compatible computer-readable format, such as numeric format. For example, the state variables of the actuators 124 may be converted to "1" for "CLOSED" or "OFF", and "2" for "OPEN" or "ON". Further, there is a transient state that occurs for a short duration, such as a few seconds, when an actuator 124 changes state, such as opening a motorized valve that is initially closed. This transient state may also be represented as "1".

The system 300 further includes a set of machine modelling modules for constructing a set of behavioural models for each invariant, the behavioural models representing normal behaviour of the respective state variables of the invariant. More specifically, the machine modelling modules are configured to construct the behavioural models for each invariant using a set of machine learning algorithms and the historical measurement data of the respective state variables of the invariant. The machine modelling modules utilize the historical measurement data corresponding to normal operation of the ICS 100 to learn its process dynamics and control strategies.

Additionally, machine learning algorithms are utilized together with design knowledge from the invariants for effective modelling of the system behaviour of the ICS 100. As mentioned above, there are transient states when actuators 124 change states. For example, the opening and closing of the motorized valve MV101 is not immediate.

- 5 When PLC1 sends the control signal to close MV101, the inflow rate measured by FIT101 will not be instantly zero. Figure 6 illustrates the impact of the transient state of MV101 in Stage 1. When MV101 receives the control signal to close, there is some non-zero inflow rate for a few seconds, causing the measurement data from LIT101 to increase by a small value δ from H. Such transient states have an impact
- 10 on the water treatment process 200, such as time delay in the water flow to reach zero and the increase or decrease in the water level of tanks from their level markers. Such operational discrepancies caused by the transient states can be learned and modelled in the behavioural models using the data-centric machine learning algorithms together with the design-centric invariants. This improves accuracy of the behavioural models
- 15 for more precise anomaly detections and minimizing false positives.

- The system 300 further includes an anomaly detection module 320 that runs concurrently with the ICS 100 for continuously monitoring the physical process 110 to effectively detect anomalies. The anomaly detection module 320 includes a data
- 20 analyzer module 322 that receives the current measurement data of the state variables from the data pre-processor module 310 for detecting the anomalies. The anomaly detection module 320 operates over the spatiotemporal dependencies among the state variables and hence the data pre-processor module 310 ensures that correct and valid measurement data is sent to the data analyzer module 322. In case of any
- 25 delay, glitches, or noise in the measurement data, the data pre-processor module 310 may be configured to neglect the affected set of measurement data and sends the previous unaffected set to the data analyzer module 322 for continuous monitoring.

- The data analyzer module 322 is configured for predicting measurement data of the
- 30 state variables of each invariant using the behavioural models and the historical measurement data of the respective state variables. The predicted measurement data represents the expected or normal behaviour of the state variables. The data analyzer module 322 compares between the current measurement data and predicted

measurement data of the state variables of each invariant. The data analyzer module 322 detects anomalies based on deviations between the current measurement data and predicted measurement data. The anomaly detection module 320 includes a responder module 324 that generates an alert if an anomaly is detected by the data
5 analyzer module 322. This alert reports to the operator of the ICS 100 the anomaly together with the components that are likely under threat or attack. Further, through the analysis of deviations between the actual and predicted behaviours, semantics of the abnormal system state can be acquired by the operators for forensic analysis.

10 In many embodiments, the behavioural models include a first behavioural model 330 for each of the invariants governing only continuous state variables, and/or a second behavioural model 340 for each of the invariants governing both continuous and discrete state variables. Correspondingly, the machine modelling modules include a first machine modelling module 350 for constructing the first behavioural models 330,
15 and/or a second machine modelling module 360 for constructing the second behavioural models 340. The first machine modelling module 350 may be referred to as "AiBox" and the second machine modelling module 360 may be referred to as "RuleBox".

20 The first behavioural model 330 models the behaviour of the continuous state variables of the respective invariant based on the interactions with dependent state variables to predict the time-dependent behaviour. The second behavioural model 340 models the behaviour of the continuous and discrete state variables of the respective invariant, specifically the dependencies across the state variables, based on the non-
25 linear correlation and behavioural patterns of the state variables in response to a given control signal.

In constructing the first behavioural models 330, the invariants that govern only the continuous state variables are assumed to be a regression problem. The first
30 behavioural models 330 are constructed by the first machine modelling module 350 using a suitable machine learning algorithm, such as but not limited to a multilayer perceptron neural network (MLP). The construction of an exemplary first behavioural model 330 for the invariant INV 1.3 in Figure 5 is described below.

The invariant INV 1.3 represents the dependency of the water level sensor LIT101 with the flow sensors FIT101 and FIT201 measuring the inflow and outflow rates, respectively. Accordingly, the value of LIT101 at time t can be estimated using its previous value $LIT101(t-1)$ and the current values of the inflow rate $FIT101(t)$ and outflow rate $FIT201(t)$. Further, as the transient state of MV101 impacts the process dynamics, the invariant INV 1.3 is modified according to Equation 9. d_1 and d_2 correspond to the time delay considered in the flow sensors FIT101 and FIT202, respectively, in order to measure a more accurate value of LIT101 at time t .

Equation 9

$$LIT101(t) = LIT101(t - 1) + \eta[FIT101(t - d_1) - FIT201(t - d_2)]$$

However, practically deriving the relationship among the state variables in Equation 9 is a complex task due to the existence higher order and non-linear correlations. Further, the heterogeneous nature of the state variables with different operational range complicates the modelling process. To overcome such complexity, a data-centric approach using machine learning algorithms such as MLP is adopted for automated feature learning using a training dataset 314 comprising the historical measurement data. A non-linear transformation function $f(\cdot)$ is defined in Equation 10 to estimate the measurement value of a continuous state variable.

Equation 10

$$LIT101(t) = f(LIT101(t - 1), FIT101(t - d_1), FIT201(t - d_2))$$

In some embodiments, Equation 10 is defined using a neural network such as MLP. Using the training dataset 314, the first machine modelling module 350 receives the values of independent state variables as inputs to the MLP neural network and trains the first behavioural model 330 to predict the value of the dependent state variable with minimal error.

With reference to Figure 4B, the first machine modelling module 350 includes a model generator module 352 for training a plurality of MLP models using the historical

measurement data of the respective state variables of each invariant and by varying hyperparameters of the MLP neural network. The model generator module 352 generates the MLP models by varying the hyperparameters such as number of hidden layers, hidden units, learning rate, momentum, and activation functions. The hyperparameters may be varied or fine-tuned using a grid search algorithm. The model generator module 352 then trains the MLP models using the historical measurement data contained in the training dataset 314.

The first machine modelling module 350 further includes a model pruner module 354 for pruning the MLP models and selecting the best MLP model for the invariant. More specifically, the a model pruner module 354 calculates, for each MLP model, a statistical score based on actual and predicted measurement data of the respective state variables derived from the historical measurement data and the MLP model. The statistical scores may be calculated using Theil's U1 Statistic (TU1) as shown in Equation 11, where $x(t)$ and $\hat{x}(t)$ are the actual and predicted value of the state variable at t^{th} timestamp, k is the number of samples, and H_p denotes the set of hyperparameters. The model pruner module 354 then selects the MLP model with the minimum statistical score as the first behavioural model 330, i.e. the best MLP model, for the invariant.

20

Equation 11

$$TU1 = \sqrt{\frac{[\sum_{t=1}^k (\hat{x}(t) - x(t))^2]}{[\sum_{t=1}^k (x(t))^2]}}; wrt H_p$$

The selected first behavioural model 330 is used to predict the measurement data that is compared against the actual current measurement data for detecting anomalies based on deviations between the current and predicted measurement data. These deviations may also be referred to as residuals. During normal operation of the ICS 100, there would be slight deviations which would be detected as anomalies but are likely to be false positives. The computed deviations should be within an acceptable range τ to minimize the occurrence of false positives, such that only deviations outside of the acceptable range will be detected as anomalies. As the performance of the anomaly detection system 300 is sensitive to the acceptable range for the deviations,

30

a poor choice of the acceptable range might lead to an unacceptable number of either false positives or false negatives.

In some embodiments, constructing the first behavioural model for each invariant
 5 further includes calculating the acceptable range for the deviations based on a statistical distribution of historical deviations, wherein the historical deviations are derived from the historical measurement data from normal operation of the ICS 100. The historical deviations or residuals $r(t)$ are generated as a sequence of random variables. However, during an attack, there would be non-random dynamics in these
 10 variables which can be effectively characterised through a statistical technique.

The first machine modelling module 350 may include a scorer module 356 for analyzing the statistical distribution of historical deviations to thereby calculate the acceptable range. In one embodiment, this analysis is done using a cumulative sum
 15 control chart (CUSUM) technique. CUSUM is a stateful approach that computes the cumulative sum of the residual sequence to detect discrepancies in their magnitudes. The upper control limit (UCL) and lower control limit (LCL) are defined in Equations 12 and 13 for calculating the acceptable level of positive and negative side deviations of the residual sequence, i.e. the upper and lower limits of the acceptable range for the
 20 deviations.

Equation 12

$$UCL = \text{Max}(P(t))$$

25 Equation 13

$$LCL = \text{Min}(N(t))$$

Equation 14

$$P(t) = \text{Max}(0, r(t) - \tau - b) \forall t, 1 \leq t \leq T$$

30

Equation 15

$$N(t) = \text{Min}(0, r(t) - \tau - b) \forall t, 1 \leq t \leq T$$

$P(t)$ and $N(t)$, expressed in Equations 14 and 15, $P(t)$ and $N(t)$ are the allowable positive and negative side deviations, respectively, in the acceptable range τ , and b corresponds to the allowable slack. The anomaly detection module 320 is configured to detect anomalies when the probability of a deviation is greater than a predefined threshold computed from $r(t)$, i.e. when $P(t)$ is more than UCL or $N(t)$ is less than LCL .

During normal operation of the ICS 100, there may be a few short-term deviations due to operational discrepancies, sensor noises, and temporal glitches. Such deviations should not be accounted as anomalies since they are not due to cyberattacks. As shown in Equations 16 and 17, two additional parameters – a time window and an allowable number of deviations within the time window – may be introduced to minimize such false positives. In some embodiments, constructing the first behavioural model 330 for each invariant may further include determining the time window (w_s) and the allowable number of deviations (c_s) within the time window. An anomaly function $a(t)$ detects an anomaly at time t when the number of deviations in the time window exceed the allowable number.

Equation 16

$$f(t) = \begin{cases} 1, (r(t) > UCL, \text{ or } r(t) < LCL) \\ 0, \text{ otherwise} \end{cases}$$

Equation 17

$$a(t) = \begin{cases} \text{Anomaly}, (\prod_{t=1}^{w_s} f(t)) > c_s \\ \text{Normal}, \text{ otherwise} \end{cases}$$

In constructing the second behavioural models 340, the invariants that govern both the continuous and discrete state variables are assumed to be a classification problem, since each discrete state variable have only two states. The second behavioural models 340 are constructed by the second machine modelling module 360 using a suitable machine learning algorithm, such as but not limited to a decision tree algorithm. With reference to Figure 4B, the second machine modelling module 360 includes a rule generator module 362 for generating a plurality of rule models for the

state variables of the invariant using the machine learning algorithm and the historical measurement data of the respective state variables. A sample of the rule models or conditions generated for the actuators 124 used in Stage 1 of the water treatment process 200 is shown in Figure 7.

5

In one embodiment, a second behavioural model 340 is constructed for the invariant INV 1.1 in Figure 5. The invariant INV 1.1 represents the dependency of the water level sensor LIT101 with the motorized valve MV101 connected to the inlet of the tank T101. More specifically, the rule generator module 362 generates several rule models or conditions for MV101 to be open or closed based on the LIT101 measurement data using the decision tree algorithm and the historical measurement data contained in the training dataset 314.

The second machine modelling module 360 further includes a rule pruner module 364 for pruning the rule models and selecting the best rule model for the invariant. More specifically, the rule pruner module 364 calculates a weight for each rule model and selects the rule model with the maximum weight as the second behavioural model 340, i.e. the best rule model, for the invariant. The selected second behavioural model 340 is used to predict the measurement data that is compared against the actual current measurement data for detecting anomalies based on deviations between the current and predicted measurement data.

In some situations, a few false positives may be observed while constructing the second behavioural model 340 representing the interrelationship across the flow sensor and its dependent motorized valve, as well as across the chemical property sensor and its dependent chemical dosing pump. The first case of the flow sensor and motorized valve is due to the transient state of the motorized valve. To minimize the false positives, constructing the second behavioural model 340 for the invariant may include determining a time delay before the deviations are detected as anomalies. In other words, an alert would be raised if the deviations are detected for longer than the time delay. The second case is due to the non-uniformity in the “on” time of the chemical dosing pump injecting the chemicals into the raw water. To minimize the false positives, the amount of chemicals injected is mapped with the “on” time of the

chemical dosing pump, and then the generation of the rule models is carried out using information available in pump specifications.

The effectiveness of anomaly detection system 300 was assessed experimentally by designing and launching cyberattacks in real-time against the operational SWaT plant. A cyberattack refers to a sequence of actions that influence the behaviour of one or more cyber- or physical components to alter the water treatment process 200. It is assumed that the primary objective of an attacker is to cause the underlying process to move into an undesirable state. The attacker is assumed to have complete access to the design specifications of each component of the plant under attack and thus can manipulate one or more sensor measurements without being detected. Such attacks are considered to be stealthy. The built-in safety mechanisms, such as mechanical breaker and alarm generator in the plant, are assumed to not have been activated by the attacker. The attacker is assumed to be able to compromise only a small subset of the sensors 122 and/or actuators 124 for launching coordinated multipoint attacks. The attacker is assumed to be able to assert complete control over the compromised PLCs 132, i.e. the firmware or the control codes of the PLCs 132 can be accessed and modified by the attacker in real-time. Further, the sensor measurements from the compromised PLCs 132 are considered legitimate and recorded in the data historian 146. The plant is assumed to be disconnected from other external network and the attacker is of the insider type.

As mentioned above, a cyberattack is a sequence of actions initiated against an operational plant with a motive to affect its state. An attack model can be represented as a three tuple $\langle A_{type}, A_{point}, A_{obj} \rangle$ where, A_{type} denotes the type of attack launched through the finite set of points (A_{point}) with an intention (A_{obj}). An attack type, for example, can be of any kind readily known to the skilled person. An attack point refers to the components compromised for launching the specific attack. The intention of the attacker (A_{obj}) can be specified as a statement such as "Damage Pump P101". The success of an attacker in realizing the intention depends on the effectiveness of the defense mechanisms utilized to protect the plant as well as the technical knowledge and proficiency of the attacker. In these experiments, the attackers are researchers who possess complete knowledge of the SWaT plant and thus the effectiveness of the

anomaly detection system 300 is analyzed in terms of timely detection of attacks before the intention is realized.

Prior to the launch of an attack, the plant was brought to a steady state. No attacks
5 were launched during the plant's transient state, i.e. during the startup or shutdown. The state of the plant is ensured by monitoring the quantity of water produced at various time instances and the absence of warnings related to water level sensors in the SCADA system 142. Further, to avoid the effect of one attack on another, a minimum interval of 5 minutes is maintained between successive attacks for the plant
10 to stabilize from the impact of an attack. The procedure for launching attacks is described follows.

1. Identify the memory location of the target state variable where the PLC 132 receives the measurement data.
 - 15 2. Compromise the link between the PLC 132 and corresponding sensor 122 for an attack on the Level 0 communication network.
 3. Compromise the link between the PLC 132 and SCADA system 142 for an attack on the Level 1 communication network.
 - 20 4. Generate an alert when the measurement data of a state variable is significantly different from the measurement data received by the PLC 132. Without an in-built attack detection logic in the PLC 132, the manipulated data is assumed to be the actual state while the control actions of the actuators 124 are initiated based on the manipulated data.
- 25 The experiments were conducted to evaluate the performance of the anomaly detection system 300 over selected existing anomaly detectors designed for the SWaT plant. The system 300 was implemented on a desktop-grade computer with using Python 3.7, Keras deep learning, and scikit-learn libraries., the same configuration system was used for all the experiments from constructing the behavioural models to
30 testing on the live plant. This also shows the computational efficiency of the system 300 in running multiple behavioural models 330,340 concurrently on the desktop-grade computer for the detection of anomalies.

Two modes of evaluation were used in these experiments, namely static and live, to demonstrate the effectiveness of the anomaly detection system 300 against selected existing anomaly detectors. In the static evaluation mode, the experiments were based on measurement data previously collected from the SWaT plant under a set of 26 attacks listed in Figure 8A. The collected measurement data may be referred to as the static dataset. In the live evaluation mode, the behavioural models were trained and the system 300 was evaluated using live measurement data recorded in the data historian 146 during live operation of the SWaT plant and under a set of 13 attacks listed in Figure 8B.

In both evaluation modes, an attack is detected only when the alert is raised by the corresponding behavioural model constructed for an invariant that governs a particular state variable, or indirectly by its dependent state variable. For example, if there is no behavioural model constructed for an invariant that governs a different pressure indicator DPIT301. But an attack on DPIT301 will have a cascading effect on motorized valves MV301, MV302, MV303, or MV304, and pumps P301 or P302. When the alert is raised by the corresponding behavioural model monitoring these actuators 124, the attack is detected. If another behavioural model which monitors state variables unrelated to DPIT301 raises the alert, the attack is considered as a false positive.

In the static evaluation mode, the use of the static dataset is a traditional way of assessing the performance of anomaly detectors. Unlike in practical situations, the attacks last for a particular duration during which the measurement data can be collected for the theoretical experiments. Each measurement data record is a multivariate time series and is considered as a single data point. Anomaly detection is carried out for every time instance using a scoring technique, such as the detection rate expressed in Equation 18, to quantify the performance of the anomaly detectors based on the correctly detected anomalies. T_p represents the number of correctly detected as anomalies and F_n represents the number of incorrectly detected anomalies, i.e. anomalies that were incorrectly classified as “normal”.

Equation 18

$$D_r = \frac{T_p}{T_p + F_n}$$

The performance of the anomaly detection system 300 was compared against existing anomaly detectors that use machine learning algorithms such as Support Vector Machine (SVM), Deep Neural Network (DNN), 1D Convolutional Neural Network (1D-CNN), TABOR (a graphical model-based approach for anomaly detection), and an anomaly detection system based on Logical Analysis of Data (LAD-ADS). The static evaluation mode was conducted by launching the set of 26 attacks shown in Figure 8A and the results are shown in Figure 9A.

It was observed that the anomaly detection system 300 has an overall detection rate above 90% for the attacks against the actuators 124. This is due to the accurate modelling of interactions among the continuous and discrete state variables through the integration of design-centric (invariants) and data-centric (machine learning algorithms) by the machine modelling modules 350,360. Similarly, for attacks targeting the sensors 122, the overall detection rate was above 75% excluding the stealthy attacks in attack scenarios 3 and 16. These stealthy attacks were detected after they were removed, i.e. the plant moving from the attack state to normal state. The reason for the late detection of stealthy attacks is the rate at which the state variables are manipulated. As described above, the behaviour models that monitor the behaviour of the sensors 122 may be affected by additional parameters such as upper control limit (*UCL*), lower control limit (*LCL*), time window (*w_s*), and allowable number of deviations (*c_s*) within the time window. As the rate of change was minimal – for example the water level measured by LIT101 increased or decreased by 1 mm per second – the plant behaviour was assumed to be normal and no alert was raised after the attack launch and during the attack. However, once the attack has been removed, there was a notable difference in the behaviour of such sensors 122, and hence alerts were raised.

It was also observed that there were similar performances between the anomaly detection system 300 and the LAD anomaly detector in several attack scenarios because both operate over the interactions among the plant components for the anomaly detection process. However, in a few attack scenarios (7, 17, and 27), the anomaly detection system 300 dominated the LAD anomaly detector due to the

inclusion of sensor behaviour in the system 300 that enables it to more effectively detect anomalies.

In the live evaluation mode, the anomaly detection process was conducted on the operational SWaT plant in real-time and this presented challenges unlike the static evaluation mode which used a static or benchmark dataset in a simulated environment. Attackers in the live evaluation mode make use of several strategies to prevent the early detection to achieve their intention. The live evaluation mode was conducted in two sub-modes to evaluate the performance of the anomaly detection system 300.

The first sub-mode was conducted by operating the plant in normal conditions to evaluate the performance of the system 300 in terms of false positives. The second sub-mode was conducted by launching the set of 13 attacks shown in Figure 8B against the plant in real-time to evaluate the performance of the system 300 in terms of attack detection time.

The performance of the anomaly detection system 300 was compared against existing anomaly detectors such as the design-centric Distributed Attack Detector (DAD) and an earlier MLP-based data-centric anomaly detector that operates over temporal dependencies of state variables.

The SWaT plant has several modes of operation that are frequently shifted based on operational and domain constraints and service demands. For example, there can be different operation modes for public holidays and normal days. Such dynamic nature affects the behaviour of the plant components which has a considerable impact on the underlying process dynamics. For example, the chemical dosing pump in a water treatment plant will be operated at a variable speed based on the pH value of incoming raw water. Such changes in the process dynamics cannot be accounted as anomalies. An anomaly detector that adheres to the dynamic operational characteristics of the SWaT plant should not generate alarms during the state changes that occurred during operation mode shifts.

In the first sub-mode for evaluation in terms of false positives, the SWaT plant was operated continuously for 2 hours under normal conditions. The SWaT plant was

operated initially in automated mode with a default set of configurations. Subsequently, it was shifted to the manual mode by increasing the water storage capacity of the UF feed tank T301 from 800-1000 mm to 250-1000 mm. The results from this first sub-mode are shown in Figures 9B and 9C.

5

It was observed from Figure 9B that none of the anomaly detectors raised false alarms, i.e. there were no false positives, during operation of the SWaT plant in automated mode. During operation of the SWaT plant in manual mode, the MLP-based anomaly detector had a false positive rate of 0.87%.

10

It was observed from Figure 9C that there were abnormal spikes in the performance of the MLP-based anomaly detector in the manual mode that were absent for the anomaly detection system 300 also operating in the manual mode. The abnormal spikes were correspond to the false alarms from the MLP-based anomaly detector and had occurred due to the nature of its training dataset which has LIT301 sensor values in the range 800-1000 mm. As this range was altered to 250-1000 mm during the shift to the manual mode, false alarms were generated due to a lack of behavioural patterns for the altered range.

20 These abnormal spikes were absent for the anomaly detection system 300 because it has incorporated design knowledge in the form of the invariants. Although the range was altered, the dependency of LIT301 and its correlated state variables is retained in the plant by design. Thus, the anomaly detectors with design knowledge, i.e. the system 300 and DAD, did not generate any false alarms. Another reason for the zero
25 false positive rate of the system 300 is careful tuning of the parameters described above for the construction of the first behavioural models 330 as well as considering the effect of the transient state of actuators 124 on the continuous state variables. Further, the residual sequence generated during the normal plant operation was close to zero. This shows that the behavioural models constructed in the system 300 are
30 able to estimate the system state closer to the actual system state.

In the second sub-mode for evaluation in terms of attack detection time, the 13 attacks as shown in Figure 8B were launched against the SWaT plant at different time instances and the results are shown in Figure 9D.

- 5 It was observed that 11 out of the 13 attacks were successfully detected by the anomaly detection system 300, but the attacks under scenarios A1.1 and A6.1 were not detected by the system 300. This was because both attacks were designed to exploit the vulnerabilities in the SCADA workstation 144 to monitor the status of LIT101 and P101, and the attacks do not have any physical impact in the water treatment
10 process 200 flow in the SWaT plant.

It was also observed from the successfully detected attacks both single-point and multi-point attacks were detected at a faster rate – approximately 2 to 32 seconds from attack launch – before the attacker's intention was achieved. Single-point attacks of
15 constant type were detected in less than 10 seconds while it took 32 seconds to detect multi-point attacks against the Stage 1 components. Generally, the detection of stealthy multi-point attacks is challenging as the attacker compromises or spoofs more than one other state variable to avoid the detection. For example in attack scenario A5, the attacker attempted to flood tank T101 by compromising valve MV101 and flow
20 sensor FIT101. Initially, during attack launch, there was no significant physical impact on plant operation. However, after 32 seconds from the attack launch, when the water level of tank T101 crossed H+50 mm, the alert was raised by the second behavioural model 340 that monitors the behaviour of valve MV101. There is a buffer above the H+50mm level and the attacker intention was realised only when the water level
25 crossed that buffer. Further, there is an alarm generator in case of tank overflow. Since the system 300 raised the alert before the actual tank overflow can happen, the attack was unsuccessful in realizing its intention.

The alerts on the anomalies were only raised by the behavioural models that monitor
30 the targeted state variables or their correlated state variables. This enables localization of the anomalies and facilitates the plant operator to identify the components under threat as well as to locate the area of impact of an attack on the plant, especially if the plant is geographically dispersed. The plant operator or automated plant protection

mechanisms can then take appropriate actions to mitigate the anomalies and prevent component damage.

The performance comparison of the anomaly detection system 300 against DAD and the MLP-based anomaly detector is shown in Figure 9E. It was observed that the detection ability of the MLP-based anomaly detector is limited only to the continuous state variables since it operates over temporal dependencies. Further, due to a lack of design knowledge, the MLP-based anomaly detector had attack detection times that were much higher than DAD and the system 300. The system 300 requires more time compared to DAD, especially in detecting attacks against continuous state variables, because of the impact of the parameters introduced to minimize false positives due to temporal glitches. More importantly, the system 300 is able to avoid false positives and accurately detect the attacks in minimal time before the attacker's intention was achieved, indicating that the interactions among the sensors 122 and actuators 124 were learned effectively by the behavioural models.

Therefore, the anomaly detection system 300 described herein integrates a design-centric approach using the invariants together with a data-centric approach using the machine-learned behavioural models to learn and model the spatiotemporal dependencies among plant components for accurate and timely detection of process anomalies. The experiments described above showed that the anomaly detection system 300 is able to effectively detect cyberattacks in minimal time and with zero or near-zero false positives, thus achieving improved performance in anomaly detections. Although the experiments were performed on the SWaT plant which is a scaled down water treatment plant, the experiment results are encouraging and the anomaly detection system 300 can be installed in larger commercial plants or ICS 100, such as city-scale water treatment plants. The system 300 can also be applicable for ICS 100 in other domains such as power, energy, and transportation.

In some embodiments, the anomaly detection system 300 continues to learn the plant behaviour in real-time during operation of the ICS 100. The behavioural models may include a primary set and an identical secondary set of behavioural models for each invariant. In one example, there is a primary first behavioural model 330 and a

secondary first behavioural model 330 for the invariant 1.3. In another example, there is a primary second behavioural model 340 and a secondary second behavioural model 340 for the invariant 1.1. The pair of primary set and secondary set of behavioural models may be referred to as the "Twin Detector" system.

5

When the anomaly detection system 300 is deployed with the "Twin Detector" system, the current measurement data and the primary set of behavioural models are used by the anomaly detection module 320 to detect the anomalies. At the same time, the secondary set of behavioural models is trained by the machine modelling modules 10 350,360 using the current measurement data. Anomalies detected by the primary set of behavioural models are not learned by the secondary set of behavioural models. The secondary set of behavioural models is thus continuously trained with new measurement data in real-time while the primary set of behavioural models stagnates as they are being used to detect anomalies using real-time measurement data. The 15 machine modelling modules 350,360 may replace the primary set of behavioural models (which are less mature) with the trained secondary set of behavioural models (which are more mature), and continue training of the secondary set of behavioural models. This replacement may be done at variable or predefined intervals, such as daily, weekly, or monthly, which may depend on plant operational demand. As more 20 mature behavioural models are continually being deployed, the ability of the anomaly detection system 300 to detect anomalies is constantly improving, such that false positives can be further decreased towards zero and success rate of anomaly detection can be further increased.

25 In the foregoing detailed description, embodiments of the present disclosure in relation to a system 300 and method for detecting anomalies in an ICS 100 are described with reference to the provided figures. The description of the various embodiments herein is not intended to call out or be limited only to specific or particular representations of the present disclosure, but merely to illustrate non-limiting examples of the present 30 disclosure. The present disclosure serves to address at least one of the mentioned problems and issues associated with the prior art. Although only some embodiments of the present disclosure are disclosed herein, it will be apparent to a person having ordinary skill in the art in view of this disclosure that a variety of changes and/or

modifications can be made to the disclosed embodiments without departing from the scope of the present disclosure. Therefore, the scope of the disclosure as well as the scope of the following claims is not limited to embodiments described herein.

Claims

1. An anomaly detection system for detecting anomalies in an industrial control system, the anomaly detection system comprising:
- 5 a data pre-processor module configured for:
- determining state variables of the industrial control system based on a system design of the industrial control system;
- determining invariants governing the state variables based on the system design; and
- 10 receiving historical measurement data of the state variables of each invariant from the industrial control system; and
- receiving current measurement data of the state variables during operation of the industrial control system;
- a set of machine modelling modules configured for:
- 15 constructing a set of behavioural models for each invariant using a set of machine learning algorithms and the historical measurement data of the respective state variables of the invariant, the behavioural models representing normal behaviour of the respective state variables; and
- 20 an anomaly detection module configured for:
- predicting measurement data of the state variables of each invariant using the behavioural models and the historical measurement data of the respective state variables; and
- 25 detecting the anomalies based on deviations between the current measurement data and predicted measurement data of the state variables of each invariant.
2. The anomaly detection system according to claim 1, wherein the machine modelling modules comprise:
- 30 a first machine modelling module for constructing a first behavioural model for each of the invariants governing only continuous state variables; and

a second machine modelling module for constructing a second behavioural model for each of the invariants governing both continuous and discrete state variables.

5 3. The anomaly detection system according to claim 2, wherein the continuous and discrete state variables measure states of sensors and actuators, respectively, of the industrial control system.

4. The anomaly detection system according to claim 2 or 3, wherein the machine
10 learning algorithms comprise a multilayer perceptron neural network for constructing the first behavioural models.

5. The anomaly detection system according to claim 4, wherein said constructing,
by the first machine modelling module, the first behavioural model for each invariant
15 comprises:

training a plurality of multilayer perceptron models using the historical measurement data of the respective state variables and by varying hyperparameters of the multilayer perceptron neural network;

calculating, for each multilayer perceptron model, a statistical score
20 based on actual and predicted measurement data of the respective state variables derived from the historical measurement data and the multilayer perceptron model; and

selecting the multilayer perceptron model with the minimum statistical score as the first behavioural model for the invariant.

25

6. The anomaly detection system according to claim 5, wherein the hyperparameters of multilayer perceptron neural network are fine-tuned using a grid search algorithm.

30 7. The anomaly detection system according to claim 5 or 6, wherein the statistical scores are calculated using Theil's U1 Statistic.

8. The anomaly detection system according to any one of claims 4 to 7, wherein constructing, by the first machine modelling module, the first behavioural model for each invariant further comprises calculating an acceptable range for the deviations based on a statistical distribution of historical deviations.

5

9. The anomaly detection system according to claim 8, wherein constructing, by the first machine modelling module, the first behavioural model for each invariant further comprises analysing the statistical distribution of historical deviations using a cumulative sum control chart technique to thereby calculate the acceptable range.

10

10. The anomaly detection system according to any one of claims 4 to 9, wherein constructing, by the first machine modelling module, the first behavioural model for each invariant further comprises determining a time window and an allowable number of deviations within the time window.

15

11. The anomaly detection system according to claim 2 or 3, wherein the machine learning algorithms comprise a decision tree algorithm for constructing the second behavioural models.

20 12. The anomaly detection system according to claim 11, wherein said constructing, by the second machine modelling module, the second behavioural model for each invariant comprises:

generating a plurality of rule models for the state variables of the invariant using the decision tree algorithm and the historical measurement data of the respective state variables;

25

calculating a weight for each rule model; and

selecting the rule model with the maximum weight as the second behavioural model for the invariant.

30 13. The anomaly detection method according to claim 11 or 12, wherein constructing, by the second machine modelling module, the second behavioural model for each invariant further comprises determining a time delay before the deviations are detected as anomalies.

14. The anomaly detection system to any one of claims 1 to 13, further comprising a twin detector system, the twin detector system comprising a primary set of the behavioural models and an identical secondary set of the behavioural models for each invariant.
15. The anomaly detection system according to claim 14, wherein:
- the anomaly detection module is configured for detecting the anomalies using the primary set of behavioural models and the current measurement data during operation of the industrial control system; and
 - the machine modelling modules are configured for training the secondary set of behavioural models using the current measurement data.
16. The anomaly detection system according to claim 15, wherein the machine modelling modules are configured for:
- replacing the primary set of behavioural models with the trained secondary set of behavioural models; and
 - continuing training of the secondary set of behavioural models.
17. The anomaly detection system according to claim 16, wherein the machine modelling modules are configured for replacing the primary set of behavioural models at predefined intervals.
18. A computerized anomaly detection method for detecting anomalies in an industrial control system, the anomaly detection method comprising:
- determining state variables of the industrial control system based on a system design of the industrial control system;
 - determining invariants governing the state variables based on the system design;
 - receiving historical measurement data of the state variables of each invariant from the industrial control system;
 - constructing a set of behavioural models for each invariant using a set of machine learning algorithms and the historical measurement data of the

respective state variables of the invariant, the behavioural models representing normal behaviour of the respective state variables;

predicting measurement data of the state variables of each invariant using the behavioural models and the historical measurement data of the respective state variables;

receiving current measurement data of the state variables during operation of the industrial control system; and

detecting the anomalies based on deviations between the current measurement data and predicted measurement data of the state variables of each invariant.

19. The anomaly detection method according to claim 18, wherein the behavioural models comprise:

a first behavioural model for each of the invariants governing only continuous state variables; and

a second behavioural model for each of the invariants governing both continuous and discrete state variables.

20. The anomaly detection method according to claim 19, wherein the continuous and discrete state variables measure states of sensors and actuators, respectively, of the industrial control system.

21. The anomaly detection method according to claim 19 or 20, wherein the machine learning algorithms comprise a multilayer perceptron neural network for constructing the first behavioural models.

22. The anomaly detection method according to claim 21, wherein constructing the first behavioural model for each invariant comprises:

training a plurality of multilayer perceptron models using the historical measurement data of the respective state variables and by varying hyperparameters of the multilayer perceptron neural network;

calculating, for each multilayer perceptron model, a statistical score based on actual and predicted measurement data of the respective state

variables derived from the historical measurement data and the multilayer perceptron model; and

selecting the multilayer perceptron model with the minimum statistical score as the first behavioural model for the invariant.

5

23. The anomaly detection method according to claim 22, wherein the hyperparameters of multilayer perceptron neural network are fine-tuned using a grid search algorithm.

10 24. The anomaly detection method according to claim 22 or 23, wherein the statistical scores are calculated using Theil's U1 Statistic.

25. The anomaly detection method according to any one of claims 21 to 24, wherein constructing the first behavioural model for each invariant further comprises
15 calculating an acceptable range for the deviations based on a statistical distribution of historical deviations.

26. The anomaly detection method according to claim 25, wherein constructing the first behavioural model for each invariant further comprises analysing the statistical
20 distribution of historical deviations using a cumulative sum control chart technique to thereby calculate the acceptable range.

27. The anomaly detection method according to any one of claims 21 to 26, wherein constructing the first behavioural model for each invariant further comprises
25 determining a time window and an allowable number of deviations within the time window.

28. The anomaly detection method according to claim 19 or 20, wherein the machine learning algorithms comprise a decision tree algorithm for constructing the
30 second behavioural models.

29. The anomaly detection method according to claim 28, wherein constructing the second behavioural model for each invariant comprises:

generating a plurality of rule models for the state variables of the invariant using the decision tree algorithm and the historical measurement data of the respective state variables;

calculating a weight for each rule model; and

5 selecting the rule model with the maximum weight as the second behavioural model for the invariant.

30. The anomaly detection method according to claim 28 or 29, wherein constructing the second behavioural model for each invariant further comprises
10 determining a time delay before the deviations are detected as anomalies.

31. The anomaly detection method according to any one of claims 18 to 30, wherein the behavioural models comprise a primary set and an identical secondary set of behavioural models for each invariant.

15

32. The anomaly detection method according to claim 31, comprising:

detecting the anomalies using the primary set of behavioural models and the current measurement data during operation of the industrial control system; and

20 training the secondary set of behavioural models using the current measurement data.

33. The anomaly detection method according to claim 32, comprising:

25 replacing the primary set of behavioural models with the trained secondary set of behavioural models; and

continuing training of the secondary set of behavioural models.

34. The anomaly detection method according to claim 33, wherein the primary set of behavioural models is replaced at predefined intervals.

30

35. A non-transitory computer-readable storage medium storing computer-readable instructions that, when executed, cause a computer system to perform the computerized method according to any one of claims 18 to 34.