energies

MDPI

Article

Risk Data Analysis Based Anomaly Detection of Ship Information System

Bowen Xing *,†, Yafeng Jiang, Yuqing Liu and Shouqi Cao †

College of Engineering Science and Technology, Shanghai Ocean University, Shanghai 201306, China; yfjiang@shou.edu.cn (Y.J.); yqliu@shou.edu.cn (Y.L.); sqcao@shou.edu.cn (S.C.)

* Correspondence: bwxing@shou.edu.cn; Tel.: +86-176-2186-9324

† These authors contributed equally to this work.

Received: 1 October 2018; Accepted: 25 November 2018; Published: 4 December 2018

Abstract: Due to the vulnerability and high risk of the ship environment, the Ship Information System (SIS) should provide 24 hours of uninterrupted protection against network attacks. Therefore, the corresponding intrusion detection mechanism is proposed for this situation. Based on the collaborative control structure of SIS, this paper proposes an anomaly detection pattern based on risk data analysis. An intrusion detection method based on the critical state is proposed, and the corresponding analysis algorithm is given. In the Industrial State Modeling Language (ISML), risk data are determined by all relevant data, even in different subsystems. In order to verify the attack recognition effect of the intrusion detection mechanism, this paper takes the course/roll collaborative control task as an example to carry out simulation verification of the effectiveness of the intrusion detection mechanism.

Keywords: cybersecurity; intrusion detection; risk data analysis; signal attack; ship information system

1. Introduction

With the constant intellectualization of industrial equipment, the working performance of a Supervisory Control And Data Acquisition (SCADA) system turns out to be highly dependent on the accuracy and security of communication among the units in closed-loops [1]. However, such systems are normally intricate and fragile, and several striking examples have confirmed that even well-protected SCADA systems can be ultimately crashed [2–6]. Therefore, the cyber-physical security of SCADA systems will be more and more important [7], which has been described by IEEE: "In contrast to cyber security, the goal of cyber-physical security is to protect the whole cyber-physical system, which uses widespread sensing, communication and control to operate safely and reliably." [8]. Normally, according to the different focus points, the study of cyber-physical security can be divided into defense [9,10], detection [11,12], and maintenance(including repair, reconstitution, etc.) [3,13,14]. Here, we mainly focus on the anomaly detection topic in the cyber-physical security of SCADA. Actually, several effective anomaly detection methods have already been proposed in the anomaly detection area such as system modeling [15–20] and data-based analysis [20–25], which should always accept a compromise in the modeling uncertainty and data complexity. Besides model-based and coupled data-based intrusion detection, some intrinsic properties of SCADA are considered in detection. In [26], a methodology that uses information extracted from Radio Frequency (RF) features to identify changes was proposed. Meanwhile, S.-M.Jung and J.-G. Song et al.presented an idle-time measurement system in data spoofing detection [27]. Actually, these solutions bring a new perspective to anomaly detection, and the applications for these theories are restricted more or less, which cannot be overcome. However, in [28], an innovative approach based on the concept of critical state analysis and state proximity was presented: attacks can be detected by a set of critical rules, which are formulated in

the Industrial State Modeling Language (ISML). Such rules are based on all related data that are not confined to the coupled data. Therefore, the anomaly detection mode we propose in this paper is based on the risk data, which are obtained by the analysis of critical data.

2. Model Description

As a typical SCADA system, the Ship Information System (SIS) is widely used in the connection between each ship electronic system, which are spread across the whole ship. As the data in SIS are designed to be closed and inflexible, firewall updating and off-line detection are difficult to implement. Furthermore, for a ship on a long voyage, any physical or informational damage may cause an irreparable breakdown, which leads to a helpless situation.

Although different ships have different functions [29–35], all definitions found in the literature for SIS have one key feature in common. As shown in Figure 1, this defining feature is that SIS is composed of several independent subnetworks and a total ship communication network, which can exchange information (reference input, plant output, control input, etc.) among subnetworks and systems. The architecture of SIS is similar to each normal SCADA, which is listed in the following.

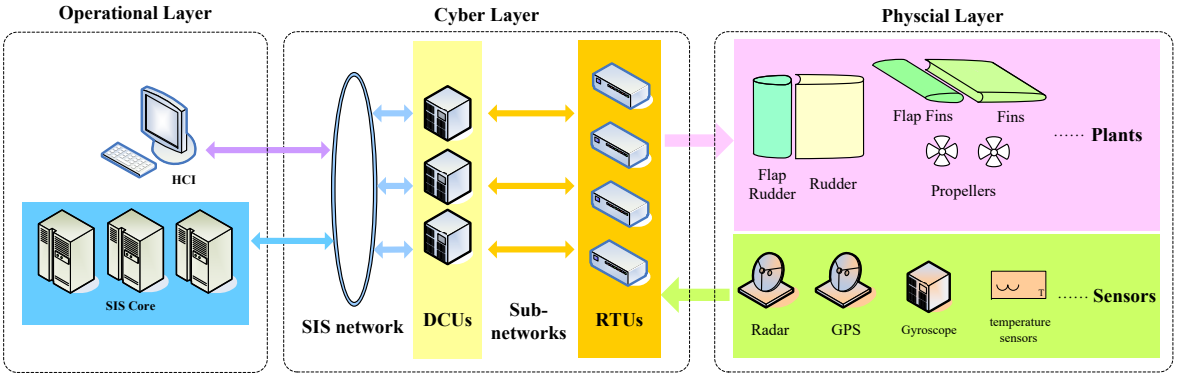

Figure 1. Interactions of a Ship Information System (SIS). DCU, Distributed Controller Unit; RTU, Remote Terminal Unit.

2.1. Structure

2.1.1. Components

Operational Units

Due to the different command types of the mission and situation, a supervisory control command can be released by a user through the Human Computer Interface (HCI) or SIS core. The SIS core provides a common environment hosting the majority of the ship's applications on a redundant infrastructure, whose targeted hardware location is designed to be transparent to the applications. Some basic control commands are released by the core automatically and spontaneously, such as fire detection, anti-rolling control, etc. Meanwhile, real-time assistant decision support is also done by the core and submitted to HCI.

Distributed Controller Units

The Distributed Controller Units (DCUs) in SIS are networked and interfaced with the SIS network; their duty is to monitor and control every closed-loop system in the ship by coding predefined sequences and control algorithms.

Remote Terminal Units

The Remote Terminal Units (RTUs) in SIS serve as the interface point to DCUs and a variety of analog and digital sensors and actuators. Every RTU is connected with its closest DCU. The telemetry hardware structure of RTU has the capability of sending digital sensor data to the DCU and receiving digital commands from it.

It should be mentioned that, in order to keep the security and anti-damage capability of SIS, each RTU is connected to two DCUs simultaneously, but only one is activated during the running process; such a dual-station structure is widely used in SCADA. For each $DCUi-1$, its backup DCU is denoted as $DCUi-2$.

2.1.2. Networks

SIS Network

As a backbone network of the ship information system, the fundamental layering rule of the SIS network is to add maximum DCUs in the ship environment, which leads to a dual-ring network being chosen in the design of the SIS network. Every DCU is led into the SIS network proximally.

Subnetworks

As introduced in [35], numerous DCUs are connected in the SIS network; meanwhile, each of them has the responsibility for dozens of RTUs, which constitute an underlying subnetwork.

Due to the limited space, more design details and examples of implementations can be seen in [29,36].

For SIS, due to different emphases, there exist two structures of SIS, which are the cooperative control structure (Str_{CC}) and the hierarchical task structure (Str_{HT}). Str_{CC} is used to describe the implementation method of task control in SIS, while Str_{HT} is for the description of task capabilities.

2.2. Cooperative Control Structure of SIS

As the connection principle between DCU and RTU is mere proximity instead of task relation, which has reduced the difficulty of planning and laying out networks greatly, but has increased the complexity of every control process in SIS, most missions in SIS are designed to be completed by several DCUs cooperatively. For definiteness and without loss of generality, the cooperative control mode of SIS is shown in Figure 2.

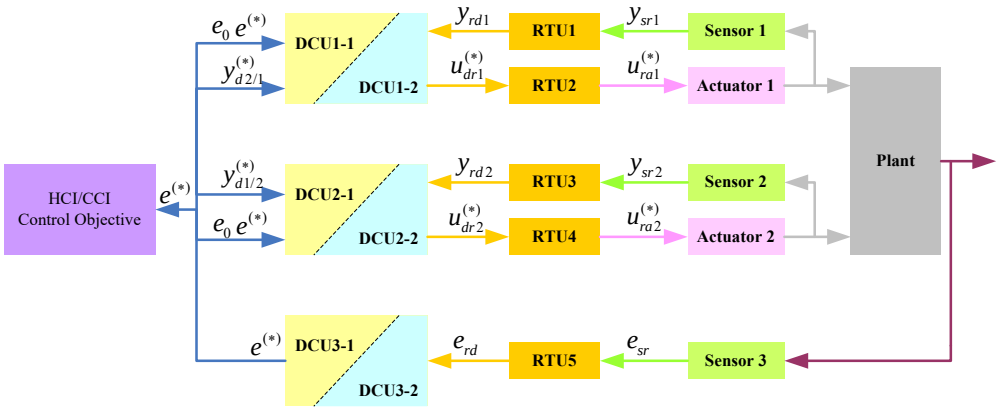

Figure 2. Cooperative control mode of SIS.

As our attention in this paper is the cybersecurity issue in SCADA, to facilitate distinction, all the data in different lines are relabeled, as is listed in Table 1.

The control objective e_0 should be performed by Actuator 1 and Actuator 2 cooperatively. $e_{sr}(k)$ is the control feedback of the control objective at step k, and these data are sampled by a global sensor, Sensor 3. Here, Sensor 1 and Sensor 2 are regarded as local sensors, which are used to sample the running conditions of Actuator 1 and Actuator 2, respectively, and these data are finally received by DCU2-1(2)and DCU1-1(2). Taking DCU1-1 as an example, due to the cooperative structure with the coupling relationship with DCU2-1(2), the control output u_{dr1} is determined by the preset distributed control algorithm based on e, $y^*_{d2/1}$, and y_{rd1}. As shown in Table 1, if the data are related to a backup DCU, they would be marked with superscript "$*$", such as $e^*(k)$, $y^*_{d1/2}(k)$, $y^*_{d2/1}(k)$, etc.

Table 1. Notations of Figure 2.

Annotation	Notations
e_0	Control objective
$e_{sr}(k)$	Data from Sensor 3
$e_{rd}(k)$	Data sent by RTU 5 according to $e_{sr}(k)$
$e^{(*)}(k)$	Data sent by DCU 3-1(2) according to $e_{rd}(k)$
$y_{sr1}(k)$	Output of Actuator 1 sampling by Sensor 1
$y_{rd1}(k)$	Data sent by RTU 1 according to $y_{sr1}(k)$
$y^{(*)}_{d1/2}(k)$	Data sent by DCU 1-1(2) according to $y_{rd1}(k)$
$y_{sr2}(k)$	Output of Actuator 2 sampling by Sensor 2
$y_{rd2}(k)$	Data sent by RTU 3 according to $y_{sr2}(k)$
$y_{d2/1}(k)$	Data sent by DCU 2-1 according to $y_{rd2}(k)$
$y^{(*)}_{d2/1}(k)$	Data sent by DCU 2-1(2) according to $y_{rd2}(k)$
$u^{(*)}_{dr1}(k)$	Control command for Actuator 1 by DCU 1-1(2)
$u^{(*)}_{ra1}(k)$	Data sent by RTU 2 according to $u^{(*)}_{dr1}(k)$
$u^{(*)}_{dr2}(k)$	Control command for Actuator 2 by DCU 2-1(2)
$u^{(*)}_{ra2}(k)$	Data sent by RTU 4 according to $u^{(*)}_{dr2}(k)$

In this paper, taking DCU1 for example, we propose a cooperative state space control model, which is established in Equations (1) and (2).

$$\begin{aligned} X(k+1) &= AX(k) + BM_{\sigma 1}(k) \begin{bmatrix} \hat{u}^{(*)}_{dr1}(k) \\ \hat{y}^{(*)}_{d2/1}(k) \end{bmatrix} \\ M_{\omega 1}(k) &\begin{bmatrix} e^{(*)}(k) \\ y_{rd1}(k) \\ y^{(*)}_{d2/1}(k) \end{bmatrix} = CX(k) \end{aligned} \tag{1}$$

$$u^{(*)}_{dr1}(k) = F_{dr1}(\hat{u}^{(*)}_{dr1}(k), \hat{y}^{(*)}_{d2/1}(k), y^{(*)}_{d2/1}(k)) \tag{2}$$

where $X(k)$ is the quantity of state and $\hat{u}^{(*)}_{dr1}(k)$ and $\hat{y}^{(*)}_{d1/2}(k)$ are the obtained output for Actuator 1 and 2 from numerical calculation, respectively. As DCU 1 only focuses on the control of Actuator 1, $\hat{y}^{(*)}_{d1/2}(k)$ has no physical application, but is only used in the revision of $\hat{u}^{(*)}_{dr1}(k)$ obtained by amending function $F_{dr1}(*)$. Meanwhile, the communication access of $DCUi$ at step k is denoted as $M_{\sigma i}(k)$ and $M_{\omega i}(k)$. As the communication between DCU and RTU belongs to a kind of multi-channel real-time mechanism, the access is granted constantly. However, the communication among DCUs and the HCI/CCIcore should follow an access specification. For example, if DCU1-1 gains access to publish data $y^*_{d1/2}$ at step k, we have:

$$M_{\omega 2}(k) = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{3}$$

Equation (1) provides a new model to analyze the control progress of cooperative control mode based on data, and the control output of one actuator would be corrected by the execution conditions of its collaborators. $F_{dr1}(*)$ can be determined by a neural network, fuzzy, or linear regression algorithm, etc, and as a representative example in [37], we propose a variable universe fuzzy algorithm to correct the deviation between two cooperative rudders.

As our research interest in this paper is cybersecurity, the transmission process of a data is worth more serious study than its solution process. The optimization of $F_{dr1}(\ )$ will be researched in the future; in this paper, we assume that each DCU in the cooperative control mode would figure out a suitable control output for the corresponding actuator, giving sufficient thought about its collaborators.

More details about communication access research can been seen in [29,36]. Ideally (without considering congestion, lost packets, electromagnetic interference, etc.), if there is no evidence of attack in this system, all data remain intact, which means there are $y_{sr1} = y_{rd1} = y^{(*)}_{d1/2}$, $u^{(*)}_{dr1} = u^{(*)}_{ra1}$, $y_{sr2} = y_{rd2} = y^{(*)}_{d2/1}$, $u^{(*)}_{dr2} = u^{(*)}_{ra2}$, and $e_{sr} = e_{rd} = e^{(*)}$.

3. Signal Attack in SIS

Due to the structure analysis above, two types of attacks are proposed and researched in this paper, which are Signal Attack $SA$and Mode Attack MA. Ultimately, the core objective of all the attacks intruded in a SIS is to cause a risk. For an SA, it has the ability to modify a regular signal into a dangerous one, while MA can create a risky task. In this paper, our attention is mainly focused on SA.

3.1. Signal Attack Form

It should be mentioned that there are some differences between the signal attack and jamming attack. The signal attack can modify the content of the data flow, but keep the format and reachability. The data flow would be blocked by the jamming attack.

Equation (4) is a typical form of SA, which has the ability to denote almost all of inside attacks existing in SCADA except the Denial of Service (DoS) attack.

$$q(k) = f_{Ni-p}(P[k,n], q(k-1), \Delta_p) \tag{4}$$

where $P[k,n]$ is a set of input data $p(k)$, $P[k,n] = [p(k), p(k-1), \cdots, p(k-n)]$, $q(k)$ is the output of $p(k)$ modified by the attack at step k, and Δ_p is a threshold of max(min)variation between $p(k_i - 1)$ and $p(k_i)$. The subscript Ni is the label of the device (including DCUs and RTUs), which is planned to handle $p(k)$.

It should be mentioned briefly that the DoS attack on a controlled closed-loop in SCADA is an attempt to make the network resource unavailable, against its requirements of reachability and observability. Based on the research results, we proposed in [38,39] that if a DCU has sent $w_{\rho j}$-data in one data flow by $DCUi$ to keep the closed-loop system Q_j l_j-step observable, Q_j is considered to be attacked on $DCUi$, if for each $M_{\omega 1}(k_m)$ where $m \in [0, l_j - 1]$, we have $|M_{\omega 1}(k_m)| \geq w_{\rho j} + 1$. That means the DoS attack has the ability to modify the DCU's predefined sequences. A DoS attack on $DCUi$ can be detected by a related $DCUj$ or CCI, intuitively, if they fail to receive scheduled and planned data from $DCUi$.

3.1.1. An Example of the Signal Attack Algorithm

Based on the definition of the signal attack, an example of the insertion attack algorithm is given in Algorithm 1, which operates as follows.

It takes as input original input data $p(k)$. Line 1 of the algorithm denotes each element in set $P[k,n]$ as $[p_0, p_1, \cdots, p_n]$. After that, a set $P_\Delta = [p_{\Delta 0}, p_{\Delta 1}, \cdots, p_{\Delta n}]$ is initialized in Line 2 to store the variation between each p_i and p_{i+1} (Lines 4–6). Basic modified data are stated by choosing data in $P[k,n]$ (Lines 7–8) randomly. Here, $RNG :: uniform(a,b)$ is a typical way to select a uniformly-distributed random number, which is from the range $[a,b)$ by using the MWCalgorithm. In addition, from Lines

9–16, a further modification is executed to confuse the IDSaimed at replay attack. The addition value denoted as r is based on the maximum or minimum of P_Δ, which is determined by the numerical relationship between $p(k)$ and $p(k-1)$. Lines 11 and 14 can keep r from the IDS based on data. What is more, the detection by the threshold would also be invalid by the restraint in Line 18. The output of the insertion attack algorithm is given in Line 23.

Algorithm 1 Signal attack algorithm.

Require: Original input data $p(k)$
1: remark $P[k,n] = [p_0, p_1, \cdots, p_n]$;
2: initialize $P_\Delta = [p_{\Delta 0}, p_{\Delta 1}, \cdots, p_{\Delta n}]$;
3: $p_0 = p(k)$;
4: **for** $i = 0$ to n **do**
5: $\quad p_{\Delta i} = p_i - p_{i+1}$;
6: **end for**
7: $j = int\ RNG :: uniform(0, n)$;
8: $\tilde{p} = p_j$;
9: **if** $p(k) > p(k-1)$ **then**
10: $\quad \Delta_p = maxP_\Delta$;
11: $\quad r = float\ RNG :: uniform(0, \Delta_p)$;
12: **else**
13: $\quad \Delta_p = minP_\Delta$;
14: $\quad r = float\ RNG :: uniform(\Delta_p, 0)$;
15: **end if**;
16: $\hat{p}(k) = \tilde{p} + r$;
17: $\Delta_p = abs(\Delta_p)$;
18: $q(k) = min(q(k-1) + \Delta_p, max(\hat{p}(k), q(k-1) - \Delta_p))$;
19: **for** $i = n$ to 1 **do**
20: $\quad q(n) = q(n-1)$;
21: **end for**
22: $q(k-1) = q(k)$;
23: **return** $q(k)$;

3.1.2. The Form of the Hazard Factor-Based Signal Attack

According to the form of the basic signal attack, in order to find the balance between the hazard and invisibility requirements of signal attack, a new type of signal attack is presented in this paper. This has an additional parameter named hazard factor (denoted as η_i). For the given original data signal $p_i(k)$ and the corresponding typical signal attack $q_i(k)$, we have:

$$q_{i-HAZ}(k) = p_i(k) + \eta_i[q_i(k) - p_i(k)] \tag{5}$$

where $q_{i-HAZ}(k)$ is the output of the hazard factor-based signal attack. According to the different values of η_i, $q_{i-HAZ}(k)$ can be further classified as:

$q_{i-HAZ}(k) = p_i(k)$; if we have $\eta_i = 0$, the signal attack does not exist;
$q_{i-HAZ}(k)$ is the lower hazard, if $\eta_i \in (0, 1)$;
$q_{i-HAZ}(k)$ is equivalent to $q_i(k)$, if$\eta_i = 1$;
$q_{i-HAZ}(k)$ would be the higher hazard, if $\eta_i \in (1, \eta_{i-\max})$, and here, $\eta_{i-\max}$ is the maximum allowed hazard factor of $q_{i-HAZ}(k)$, which can be hidden from the IDS.

3.1.3. Signal Attack Zone

As shown in Figure 2, according to the refined model of SIS, there exist several point-to-point communication lines that make up a closed-loop system. The insertion attack may happen in any node between two lines. Here, we defined each probable attack zone in a cooperative SCADA, which are listed as follows.

Attack on Local Sensor Data

A local sensor is used to sample the running status of one actuator, which is activated by the cooperative control mission. Normally, for local sensor data (Sensor 1 as an example), without being attacked, we have $y_{sr1} = y_{rd1} = y^{(*)}_{d1/2}$. Such data may be attacked on RTU1, which leads to $y_{rd1} \neq y_{sr1}$ or $y^{(*)}_{d1/2} \neq y_{rd1}$ if DCU1-1(2) is attacked.

Attack on Global Sensor Data

As shown in Figure 2, for the global Sensor 3, in an ideal case, we have $e_{sr} = e_{rd} = e^{(*)}$. An insertion attack may tamper with the data as a result of $e_{rd} \neq e_{sr}$ or $e^{(*)} \neq e_{rd}$ if the attack is embedded in RTU3 or DCU3-1(2).

Attack on Actuator Control Data

Finally, an actuator also can be attacked in SCADA by causing an undetected misoperation $u^{(*)}_{ra1} \neq u^{(*)}_{dr1}$ in RTU2. Meanwhile, if the calculational result of the control algorithm in DCU1-1(1) is attacked, $u^{(*)}_{dr1}$ would be an incorrect output, which means $u^{(*)}_{dr1} \neq \hat{u}^{(*)}$, where $\hat{u}^{(*)}$ is denoted as the calculational result.

4. Critical State Analysis

4.1. Critical State Estimation

In this paper, the Critical State Estimation (CSE) algorithm we propose is based on the Industrial State Modeling Language (ISML), which was first proposed by A. Carcano et al. in [28]. The rules in the ISML are formulized as $condition \rightarrow action$ where $condition$ is a Boolean formula composed of several predicates, which are used to indicate the values that are assumed by critical components. The definition of ISML is listed in the following.

$$
\begin{aligned}
\langle rule \rangle &::= \langle condition \rangle \rightarrow \langle action \rangle : \langle l_t \rangle \\
\langle action \rangle &::= \langle Alert \rangle | Log \qquad \langle l_t \rangle ::= 1 | \ldots | 5 \\
\langle condition \rangle &::= \langle predicate \rangle | \langle predicate \rangle, \langle condition \rangle \\
\langle object_{bin} \rangle &::= DCU\langle ID \rangle.\langle bincomp \rangle.\langle index \rangle \\
\langle object \rangle &::= DCU\langle ID \rangle.\langle comp \rangle.\langle index \rangle \\
\langle predicate \rangle &::= \langle object \rangle.\langle rel \rangle \langle val \rangle \\
&\quad | \langle object_{bin} \rangle.\langle binrel \rangle \langle binval \rangle \\
\langle ID \rangle &::= IPaddress : Port \\
\langle val \rangle &::= 0 | \ldots | 2^{16} - 1 \qquad \langle comp \rangle ::= HR | IR \\
\langle index \rangle &::= 0 | \ldots | 2^{16} - 1 \qquad \langle binrel \rangle ::= = | \neq \\
\langle rel \rangle &::= \leq | \geq | < | > | = | \neq \\
\langle bincomp \rangle &::= CO | DI \qquad \langle binval \rangle ::= 0 | 1
\end{aligned}
$$

where $\langle comp \rangle$ is a register; Discrete Input, Coli, Input Register, and Holding Register are denoted as DI, CO, IR, HR, respectively. The ISML is used to describe a particular class of system states called critical states that correspond to dangerous or unwanted situations in SIS. Here, the risk level of each state is reversed by its confidence. The risk level l_t is considered to be in the critical state, where a value of one means low risk, while five is a surely dangerous critical state of SIS. Here, $object | object_{bin}$ denotes one kind of data in SIS, and $\langle object \rangle : \langle condition \rangle \rightarrow \langle Alert \rangle : \langle l_t \rangle ::= \langle rel \rangle \langle val \rangle$ means the critical state value ($\langle rel \rangle \langle val \rangle$) of $object$ when the risk level reaches l_t by $\langle condition \rangle$, for example if such a rule is set in SIS:

$$
\begin{pmatrix} DCU[10.0.0.001 : 502].HR[1] > 3000 \\ DCU[10.0.0.002 : 502].IR[2] > 2500 \end{pmatrix} \rightarrow Alert : 5 \tag{6}
$$

We have $\langle DCU[10.0.0.002:502].IR[2]\rangle : \langle DCU[10.0.0.001:502].HR[1] > 3000\rangle \rightarrow \langle Alert:5\rangle ::= \langle > 2500\rangle$, which means for $DCU[10.0.0.002:502].IR[2]$, the critical state value is >2500 (with risk of $l_t = 5$), when its related data $DCU[10.0.0.001:502].HR[1] > 3000$.

Therefore, the anomaly detection and critical state estimation algorithm is given in Algorithm 2, which operates as follows.

Algorithm 2 Critical state estimation algorithm.

Require: $y_i(k)$, p_i (RT level of $y_i(k)$), $y_i(k)$-related rule set $R[y_i]$, $R[y_i]$-related dataset $\mathsf{y}_i(\mathsf{k})$

1: reorder and remark $R[y_i] = R_1[1], R_1[2], \cdots, R_1[m_1], R_2[1], R_2[2], \cdots, R_2[m_2], \cdots, R_5[m_5]$;
2: **for** $p = p_i$ to 5 **do**
3: **for** $q = 1$ to m_p **do**
4: Initialize interval $I_p = [y_{i-min}, y_{i-max}]$
5: remark the $R_p[q]$-related subset of $\mathsf{y}_i(\mathsf{k})$ as $y_{i-pq}(k)$;
6: Set subinterval $I_{pq} = \mathsf{Q} - \langle rel_{pq}\rangle\langle val_{pq}\rangle = \mathsf{Q} - \langle y_i\rangle : \langle y_{i-pq}(k)\rangle \rightarrow \langle Alert\rangle : \langle p\rangle$
7: $y^{sup}_{i-pq}(k) = sup\{I_{pq}\}$
8: $y^{inf}_{i-pq}(k) = inf\{I_{pq}\}$
9: Set interval $I_p = I_p \cap I_{pq}$
10: $y^{sup}_{i-p}(k) = sup\{I_p\}$
11: $y^{inf}_{i-p}(k) = inf\{I_p\}$
12: **if** $y^{sup}_{i-pq}(k) = y^{sup}_{i-p}(k)$ **then**
13: $y^{ssup}_{i-p}(k) = y_{i-pq}(k)$;
14: **end if**;
15: **if** $y^{inf}_{i-pq}(k) = y^{inf}_{i-p}(k)$ **then**
16: $y^{sinf}_{i-p}(k) = y_{i-pq}(k)$;
17: **end if**;
18: **if** $y^{sup}_{i-pq}(k) = y_{i-max}$ **then**
19: $y^{ssup}_{i-p}(k) = null$;
20: **end if**;
21: **if** $y^{inf}_{i-pq}(k) = y_{i-min}$ **then**
22: $y^{sinf}_{i-p}(k) = null$;
23: **end if**;
24: **end for**
25: **if** $I_p = \emptyset$ or $y_i(k) \notin I_p$ **then**
26: $y_i(k)$ is beyond p-level risk;
27: **else**
28: $y_i(k)$ is p-level non-risk;
29: **end if**;
30: **end for**
31: **return** $\mathsf{y}_{\mathsf{i-sd}}(\mathsf{k}) = [y^{inf}_{i-5}(k), y^{sup}_{i-5}(k), y^{inf}_{i-4}(k), y^{sup}_{i-4}(k), \cdots, y^{inf}_{i-p_i}(k), y^{sup}_{i-p_i}(k)]$ and $\mathsf{y}_{\mathsf{i-ps}}(\mathsf{k}) = [y^{ssup}_{i-p_i}(k), y^{sinf}_{i-p_i}(k)]$

It takes as input original input data $y_i(k)$, where the physical meaning of y_i is determined by its *object*; meanwhile, the $y_i(k)$-related rule set $R[y_i]$ is needed, as well. According to $R[y_i]$, all related *objects* are necessary and stored in dataset $\mathsf{y}_i(\mathsf{k})$. It should be mentioned that $\mathsf{y}_i(\mathsf{k})$ is not equivalent to the set of all $y_i(k)$ coupling data. Line 1 of the algorithm restores each rule of $R[y_i]$ by its risk level and denotes these rules as $R_p[q]$ where p is the risk level. Lines 2–30 show the critical state estimation method, for each rule $R_p[q]$ and its related dataset $\mathsf{y}_{i-pq}(\mathsf{k})$; a critical state of y_i is determined. Line 6 creates a safe subinterval for y_i under rule $R_p[q]$, and the interval is shrunk during each loop computation in Line 9. Lines 10–11 show the upper and lower bound of I_p, denoted as $y^{sup}_{i-p}(k)$ and $y^{inf}_{i-p}(k)$, respectively. Lines 12–23 show the way to find the determinant factors (denoted as y^{ssup}_{i-p} and y^{sinf}_{i-p}), which leads to $y_i(k)$ being risk data or not. Lines 25–29 shows the anomaly detection method, if $y_i(k) \in I_p$, $y_i(k)$ is the p level risk of non-arrival or it is called beyond the p level risk. Due to the different importance of each $y_i(k)$, its Risk Tolerance $RT$is different. For four-level RT data $y_i(k)$, if the judgment result is beyond four levels of risk, $y_i(k)$ is anomalous data. Line 31 returns all upper and lower bounds of each risk level for $y_i(k)$, which is $y_{i-sd}(k) = [y^{inf}_{i-5}(k), y^{inf}_{i-4}(k), y^{inf}_{i-3}(k), y^{inf}_{i-2}(k), y^{inf}_{i-1}(k),$

$y_{i-1}^{sup}(k), y_{i-2}^{sup}(k), y_{i-3}^{sup}(k), y_{i-4}^{sup}(k), y_{i-5}^{sup}(k)]$. Meanwhile, the the determinant factors (denoted as y_{i-p}^{ssup} and y_{i-p}^{sinf}) are uploaded, as well.

4.2. Bi-Critical Data Analysis

According to Algorithm 2, such data, the value of which is beyond the critical state, are determined; however, there exists the possibility that one normal datum is miscalculated, caused by a related datum, which is actually anomalous. Here, we propose the definition of bi-critical data to identify the true abnormal data from two related data.

Definition 1 (Bi-critical Data *BD*): Two critical data $p_a - RT$, data $y_a(k)$, and $p_b - RT$, data $y_b(k)$, in SIS are regarded as a pair of *BD*, if $y_b(k) \in \mathsf{y_{a-p_a s}(k)}$ or $y_a(k) \in \mathsf{y_{b-p_b s}(k)}$.

According to Definition 1, a further analysis of critical state discrimination is proposed in Algorithm 3. Here, we assume that $y_b(k) \in \mathsf{y_{a-p_a s}(k)}$.

Algorithm 3 Critical data discrimination algorithm for a pair of *BD*.

Require: $y_a(k)$, $y_b(k)$, $y_b(k-1)$, p_a, $y_a(k)$-related rule set $R[y_a]$, $R[y_a]$-related dataset $\mathsf{y}_a(\mathsf{k})$
1 initialize a $R[y_a]$-related dataset $\mathsf{y}_{a/b}(\mathsf{k})$, which includes every type of data belong to $\mathsf{y}_a(\mathsf{k})$, except $y_b(k)$
2 initialize a $y_a(k)$-related, but $y_b(k)$ non-related rule set $R[y_{a/b}]$
3 choose $y_a(k)$, p_a, $R[y_{a/b}]$, $\mathsf{y}_{a/b}(\mathsf{k})$ as inputs, and run Algorithm 2
4 **if** the result of Algorithm 2 shows that $y_a(k)$ is beyond p_a-level risk **then**
5 **return** $y_a(k)$ is a definitely beyond p_a-level risk (*DRD*)
6 **else**
7 reset $y_b(k) = y_b(k-1)$
8 choose $y_a(k)$, p_a, $R[y_a]$, $\mathsf{y}_b(\mathsf{k})$ as inputs, and rerun Algorithm 2
9 **if** the result of Algorithm 2 shows that $y_a(k)$ is beyond p_a-level risk data **then**
10 **return** $y_a(k)$ is definitely beyond p_a-level risk data (*DRD*)
11 **else**
12 **return** $y_a(k)$ is potentially beyond p_a-level risk data (*PRD*)
13 **end if**
14 **end if**

Here, in Algorithm 3, we propose a two-layer discrimination mode. It takes as input original input data $y_a(k)$, $y_b(k)$, p_a, $y_a(k)$-related rule set $R[y_a]$, and $R[y_a]$-related dataset $\mathsf{y}_a(\mathsf{k})$. In Lines 2–3, two subsets of $\mathsf{y}_a(\mathsf{k})$ and $R[y_a]$, which exclude the factor of $y_b(k)$, are denoted as $\mathsf{y}_{a/b}(\mathsf{k})$ and $R[y_{a/b}]$, respectively. In Line 4, Algorithm 2 is called to calculate the critical data situation of $y_a(k)$ without considering the existence of $y_b(k)$. If the result shows that $y_a(k)$ is still a beyond p_a-level risk, that means $y_a(k)$ is Definitely Risk Data (*DRD*) whatever the value of $y_b(k)$. If $y_a(k)$ is a p_a-level non-risk by the analysis based on Algorithm 2, further discrimination is presented in Lines 7–13. As the data sampling mode of SIS is based on the zero-order holder (ZOH), here we treat $y_b(k)$ as unadopted data and reset y_b's value at step k as $y_b(k-1)$. Then, Algorithm 2 is called again at Line 8. If $y_a(k)$ is still a p_a-level risk, it means $y_a(k)$ is a *DRD*, while y_b is a core determinant of $y_a(k)$'s riskiness. If $y_a(k)$ is not a p_a-level risk according to the modified result of $y_b(k)$, such a possibility should exist that $y_a(k)$ is non-risk data, but it should be marked in a miscalculation caused by the risk data $y_b(k)$. Due to the nondeterminacy of risk level, this kind of $y_a(k)$ is named as Potentially Beyond p_a-level risk Data (*PRD*). In this paper, due to the uncertainty of the risk level, a *DRD* issue in SIS would be considered as a higher priority than a *PRD*.

5. Simulation

5.1. Modeling of the Ship Cooperative Motion Control System

Due to the complexity of ship motion, it has six Degrees Of Freedom (DOF) as a general rule, which can be described as u (surge velocity), v (sway velocity), w (heave velocity), r (yaw rate), p

(rolling rate), and q (pitching rate). In this paper, we mainly focus on three motions: ship surging, ship heading, and ship rolling, and the ship motion model we chose in this paper is in Equation (7).

$$\begin{cases} X_{\sum} = m[\dot{u} - vr + wq - x_G(q^2 + r^2) + y_G(pq - \dot{r}) + z_G(pr + \dot{q})] \\ N_{\sum} = J_{zx}\dot{p} + J_{yz}\dot{q} + J_z\dot{r} + (J_{xy}p + J_yq + J_{yz}r)p - (J_xp + J_{xy}q \\ \quad + J_{zx}r)q + m[x_G(\dot{v} + ur - wp) + y_G(-\dot{u} - vp + uq)] \\ K_{\sum} = J_x\dot{p} + J_{xy}\dot{q} + J_{zz}\dot{r} + (J_{zx}p + J_{zy}q + J_zr)q - (J_{xy}p + J_yq \\ \quad + J_{yz}r)r + m[y_G(\dot{w} + vp - uq) + z_G(-\dot{v} - ur + wp)] \end{cases} \tag{7}$$

where m is the mass of the ship and $\dot{p}$, $\dot{q}$, $\dot{r}$ are respectively denoted as the rolling, pitching, and yawing angular acceleration. $R_G = (x_G\, y_G\, z_G)^T$ is the coordinates of the position vector about the center of the ship's gravity in the moving coordinate system. $X_{\sum}$, $N_{\sum}$, and $K_{\sum}$ are respectively denoted as the longitudinal force, heading resultant moment, and rolling resultant moment. J is the inertia matrix of the ship, when the origin of the coordinate system is not the center of the ship's gravity, as Equation (8) shows.

$$J = \begin{bmatrix} J_x & J_{xy} & J_{zx} \\ J_{yx} & J_y & J_{yz} \\ J_{zx} & J_{zy} & J_z \end{bmatrix} \tag{8}$$

As the system is constituted by two rudders, two propellers, and a pair of fins, the compositions of $X_{\sum}$,$N_{\sum}$, and $K_{\sum}$ are shown in Equation (9):

$$\begin{cases} X_{\sum} = X_I + X_H + X_{RP} + X_{LP} + X_{RR} + X_{LR} + X_F + X_D \\ N_{\sum} = N_I + N_H + N_{RP} + N_{LP} + N_{RR} + N_{LR} + N_F + N_D \\ K_{\sum} = K_I + K_H + K_{RP} + K_{LP} + K_{RR} + K_{LR} + K_F + K_D \end{cases} \tag{9}$$

where $I, H, RP, LP, RR, LR, RF, F, D$ are respectively denoted as fluid inertia, fluid viscosity, right propeller, left propeller, right rudder, left rudder, fins, and disturbances. As is shown, every plant working in the system has the ability to change the ship's surging, heading, and rolling more or less, which depends on the moment it produced in different DOF. This behavior increases the importance of cooperative control algorithms, which means we also need a real-time communication environment.

Of many possible external disturbances acting on the ship motion process, the waves are the most important external disturbances and dominantly influence the control performance. As the wave disturbance can be treated as a typical stationary random process satisfying a Gaussian distribution, the spectrum of the random ocean wave is given in Equation (10):

$$S_{\tilde{\zeta}}(\omega_e) = \frac{S_{\tilde{\zeta}}(\omega)}{1 - 2\omega/gV\cos\mu} \tag{10}$$

where P-Mspectrum $S_{\tilde{\zeta}}(\omega)$ is chosen as the initial spectrum, V is the ship speed, and μ is the wave angle.

Therefore, the interfering moment of wave disturbance N_{wave} can be determined as:

$$\begin{aligned} N_{wave} = \sum_{i=1}^{M} R_1[B_m^2 \sin R_2(R_3\cos R_3 - \sin R_3)/R_3^2 - \\ L^2 \sin R_3(R_2\cos R_2 - \sin R_2)/R_2^2]\times \\ \zeta_{ai}\cos(\omega_{ei}t + \varepsilon_{ni}) \end{aligned} \tag{11}$$

where $R_1 = \rho g(1 - e^{-k_1 d_m})\Big/k_1$, $R_2 = (k_1 L/2)\cos\mu_e$, $R_3 = (k_1 B_m/2)\sin\mu_e$, ζ_{ai} is the amplitude of each harmonic, M is the number of energy partitions, B_m is the beam, L is the length, and d_m is the average draft.

In this paper, the ship chosen in the simulation has a displacement of 2500 tons, and we have $B_m = 14$, $L_m = 115$, $d_m = 3.8$.

According to the wave disturbance model above, in such conditions in which significant wave height is four meters and the wave angle is 30°, the force and moment of sea wave disturbance are shown in Figure 3.

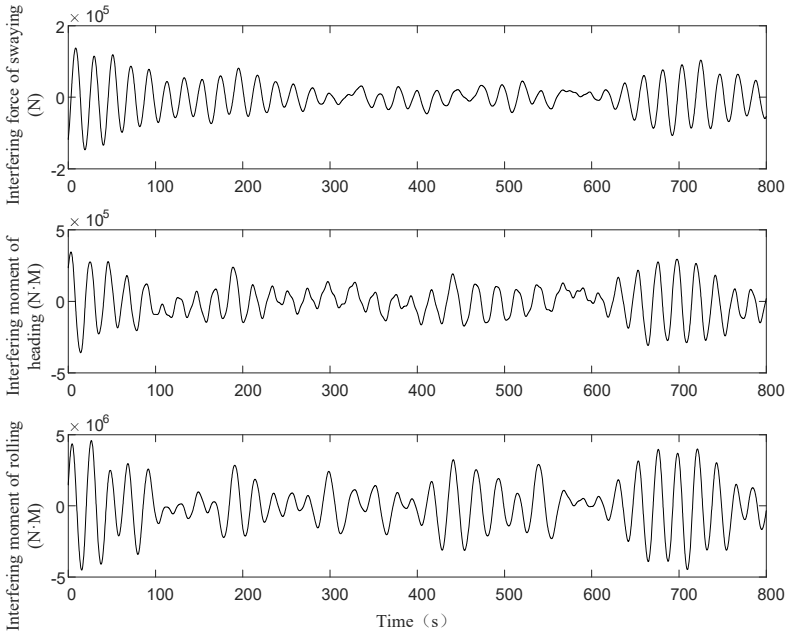

Figure 3. Force and moment of sea wave disturbance at 30°.

In addition, Figure 4 shows the heading and rolling angles of the ship without any control commands under wave disturbance. The test platform in this paper is based on a semi-physical simulation platform, which was introduced in [36].

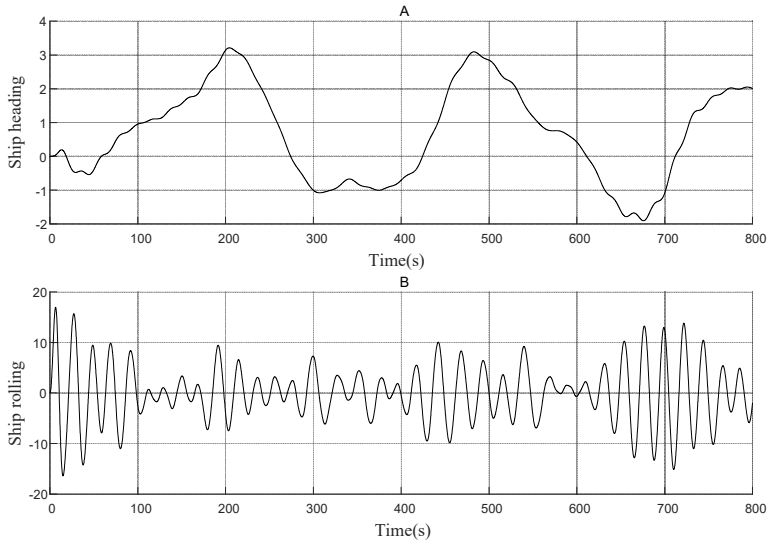

Figure 4. Heading (**A**) and rolling (**B**) angles of the ship without control commands.

5.2. Influence of Signal Attack in SCMCS

Under the simulation results of ship dynamics, in this subsection, the influence of signal attack in the Ship Cooperative Motion Control System (SCMCS) is researched and analyzed. In order to

establish the closed loop, the distributed collaborative control algorithm of ship heading and rolling we adopt here is based on a PID-fuzzy fusion control law, which was proposed in [40]. This fuzzy PID fusion controller was designed through continuous updating of its output scaling factor. Instead of using a unitary fuzzy or PID algorithm, the fusion weighted summation rule bases are used in parallel, which improved the performance of the proposed fuzzy PID controllers compared to others. The fusion FPID parameter is calculated as:

$$u = \sum_{i=1}^{n} \alpha_i \times u_i \tag{12}$$

where u_i is the output by each control algorithm (the fuzzy controller and PID controller are treated as subprograms of the fusion algorithm), α_i is the fusion factor of each subprograms, and n is the number of total subprograms in a fusion algorithm; normally, there are $n = 2$. In this paper, the fusion factor is chosen as:

$$\alpha_i = \left[1 - exp\left(-\left|u_i\right| \Big/ \sum_{i=1}^{n} |u_i|\right)\right] \times \frac{1}{n \times (1 - exp(-1/n)} \tag{13}$$

Due to the space limitation, the working principle and application effect of this algorithm will not be introduced in this paper. The simulation results of ship heading and rolling based on this control algorithm are shown by dotted lines in Figure 5A,B, respectively. Meanwhile the solid line in Figure 5A,B depicts the heading and rolling output of the ship while the signal attack acted on the heading data signal. Here, the signal attack first happened at 80 seconds, and we have $\eta_i = 0.5$. In addition, the operation states of main (flap) rudder and main (flap) fin are shown by solid (dotted) lines in Figure 5C,D, respectively. Due to the coupling relationship between ship heading and rolling, the manipulation of heading sensor data can also change the effect of the rolling control system, and the mathematical statistics results are listed in Table 2.

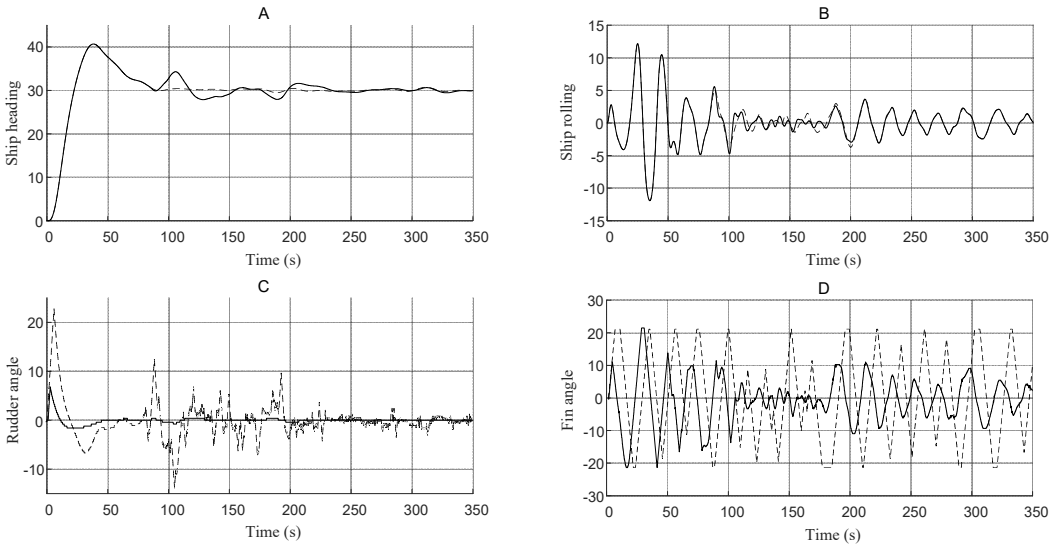

Figure 5. Cooperative control effects of ship heading and rolling under signal attack. The simulation results of ship heading and rolling based on this control algorithm are shown by dotted lines in Figure A,B, respectively. Meanwhile the solid line in (**A**,**B**) depicts the heading and rolling output of the ship while the signal attack acted on the heading data signal. Here, the signal attack first happened at 80 s, and we have hi = 0.5. In addition, the operation states of main (flap) rudder and main (flap) fin are shown by solid (dotted) lines in (**C**,**D**), respectively.

Table 2. Influence of rolling mission by heading signal attack.

	Non-Attack		With-Attack	
	Mean	**Variance**	**Mean**	**Variance**
Ship rolling	-0.019°	7.95	0.040°	10.01
Fin angle	-0.165°	45.71	-0.181°	54.37
Flap fin angle	0.771°	164.05	1.12°	184.35

5.3. Anomaly Detection Analysis of SCMCS

In this paper, based on the running effect in the semi-physical simulation platform, which was introduced in [36], the risk data rule is set as follows:

$Rule\ 1 : \langle |DCU[10.0.0.001:502].IR[2]| \rangle : \langle |DCU[10.0.0.004:502].HR[1]| < 8 \rangle \rightarrow \langle Alert:5 \rangle ::= \langle > 10 \rangle$

$Rule\ 2 : \langle |DCU[10.0.0.001:502].IR[2]| \rangle : \langle |DCU[10.0.0.004:502].HR[1]| < 8 \rangle \rightarrow \langle Alert:4 \rangle ::= \langle > 8 \rangle$

$Rule\ 3 : \langle |DCU[10.0.0.001:502].IR[2]| \rangle : \langle |DCU[10.0.0.004:502].HR[1]| < 6 \rangle \rightarrow \langle Alert:3 \rangle ::= \langle > 6.5 \rangle$

$Rule\ 4 : \langle |DCU[10.0.0.001:502].IR[2]| \rangle : \langle |DCU[10.0.0.004:502].HR[1]| < 6 \rangle \rightarrow \langle Alert:2 \rangle ::= \langle > 4 \rangle$

$Rule\ 5 : \langle |DCU[10.0.0.001:502].IR[1]| \rangle : \langle |DCU[10.0.0.004:502].HR[1]| < 8 \rangle \rightarrow \langle Alert:5 \rangle ::= \langle > 2 \rangle$

$Rule\ 6 : \langle |DCU[10.0.0.001:502].IR[1]| \rangle : \langle |DCU[10.0.0.004:502].HR[1]| < 6 \rangle \rightarrow \langle Alert:4 \rangle ::= \langle > 1 \rangle$

$Rule\ 7 : \langle |DCU[10.0.0.001:502].IR[1]| \rangle : \langle |E_0 - DCU[10.0.0.003:502].HR[1]| < 8 \rangle \rightarrow \langle Alert:3 \rangle ::= \langle > 2 \rangle$

$Rule\ 8 : \langle |DCU[10.0.0.001:502].IR[1]| \rangle : \langle |E_0 - DCU[10.0.0.003:502].HR[1]| < 5 \rangle \rightarrow \langle Alert:4 \rangle ::= \langle > 2 \rangle$

$Rule\ 9 : \langle |DCU[10.0.0.001:502].IR[2]| \rangle : \langle |E_0 - DCU[10.0.0.003:502].HR[1]| < 5 \rangle \rightarrow \langle Alert:4 \rangle ::= \langle > 15 \rangle$

$Rule\ 10 : \langle |DCU[10.0.0.001:502].IR[2]| \rangle : \langle |E_0 - DCU[10.0.0.003:502].HR[1]| < 8 \rangle \rightarrow \langle Alert:5 \rangle ::= \langle > 15 \rangle$

The notations of each rule are listed in Table 3.

Table 3. Notations of rules.

Annotation	**Notations**
$DCU[10.0.0.001:502]$	DCU for ship rudders
$DCU[10.0.0.002:502]$	DCU for ship fins
$DCU[10.0.0.003:502]$	DCU for heading sensor
$DCU[10.0.0.004:502]$	DCU for rolling sensor
$DCU[10.0.0.001:502]IR[1]$	Input register for rudder command
$DCU[10.0.0.001:502]IR[2]$	Input register for flap rudder command
$DCU[10.0.0.002:502]IR[1]$	Input register for fin command
$DCU[10.0.0.002:502]IR[2]$	Input register for flap fin command
$DCU[10.0.0.003:502]HR[1]$	Holding register for heading sensor
$DCU[10.0.0.004:502]HR[1]$	Holding register for rolling sensor
E_0	Set value of ship heading

Here, we assume that the Risk Tolerance of SCMCS in SIS is 4, which means only Rule 1, 2, 5, 6, 8, 9 and 10 need to be taken into account. And these rules are used to limit the data in $DCU[10.0.0.001:502]IR[1]$ and $DCU[10.0.0.001:502]IR[2]$. As shown in in Figure 6, according to Algorithm 3, the abnormal data of ship rudder and flap rudder are first detected at 81.7 s and 80.4 s, respectively.

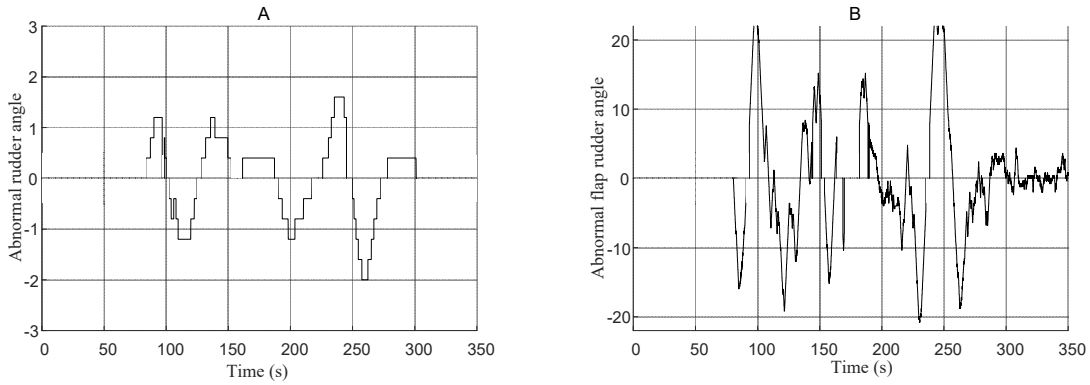

Figure 6. Data abnormalities of rudder and flap rudder. In (**A**,**B**), the abnormal data of ship rudder and flap rudder are first detected at 81.7 s and 80.4 s by Algorithm respectively.

6. Discussions and Conclusions

In this paper, the basic structure of the ship information system and its typical cooperative control mode were formulated. According to such structure, a signal attack detection method was proposed. Under the consideration of coupling data flow, we improved the Critical State Estimation (CSE) algorithm proposed in [28] by setting new sentence patterns of the Industrial State Modeling Language (ISML). Therefore, such risk data can be determined by the related data and a set of predefined rules. The simulation result shows that wherever the data are attacked by the signal attack in the cooperative control loop, this can always be detected. We have to point out that we did not focus on the prevention of signal attack in the paper. For now, waking up a related spare DCU is the typical reconstitution strategy when the signal attack is detected. More intelligent solutions can be researched in the future.

Author Contributions: B.X. conceived of and designed the experiments; S.C. performed the experiments; Y.J. and Y.L. analyzed the data; S.C. contributed analysis tools; B.X. wrote the paper.

Funding: This paper is sponsored by the Shanghai Sailing Program (No. 18YF1409900) and the Shanghai Innovation Action Plan (No. 17050502000).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Mo, Y.; Kim, H.J.; Brancik, K.; Dickinson, D.; Lee, H.; Perrig, A.; Sinopoli, B. Cyber–Physical Security of a Smart Grid Infrastructure. *Proc. IEEE* **2011**, *100*, 195–209.
2. Slay, J.; Miller, M. Lessons Learned from the Maroochy Water Breach. *Int. Fed. Inf. Process.* **2007**, *253*, 73–82.
3. Abrams, M.D. Malicious Control System Cyber Security Attack Case Study–Maroochy Water Services, Australia. In Proceedings of the Annual Computer Security Applications Conference, Anaheim, CA, USA, 8–12 December 2008; Volume 253, pp. 73–82.
4. Nicholson, A.; Webber, S.; Dyer, S.; Patel, T.; Janicke, H. SCADA security in the light of Cyber-Warfare. *Comput. Secur.* **2012**, *31*, 418–436. [CrossRef]
5. Langner, R. Stuxnet: Dissecting a Cyberwarfare Weapon. *IEEE Secur. Priv.* **2011**, *9*, 49–51. [CrossRef]
6. Ten, C.W.; Liu, C.C.; Manimaran, G. Vulnerability Assessment of Cyber Security for SCADA Systems. *IEEE Trans. Power Syst.* **2008**, *23*, 1836–1846. [CrossRef]
7. Knijff, R.M.V.D. Control Systems/SCADA Forensics, What's the Difference? *Digit. Investig.* **2014**, *11*, 160–174. [CrossRef]
8. Nate Kube.Cyberphysical Security: The Next Frontier. Available online: http://www.securityweek.com/cyberphysical-security-next-frontier (accessed on 23 March 2015).
9. Pollet, J. Developing a solid SCADA security strategy. In Proceedings of the 2nd ISA/IEEE Sensors for Industry Conference, Houston, TX, USA, 19–21 November 2002; pp. 148–156.
10. Ten, C.W.; Manimaran, G.; Liu, C.C. Cybersecurity for Critical Infrastructures: Attack and Defense Modeling. *IEEE Trans. Syst. Man. Cybern. Part A. Syst. Hum.* **2010**, *40*, 853–865. [CrossRef]

11. Barbosa, R.R.R.; Pras, A. Intrusion Detection in SCADA Networks. In *Mechanisms for Autonomous Management of Networks and Services*; Springer: Berlin/Heidelberg, Germany, 2010; pp. 163–166.
12. Cardenas, A.; Amin, S.; Sastry, S. Attacks against process control systems: Risk assessment, detection, and response. In Proceedings of the ACM Symposium on Information, Computer and Communications Security, Hong Kong, China, 22–24 March 2011; pp. 355–366.
13. Cárdenas, A.A.; Amin, S.; Sinopoli, B.; Giani, A.; Perrig, A.; Sastry, S. Challenges for Securing Cyber Physical Systems. In Proceedings of the First Workshop on Cyber-physical Systems Security, Stockholm, Sweden, 12–16 April 2010; pp. 363–369.
14. Wilson, D.C.; Pala, O.; Tolone, W.J. Recommendation-based geovisualization support for reconstitution in critical infrastructure protection. *Proc. SPIE* **2009**, *7346*. [CrossRef]
15. Zhou, C.; Huang, S.; Xiong, N.; Yang, S.H. Design and Analysis of Multimodel-Based Anomaly Intrusion Detection Systems in Industrial Process Automation. *IEEE Trans. Syst. Man Cybern. Syst.* **2015**, *45*, 1345–1360. [CrossRef]
16. Svendsen, N.; Wolthusen, S. Modeling and Detecting Anomalies in Scada Systems. *Int. Fed. Inf. Process.* **2008**, *290*, 101–113.
17. Ntalampiras, S. Detection of Integrity Attacks in Cyber-Physical Critical Infrastructures Using Ensemble Modeling. *IEEE Trans. Ind. Inform.* **2015**, *11*, 104–111. [CrossRef]
18. Goldenberg, N.; Wool, A. Accurate modeling of Modbus/TCP for intrusion detection in SCADA systems. *Int. J. Crit. Infrastruct. Prot.* **2013**, *6*, 63–75. [CrossRef]
19. Svendsen, N.; Wolthusen, S. Using Physical Models for Anomaly Detection in Control Systems. In *Critical Infrastructure Protection III*; Springer: Berlin/Heidelberg, Germany, 2009; pp. 139–149.
20. Kumarage, H.; Khalil, I.; Tari, Z.; Zomaya, A. Distributed anomaly detection for industrial wireless sensor networks based on fuzzy data modelling. *J. Parallel Distrib. Comput.* **2013**, *73*, 790–806. [CrossRef]
21. Hadžiosmanović, D.; Bolzoni, D.; Hartel, P.H. A log mining approach for process monitoring in SCADA. *Int. J. Inform. Secur.* **2012**, *11*, 231–251. [CrossRef]
22. Kang, D.H.; Kim, B.K.; Na, J.C.; Hang, K.S. Whitelists Based Multiple Filtering Techniques in SCADA Sensor Networks. *J. Appl. Math.* **2014**, *2014*, 1–7. [CrossRef]
23. Ochin, E.; Dobryakova, L.; Pietrzykowski, Z.; Borkowski, P. The application of cryptography and steganography in the integration of seaport security subsystems. *Sci. J. Marit. Univ. Szczec.* **2011**, *26*, 80–87.
24. Ochin, E. GPS/GNSS spoofing and the real-time single-antenna-based spoofing detection system. *Sci. J. Marit. Univ. Szczec.* **2017**, *52*, 145–153.
25. Kiss, I.; Genge, B.; Haller, P.; Sebestyen, G. Data clustering-based anomaly detection in industrial control systems. In Proceedings of the IEEE International Conference on Intelligent Computer Communication and Processing, Cluj-Napoca, Romania, 4–6 September 2014; pp. 275–281.
26. Stone, S.; Temple, M. Radio-frequency-based anomaly detection for programmable logic controllers in the critical infrastructure. *Int. J. Crit. Infrastruct. Prot.* **2012**, *5*, 66–73. [CrossRef]
27. Jung, S.M.; Song, J.-G.; Kim, T.-H.; So, Y.-H.; Kim, S.-S. Design of Idle-time Measurement System for Data Spoofing Detection. *J. Korea Acad.-Ind. Cooperation Soc.* **2010**, *11*, 151–158. [CrossRef]
28. Carcano, A.; Coletta, A.; Guglielmi, M.; Masera, M.; Fovino, I.N.; Trombetta, A.A. A Multidimensional Critical State Analysis for Detecting Intrusions in SCADA Systems. *IEEE Trans. Ind. Inform.* **2011**, *7*, 179–186. [CrossRef]
29. Liu, S.; Xing, B.; Li, B.; Gu, M.M. Ship information system: Overview and research trends. *Int. J. Naval Archit. Ocean Eng.* **2014**, *6*, 670–684. [CrossRef]
30. Liu, S.; Xing, B.; Li, B. Development actuality and key technology of networked control system. In Proceedings of the 32nd Chinese Control Conference, Xi'an, China, 26–28 July 2013; pp. 6692–6697.
31. Simoncic, R.; Weaver, A.C.; Cain, B.G.; Colvin, M.A. SHIPNET: A real-time local area network for ships. In Proceedings of the 1988 13th Conference on Local Computer Networks, Minneapolis, MN, USA, 10–12 October 1988; pp. 424–432.
32. Andersen, S.C.; Boyle, G.G.; Kubischata, M.D.; Marshik, J.V.; Robinson, R.P. Unisys SAFENET data transfer system (layers 1–4). In Proceedings of the 15th Conference on Local Computer Networks, Minneapolis, MN, USA, 30 September–3 October 1990; pp. 343–350.
33. Piętak, A.; Mikulski, M. On the adaptation of CAN BUS network for use in the ship electronic systems. *Pol. Marit. Res.* **2009**, *16*, 62–69. [CrossRef]

34. Jurdana, I.; Tomas, V.; Ivce, R. Availability model of optical communication network for ship's engines control. In Proceedings of the 2011 3rd International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT), Budapest, Hungary, 5–7 October 2011; pp. 1–6.
35. Henry, M.; Iacovelli, M.; Thatcher, J. DDG 1000 Engineering Control System (ECS). In Proceedings of the ASNE Intelligent Ship VIII Symposium, Philadelphia, PA, USA, 20–21 May 2009; pp. 12–26.
36. Liu, S.; Xing, B.; Zhi, P.; Li, B. Design of semi-physical simulation platform for ship cooperative control system. In Proceeding of the 11th World Congress on Intelligent Control and Automation, Shenyang, China, 29 June–4 July 2015; pp. 5962–5966.
37. Liu, S.; Chang, X.C.; Li, G.Y. Synchronous-ballistic control for a twin-rudder ship. *Control Theory Appl.* **2010**, *12*, 1631–1636.
38. Xing, B.; Liu, S.; Zhu, W. Actuator channel setting strategy for ship information systems based on reachability analysis and physical characteristic. In Proceedings of the 2015 IEEE 15th International Conference on Environment and Electrical Engineering (EEEIC), Rome, Italy, 10–13 June 2015; pp. 932–937.
39. Liu, S.; Xing, B.W.; Chen, X.; Zhi, P. Design of data flow for ship information system. *Ship Sci. Technol.* **2016**, *4*, 110–115.
40. Liu, S.; Xing, B.; Zhu, W. A fusion Fuzzy PID controller with real-time implementation on a ship course control system. In Proceedings of the 2015 23rd Mediterranean Conference on Control and Automation (MED), Torremolinos, Spain, 16–19 June 2015; pp. 916–920.

© 2018 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (http://creativecommons.org/licenses/by/4.0/).