# INTRUSION DETECTION METHOD FOR INDUSTRIAL CONTROL SYSTEMS USING SINGULAR SPECTRUM ANALYSIS

ASUKA TERAI[1], TATSUYA CHIBA[1], HIDEYUKI SHINTANI[2], SHOYA KOJIMA[2], SHINGO ABE[2] & ICHIRO KOSHIJIMA[2]
[1]Future University Hakodate, Japan
[2]Nagoya Institute of Technology, Japan

ABSTRACT

Because of their automated processing capabilities, industrial control systems (ICSs) currently play a crucial role in plant operations. It was not long before ICS had been completely insulated from the Internet. However, because of the improved reliability of ICS devices and systems, we could find only a few plants that did not use ICS in conjunction with the Internet. As a result, the extended accessibility of almost every ICS component makes such systems vulnerable to cyber-attacks. Because of this, intrusion detection systems, which monitor ICS network traffic and detect suspicious activities within the components themselves, are extremely important. Previous studies argued that packet intervals could ideally be regarded as indicators of the hazardous status of ICSs against hacking activities, and proposed intrusion detection methodologies relying solely on packet intervals. However, these methodologies with supervised machine-learning have inevitably been compromised by cyber-attacks whose characteristics are different than those of the training dataset. We hypothesize that packet intervals in an ICS network used for automated industrial processes, which are forced to produce a certain type of periodicity, reflect a particular type of packet interval patterns. In other words, certain anomalous behaviors never fail to interfere with this pattern. This paper proposes an intrusion detection method using a singular spectrum analysis to monitor time series packets. We evaluated our proposed method on our cybersecurity testbed using penetration tests. The results verified the validity of our system realized in the packet interval periodicity. Furthermore, we examined the optimum parameter set for the singular spectrum analysis in the proposed method. From this experiment, we successfully designated criteria for the parameter-set based on the period of the packet intervals during normal operations. The proposed method successfully detected all three types of attacks within 4 sec, without producing a false alert during normal operations.

*Keywords: intrusion detection, industrial control system, packet pattern, singular spectrum analysis.*

## 1 INTRODUCTION

Industrial plants operate using industrial control systems (ICSs), which monitor and control physical infrastructure and equipment. If ICSs of a critical infrastructure (such as an electric power plant) were compromised, it may have a serious impact on social life. Based on the rapid progress of digital technology, the demand to integrate ICS components with the Internet cannot be ignored. Due to the network extendibility of such ICS components, it becomes vulnerable to cyber-attacks.

In general, cyber-attacks have been used to acquire intellectual property and personal information. Cyber-attacks targeting ICS used in critical infrastructure can interfere with plant/facility operation and reduce the safety of human life. Perhaps, it seems that leakage of technical information of some high-level aggregates hacker in both quality and quantity, resulting in more frequent and sophisticated cyber-attack against ICS. In fact, a 2010 cyber-attack ruined almost one-fifth of Iran's nuclear centrifuges and caused substantial damage to Iran's nuclear program [1]. Additionally, Ukraine's power grid was attacked in 2015 and 2016 [2]. Intrusion detection systems (IDSs), which can be used to monitor ICS network traffic, are therefore more important than ever.

WIT Transactions on Engineering Sciences, Vol 121, © 2018 WIT Press
www.witpress.com, ISSN 1743-3533 (on-line)
doi:10.2495/RISK180171

IDSs are typically used to monitor network traffic and alert system administrators or network administrators when suspicious activity is detected. There are two types of IDSs: signature-based IDSs and anomaly-based IDSs. Signature-based IDSs detect threats by probing for specific patterns created by malware. Although signature-based IDSs can protect a system from known attacks, they cannot screen out attacks that are not registered in the database. In contrast, anomaly-based IDSs, which apply machine learning techniques, monitor network traffic and compare it to an established baseline. An anomaly-based approach allows previously unknown attacks to be detected.

Many machine-learning techniques have been used to increase the accurate detection of IDSs: k-nearest neighbour (kNN) methods (e.g. [3], [4]), neural networks (NNs) (e.g. [5], [6]), support vector machines (SVMs) (e.g. [5]–[8]), random forests [9], naive Bayes methods (e.g. [10], [11]), and time series association data mining [12]. However, supervised machine-learning techniques require a labelled dataset. Our previous IDS for ICSs used a dataset obtained by penetration tests to create a discriminant model [8]. However, these supervised machine-learning approaches could fail to detect new cyber-attacks whose characteristics were unexpected by the penetration tests. Unsupervised machine-learning infers the function of a hidden structure from unlabelled data. Therefore, when a normal ICS network exhibits a particular pattern, the techniques can effectively detect anomalous behaviour that interferes with that pattern.

Matta et al. [13] demonstrated the possibility that the cyber-attack was detected by a process involving only a comparison of packet intervals. The communication profile of an ICS could be represented by the packets intervals between the target hosts A and B. The researchers illustrated the packet intervals of a pseudo cyber-attack (penetration test) and showed a difference in communication profiles under the penetration tests compared with the normal operational profile. This previous work suggested that the normal network traffic produced by each facility tends to have a specific pattern, and that a cyber-attack might disturb that pattern.

In this paper, the authors hypothesize that an automated production process using ICSs is forced to produce a certain type of periodicity that results from each plant's activities, which is observed in time-series packet intervals. Simultaneously, this particular periodicity will be disturbed by hacking or intruding activities. Therefore, we propose an intrusion detection method based on a singular spectrum analysis. We evaluate the proposed method using penetration tests on our cybersecurity testbed.

## 2 INTRUSION DETECTION SYSTEM USING SINGULAR SPECTRUM ANALYSIS

### 2.1 Packet intervals

The previous work [13] suggested that packet intervals reflect the characteristics of packets in a typical ICS network. In a typical ICS network, there are IP communications between the object linking and embedding (OLE) for process control (OPC) server and the single loop controller (SLC) (programmable logic controller (PLC)). ICS communication transfers packets specified as industrial control protocols, such as Modbus/TCP [14], at specific time intervals. Packets that continue to the target machine are assumed as $\{p_0, p_1, p_2, \ldots, p_n\}$. The time stamp of the $i$th packet is represented as $t_i$. The arrival intervals $d_i$ are defined in accordance with the time stamps $t_i$ and $t_{i-1}$ as

$$d_i = t_i - t_{i-1}. \tag{1}$$

The time-series packet intervals $\{d_1,d_2,\ldots, d_n\}$ can be considered to have a type of periodicity, because they are forced by the activities of a plant to produce a certain type of periodicity. Thus, we used a singular spectrum analysis to detect disturbances caused by intruding activities.

### 2.2 Singular spectrum analysis for detecting structural change

A singular spectrum analysis is a nonparametric spectral estimation method. The analysis decomposes time-series data into a sum of components – it is applied sequentially to the initial parts of the series. The analysis constructs the corresponding subspaces and checks the distances between these subspaces and the lagged vectors that are formed from the few most recent observations. If these distances, which are referred to as variability, become too large, a structural change is suspected to have occurred in the series [15]. The technique could be used to detect changes not only in trends but also in the variability of a series. Therefore, the technique has applied to various engineering problems, like cognitive radio networks [16], and global navigation satellite system carrier phase signals [17].

First, a test matrix and trajectory matrix were defined using time-series packet intervals $\{d_1,d_2,\ldots, d_n\}$ (Fig. 1). The test matrix for the $i$th packet is

$$Z^{(i)} = \begin{pmatrix} d_{i-M-K+2} & d_{i-M-K+3} & \cdots & d_{i-M+1} \\ d_{i-M-K+3} & d_{i-M-K+4} & \cdots & d_{i-M+2} \\ \vdots & \vdots & \ddots & \vdots \\ d_{i-K+1} & d_{i-K+2} & \cdots & d_i \end{pmatrix}, \tag{2}$$

where M is the size of the sliding-window and the number of columns $K \leq M$. Additionally, a trajectory matrix is defined with lag L.

$$X^{(i)} = \begin{pmatrix} d_{i-L-M-K+2} & d_{i-L-M-K+3} & \cdots & d_{i-L-M+1} \\ d_{i-L-M-K+3} & d_{i-L-M-K+4} & \cdots & d_{i-L-M+2} \\ \vdots & \vdots & \ddots & \vdots \\ d_{i-L-K+1} & d_{i-L-K+2} & \cdots & d_{i-L} \end{pmatrix}. \tag{3}$$

A singular value decomposition (SVD) of the test and trajectory matrices is performed.

$$Z^{(i)} = U^{(i)}\Sigma^{(i)}V^{(i)T}, \tag{4}$$

$$X^{(i)} = Q^{(i)}\Gamma^{(i)}P^{(i)T}, \tag{5}$$

where

$$\Sigma^{(i)} = \begin{pmatrix} \sigma_1^{(i)} & & \\ & \ddots & \\ & & \sigma_m^{(i)} \end{pmatrix}, \qquad \sigma_1^{(i)} \geq \cdots \geq \sigma_m^{(i)}, \tag{6}$$

$$\Gamma^{(i)} = \begin{pmatrix} \gamma_1^{(i)} & & \\ & \ddots & \\ & & \gamma_m^{(i)} \end{pmatrix}, \qquad \gamma_1^{(i)} \geq \cdots \geq \gamma_m^{(i)}. \tag{7}$$

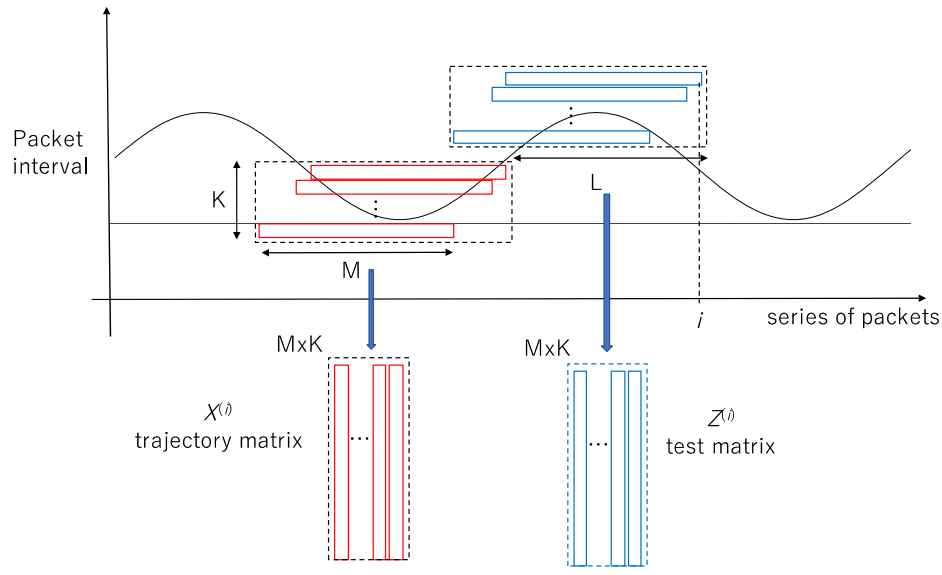


Figure 1: Definition of trajectory and test matrices.

The values $\sigma_j^{(i)}, \gamma_j^{(i)}$ are called the singular values. When the $r$ largest singular values are selected, their corresponding singular vectors from $U^{(i)}, Q^{(i)}$ are:

$$U_r^{(i)} = [\boldsymbol{u}_1^{(i)}, \boldsymbol{u}_2^{(i)}, \cdots, \boldsymbol{u}_r^{(i)}] , \tag{8}$$

$$Q_r^{(i)} = [\boldsymbol{q}_1^{(i)}, \boldsymbol{q}_2^{(i)}, \cdots, \boldsymbol{q}_r^{(i)}] . \tag{9}$$

These matrices represent the principal trajectory subspaces and test matrices, respectively. The difference between principal subspaces is defined as a change score at the *i*th packet:

$$a(i) = 1 - \left\| {U_r^{(i)}}^T Q_r^{(i)} \right\|_2 , \tag{10}$$

where $\|\cdot\|_2$ indicates the matrix norm.

$$\|A\|_2 = \max_{\boldsymbol{x} \neq 0} \frac{\|A\boldsymbol{x}\|}{\|\boldsymbol{x}\|}. \tag{11}$$

Then, $\left\| {U_r^{(i)}}^T Q_r^{(i)} \right\|_2$ is the largest singular value of ${U_r^{(i)}}^T Q_r^{(i)}$. When the change score $a(i)$ exceeds a threshold $\theta$, it is estimated that there are suspicious activities on the ICS network.

Some parameters in the singular spectrum analysis are used to detect structural change. The parameters used in this research are shown in Table 1.

## 3 EVALUATION ENVIRONMENT

The proposed method was evaluated using datasets obtained from the testbed prepared for previous research [13].

Table 1: Parameters for singular spectrum analysis.

| Parameter | Explanation |
|---|---|
| $M$ | Sliding-window size of trajectory/test matrix |
| $K$ | Number of columns in trajectory/test matrix |
| $L$ | Lag between trajectory and test matrices |
| $r$ | Number of selected left singular vectors |

### 3.1 Security testbed

Fig. 2 shows a piping and instrumentation (P&I) diagram of our cybersecurity testbed where water is heated to be circulated between two tanks. The testbed was equipped with actual control devices and controlled automatically. Yokogawa Digital Indicating Controllers (model number: UT35A and UT32A) were installed for a proportional-integral-derivative (PID) control. The controllers are operated using an ICS network.

The ICS network diagram and its configuration are shown in Fig. 3. There are three zones: one supervisory zone, and two control zones (ICS1/ICS2). The two control zones have the same structure, which consists of a gateway server, object process control (OPC) server, supervisory control and data acquisition (SCADA) monitor, and SLCs. SLC1 controls the level of Tank1 and monitors the temperature of Tank2. SLC2 controls the inlet from Tank2 and the temperature of Tank1. Additionally, SLC2 monitors the level of Tank2. In the ICS-2 network, a network tap was installed to capture the OPC2 packets during normal operation and penetration attacks as evaluation datasets. This configuration is designed by Hashimoto et al. [18].

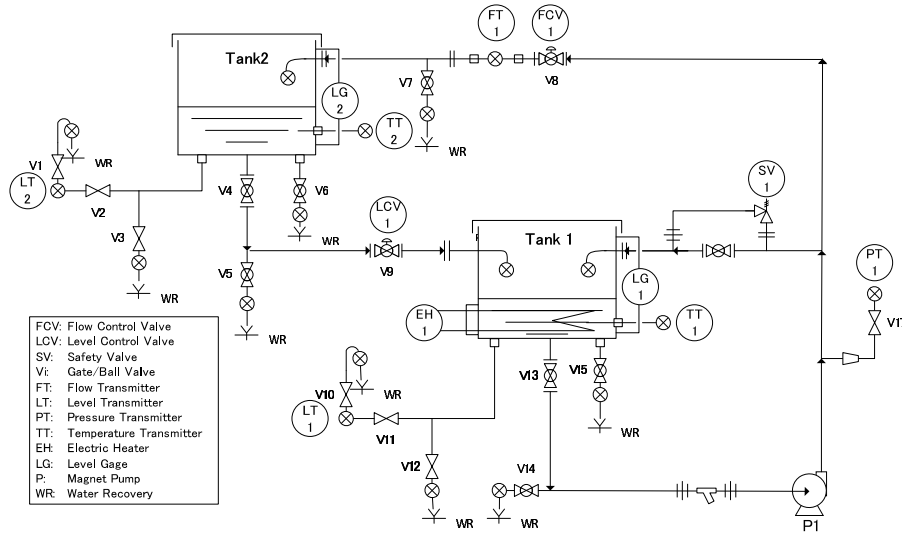


Figure 2: P&I diagram of the cybersecurity testbed.

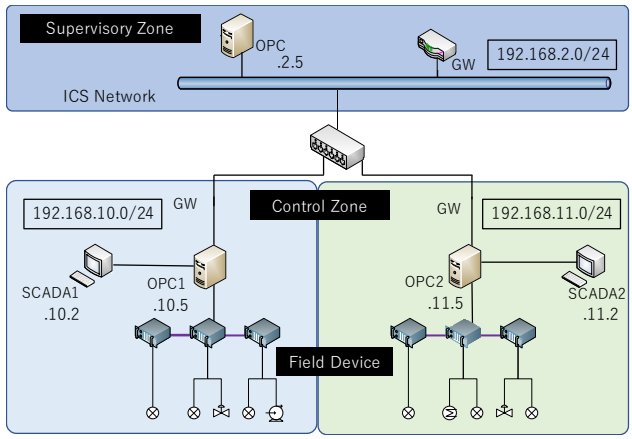


Figure 3: ICS network diagram of cybersecurity testbed.

### 3.2 Evaluation packets

During normal operations, the OPC servers that collect and exchange process data are monitored by the SCADA terminals, to maintain the water in Tanks 1 and 2 at constant levels. The packets that were sent to the controllers in the ICS-2 network were captured using the popular cross-platform packet-capturing program Wireshark. The objective of the penetration test was to crack the target OPC2 and tamper with the configuration file using the Metasploit Framework (Rapid7) attack tool. The penetration test was demonstrated for three types of cyber-attacks: reading registers, finding unit IDs, and reading coils. Table 2 presents the details of the datasets used for both normal operations and penetration test attacks.

## 4 EVALUATION RESULTS

### 4.1 Estimation of cycle period of packet intervals during normal operations

The packet intervals during normal operations have a certain type of periodicity (see Fig. 4: top). To set the parameter values of the singular spectrum analysis based on a cycle period of the packet intervals during normal operations, the cycle period was estimated using an autocorrelation analysis. The autocorrelation results are shown in Fig. 4. According to the autocorrelation analysis results, the cycle period was 8 packets.

Table 2: Datasets used for normal operations and penetration test attacks.

| Dataset | No. of packets (Modbus/TCP) | Capture period (sec) | No. of attack packets |
|---|---|---|---|
| Normal | 628 | 82.22 | - |
| Reading registers | 708 | 92.26 | 3 |
| Finding unit IDs | 1812 | 834.9 | 254 |
| Reading coils | 502 | 65.28 | 2 |

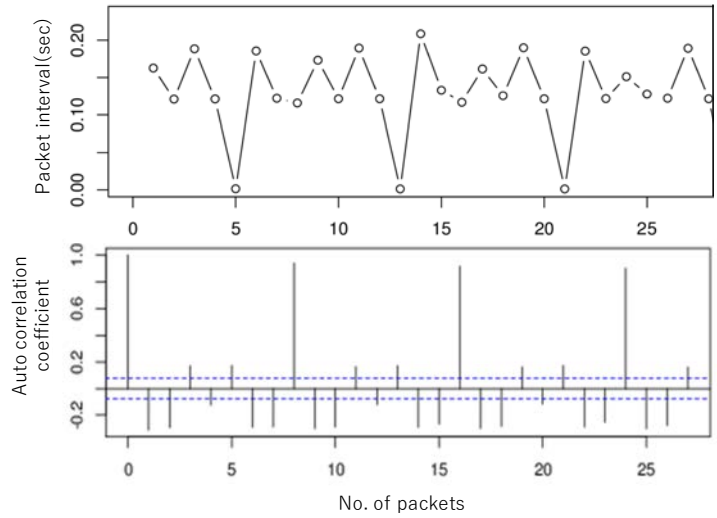


Figure 4: Packet intervals during normal operations and autocorrelation coefficients.

## 4.2 Evaluation criterions

The change score threshold for intrusion detection was set based on the change scores for the dataset during normal operations. We hypothesized that the change scores during normal operations would obey a normal distribution, and that the top 0.05% of change scores would indicate suspicious activities. Therefore, the change score threshold was defined as follows:

$$\theta = \mu + 3.29 * sd, \tag{12}$$

where $\mu$ and $sd$ indicate the average and standard deviation of the change scores during normal operations. When the change score for a packet exceeds the threshold $\theta$, the system estimates that the packet reflects an attack and it alerts the system administrators to the cyber-attack.

In some cases, the system estimates packets in normal dataset as an attack, and it causes the issues of false alerts. The system is designed to detect a cyber-attack by distinguishing between the normal and abnormal cycle periods of the packet intervals. The deviance from normal behaviour triggers an alert and it continues until the detection of the normal cycle period. Therefore, the error rate based on the number of estimated attack packets is inadequate to evaluate the system. In this research, we used the detection time for the first attack packet and the other two criteria based on the alert timing (time difference between the real attack and the alert; Fig. 5) to evaluate the system. One criterion was the maximum time differences between a real attack packet and the closest estimated attack (the maximum time difference based on real attack). The criterion indicates the maximum time to detect cyber-attack. The other criterion was the maximum time differences between an estimated attack and the closest real attack packet (the maximum time difference based on estimated attack). When the system indicates a cyber-attack during a period without a real attack, the criterion has a large value.

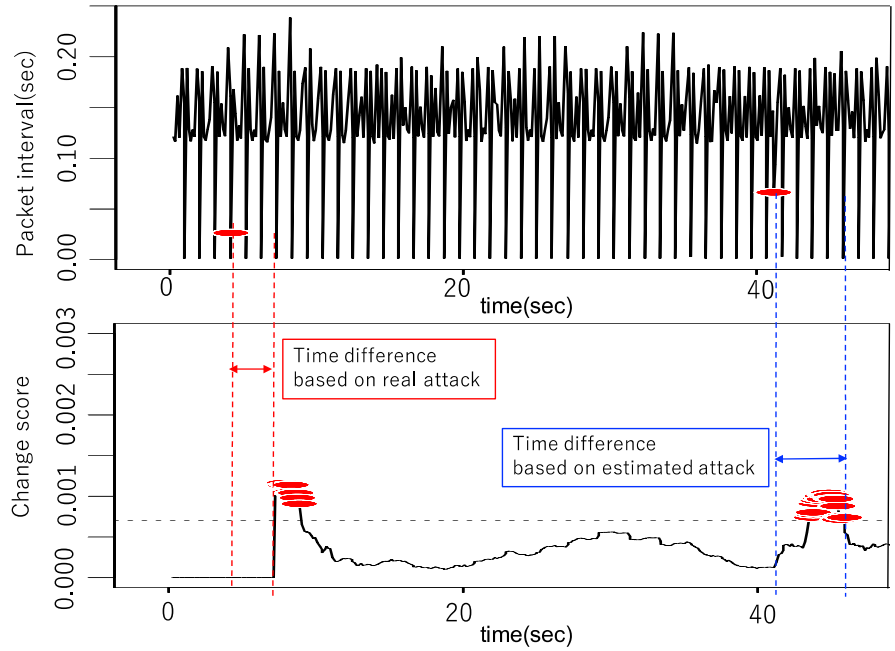


Figure 5: Image of two criteria based on the alert timing. The upper figure shows packet intervals and red dots indicate real attacks. The lower figure shows change scores and red dots indicate estimated attacks (alert) and the broken line indicates the threshold. Red lines indicate the maximum time difference based on a real attack packet and blue lines show the maximum time difference based on an estimated attack.

### 4.3 Parameter settings

The parameter values ($M$, $K$, $L$) of the singular spectrum analysis for the system were optimized. The number of singular vectors ($r$) was fixed at two.

#### 4.3.1 Fixed rate setting

The ratio of the parameters was fixed. K = M/2, L = M/4, and M were defined using integral multiplication of the cycle period during normal operations (M = 8, 16, 24, 32). The detection results are presented in Table 3. The systems estimate no attack for normal operational datasets, except for a system with (M, K, L) = (16, 8, 4). The system with (M, K, L) = (16, 8, 4) required the least amount of time to detect the first attack packet. However, one of maximum time differences based on estimated attack was longer than 10 sec, and the system produced a false alert during normal operations. In contrast, the maximum time differences for a system with (M, K, L) = (32, 16, 8) were less than 10 sec. Thus, the system is considered to exhibit the best performance.

#### 4.3.2 Fixed lag between the trajectory and the test matrices

The lag between the trajectory and test matrices was fixed at L = 8, and M = 8, 16, 24, 32. K was defined as an integral multiplication of the cycle period under the condition $K \leq M$.

Some of the systems produced false alerts for normal dataset. The detection results for the systems without false alert for normal dataset are shown in Table 4. Similarly, for the systems with a fixed value of L=8, the system with (M, K, L) = (32, 16, 8) performed the best with respect to 1) the time required to detect the first attack packet and 2) the maximum time difference based on the estimated attack.

Table 3: The detection results with fixed rate parameters.

| Dataset | M | K | L | Time required to detect first attack packet (sec) | Maximum time difference: real attack (sec) | Maximum time difference: estimated attack (sec) |
|---|---|---|---|---|---|---|
| Reading registers | 8 | 4 | 2 | 69.2 | 69.2 | 1.20 |
| Finding unit IDs | | | | 2.07 | 9.37 | 11.7 |
| Reading coils | | | | 25.6 | 25.6 | 12.9 |
| Reading registers | 16 | 8 | 4 | 0.17 | 0.30 | 3.28 |
| Finding unit IDs | | | | 0.15 | 0.15 | 3.36 |
| Reading coils | | | | 0.07 | 0.07 | 12.1 |
| Reading registers | 24 | 12 | 6 | 2.05 | 32.5 | 4.65 |
| Finding unit IDs | | | | 0.66 | 14.4 | 11.2 |
| Reading coils | | | | 2.21 | 2.21 | 11.1 |
| Reading registers | 32 | 16 | 8 | 3.09 | 3.09 | 6.48 |
| Finding unit IDs | | | | 0.79 | 0.79 | 7.10 |
| Reading coils | | | | 0.85 | 0.85 | 6.80 |

Table 4: The detection results with fixed lag time.

| Dataset | M | K | L | Time required to detect the first attack packet (sec) | Maximum time difference: real attack (sec) | Maximum time difference: estimated attack (sec) |
|---|---|---|---|---|---|---|
| Reading registers | 24 | 8 | 8 | 1.02 | 1.02 | 4.81 |
| Finding unit IDs | | | | 0.79 | 3.05 | 5.01 |
| Reading coils | | | | 0.41 | 0.41 | 12.1 |
| Reading registers | 32 | 8 | 8 | 2.05 | 2.05 | 5.86 |
| Finding unit IDs | | | | 0.79 | 0.79 | 6.03 |
| Reading coils | | | | 0.41 | 0.41 | 12.1 |
| Reading registers | 32 | 16 | 8 | 3.09 | 3.09 | 6.48 |
| Finding unit IDs | | | | 0.79 | 0.79 | 7.10 |
| Reading coils | | | | 0.85 | 0.85 | 6.80 |
| Reading registers | 32 | 32 | 8 | 5.20 | 5.20 | 8.92 |
| Finding unit IDs | | | | 3.00 | 3.00 | 9.18 |
| Reading coils | | | | 2.91 | 2.91 | 8.87 |

### 4.3.3 Detection results of the system

To demonstrate the detection ability of the system, both the time series for the packet intervals and the change scores are shown in Fig. 6. These figures show that a cyber-attack interferes with periodic patterns during normal operations and indicates that the system can use the change score to detect this interference.

## 5 CONCLUSION REMARKS

The authors confirmed that the timing of cyber-attack packets on an ICS network was associated with a periodic pattern. In this paper, we proposed an intrusion detection method using a singular spectrum analysis. Additionally, the proposed method was evaluated using pseudo-attacks on our cybersecurity testbed. When the parameters of the singular spectrum analysis were set based on the cycle period during normal operations ($T$), the performance

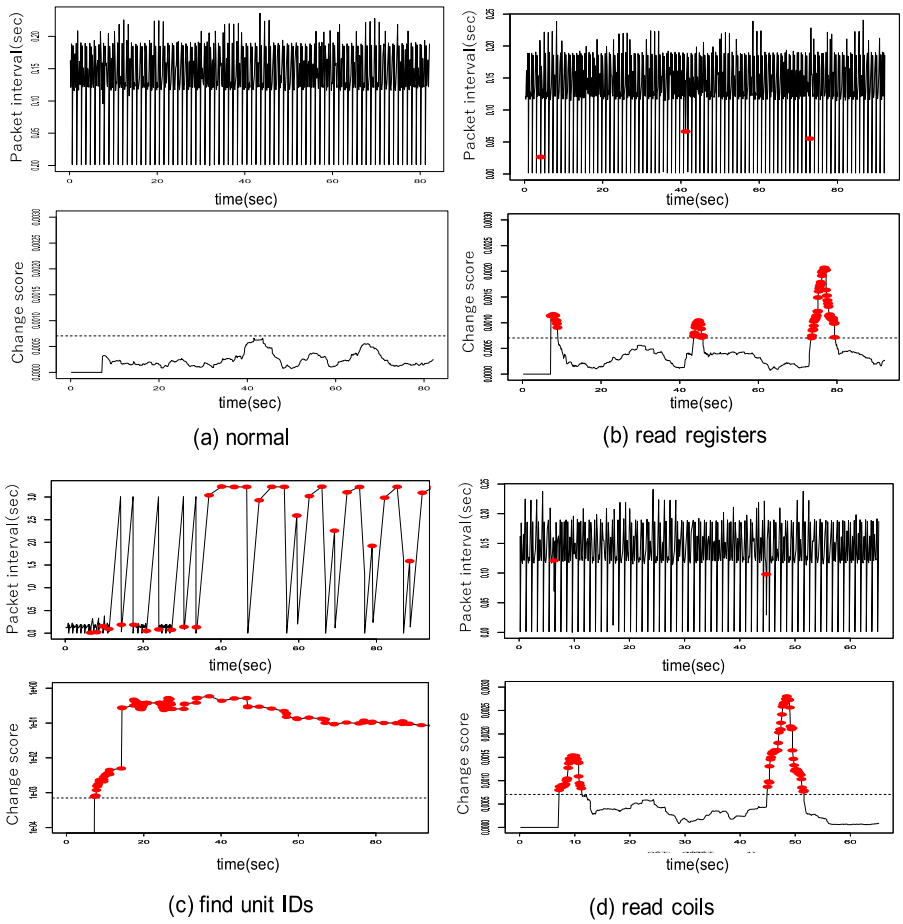


Figure 6: Packet intervals and change scores. (Upper figures show packet intervals and lower figures show change scores. Red dots indicate real or estimated attacks and the broken lines indicate the threshold.)

was the best for a system with a sliding-window size $M = 4T$, number of columns $K = 2T$, and lag $L = T$. The system could detect the first attack packet in under 4 sec.

Because the system detects unexpected packets delivered to controllers in ICS networks, an IDS was prepared for each OPC server to monitor the overall ICS. However, the system detects all behavior that results in changes to normal operation. When ICS operators make intentional changes to normal operations, the system flags such changes as anomalous. Therefore, an alert filtering system must be developed for the proposed IDS to ignore changes made by operators.

ACKNOWLEDGEMENT

The research was partially supported by the Ministry of Education Science, Sports and Culture, Grant-in-Aid for Scientific Research (A), No. 16H01837.

REFERENCES

[1] Kelley, M.B., The Stuxnet attack on Iran's nuclear plant was "far more dangerous" than previously thought, *Business Insider*, Online. http://www.businessinsider.com/stuxnet-was-far-more-dangerous-than-previous-thought-2013-11, 2013.

[2] Cherepanov, A. & Lipovsky, R., Industroyer: Biggest threat to industrial control systems since Stuxnet, 2017. *WeLiveSecurity*. www.welivesecurity.com/2017/06/12/industroyer-biggest-threat-industrial-control-systems-since-stuxnet/.

[3] Liao, Y. & Vemuri, V., Use of k-nearest neighbor classifier for intrusion detection. *Computer and Security*, **21**(5), pp. 439–448, 2002.

[4] Wang, K. & Stolfo, S.J., Anomalous Payload-based network intrusion detection. *Proceedings of Recent Advance in Intrusion Detection (RAID2004)*, pp. 203–222, 2004.

[5] Mukkamala, S., Janoski, G. & Sung, A., Intrusion detection using neural networks and support vector machines. *Proceedings of the 2002 International Joint Conference on Neural Networks. IJCNN02 (Cat. No. 02CH37290)*, **2**, pp. 1702–1707, 2002.

[6] Chen, W.H., Hsu, S.H. & Shen, H.P., Application of SVM and ANN for intrusion detection. *Computer and Operations Research*, **32**, pp. 2617–2634, 2005.

[7] Mukkamala, S., Sung, A.H. & Abraham, A., Intrusion detection using an ensemble of intelligent paradigms. *Network and Computer Applications*, **28**, pp. 167–182, 2005.

[8] Terai, A., Abe, S., Kojima, S., Takano, Y. & Koshijima, I., Cyber-attack detection for industrial control system monitoring with support vector machine based on communication profile. *2017 IEEE European Symposium on Security and Privacy Workshop*, pp. 132–138, 2017.

[9] Zhang, J. & Zulkern, M., Anomaly Based Network Intrusion Detection with Unsupervised Outlier Detection, 1-4244-0355-3, 2006.

[10] Koc, L., Mazzuchi, T.A. & Sarkani, S., A network intrusion detection system based on a Hidden Naive Bayes multiclass classifier. *Expert Systems with Applications*, **39**, pp. 13492–13500, 2012.

[11] Abd-Eldayem, M.M., A proposed HTTP service-based IDS. *Egyptian Informatics Journal*, **15**, pp. 13–24, 2014.

[12] He, W., Hu, G. & Yao, X., Large-scale communication network behavior analysis and feature extraction using multiple motif pattern association rule mining. *WSEAS Transactions on Communications*, **5**(8), 473–482, 2009.

[13] Matta, M. et al., Industrial control system monitoring based on communication profile. *Journal of Chemical Engineering of Japan*, **8**, pp. 619–625, 2015.

[14] Modicon Inc., *Modicon Modbus Protocol Reference Guide*, North Andover, USA, 1996.
[15] Moskvina, V. & Zhigljavsky, A., An algorithm based on singular spectrum analysis for change-point detection. *Communications in Statistics Simulation and Computation* **32**, pp. 319–352, 2003.
[16] Dong, Q., Yang, Z., Chen, Y., Li, X. & Zeng, K., Anomaly detection in cognitive radio networks exploiting singular spectrum analysis. *Proceedings of International Conference on Mathematical Methods, Models, and Architectures for Computer Network Security (LNCS10446)*, Springer, pp. 247–259, 2017.
[17] Mazher, K. & Tahir, M., Small cycle slip detection using singular spectrum analysis. *Proceedings of the 24th European Signal Processing Conference,* pp. 1053–1057, 2016.
[18] Hashimoto, Y. et al., Safety securing approach against cyber-attacks for process control system. *Computers and Chemical Engineering*, **57**, 181–186, 2013.