GAN-based Data Augmentation in the Design of Cyber-attack Detection Methods

Dušan Nedeljković and Živana Jakovljević, *Member, IEEE*

***Abstract*—The advent of the Industry 4.0 paradigm that relies on the concepts of Cyber-Physical Systems (CPS) and the Industrial Internet of Things (IIoT) leads to the transition from centralized to distributed control. In this approach, interconnected smart devices (sensors, actuators, etc.) as the key enablers achieve system control through coordinated work. Introduction of IIoT leads to ubiquitous communication between smart devices, thus opening up a vast area for potential malicious threats and attacks which can cause serious consequences, take to system dysfunction or even endanger human lives. Therefore, security mechanisms have to be developed to provide timely detection of different cyber-attacks and to keep the system safe and protected. Since industrial processes are often very complex and their analytical model is very difficult to determine, deep learning based methods for cyber-security mechanisms development are imposed as a technique of choice. Successful employment of data-driven solutions, particularly based on deep learning approaches usually requires a big amount of data. However, due to various limitations in the acquisition of data from the real process, its availability is still a major challenge. For instance, the Industry 4.0 factory implies frequent reconfiguration which reduces the time intervals available for experimental procedures such as data acquisition. One of the ways to deal with this issue is called data augmentation. In this paper, we apply data augmentation in the design of cyber-attack detection methods in Industrial Control Systems (ICS). In particular, we explore the possibilities for utilization of Generative Adversarial Networks (GAN) to generate the necessary amount of data for deep learning based modeling using a relatively small number of available samples on input.**

***Index Terms*—Data augmentation; Cyber security; Generative Adversarial Networks; Deep learning; Convolutional Neural Networks.**

I. Introduction

IMPLEMENTATION of Cyber-Physical Systems (CPS)-based smart devices at industrial plants represents the basis for the digitization of manufacturing processes and leads to the next step in industrial evolution known as Industry 4.0 [1]. Industrial Control Systems (ICS) embrace Industrial Internet of Things (IIoT) and go through significant changes. In addition to enormous benefits, introduction of IIoT at shop-floor has a number of drawbacks, where the ubiquitous wired or wireless communication between smart devices and connection of ICS to Internet can be singled out. Since ICS are no longer isolated, communication links between IIoT devices become vulnerable for the attacks by different malicious adversaries (Fig. 1). To address this issue specially designed systems for ICS cyber-attacks detection and communication links protection are necessary.

Considering the real-time operation of ICS and safety related issues such as catastrophic damages or even threat to human lives that cyber-attacks can bring about, the requirements for cyber-security systems in ICS differ from general information technologies (IT). Opposite to general IT where the data confidentiality is paramount, in ICS data availability, along with its integrity, is the most important [2, 3]. Another key aspect of ICS is that expected lifetime of ICS components (at least 10-15 years) is significantly longer than in general IT (3-5 years) [4] and that, as a result, ICS contain a large number of devices that use legacy communication protocols that do not even utilize the basic protection mechanisms such as authentication [5]. In addition to multilayered protection mechanisms based on network segmentation and segregation, one of the key concepts for cyber-security in ICS is Defense-in-Depth. This concept implies the ability of the devices within ICS to recognize an attack if it bypasses previous layers and that the device should do this before the attack achieves the desired effects [6]. For these purposes host based (installed on the device) Intrusion Detection Systems (IDS) are developed.

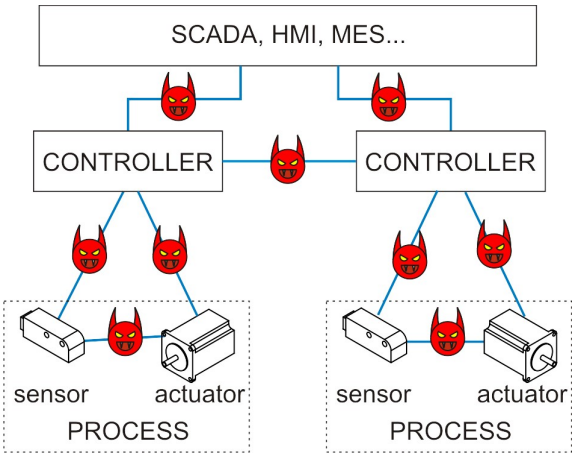

Fig. 1. Cyber-attacks on communication links in ICS within Industry 4.0

Generally, IDS within ICS are based on the model of the system behavior or the data communicated between system

Dušan Nedeljković, MSc (ME) is with the Faculty of Mechanical Engineering, University of Belgrade, 16 Kraljice Marije, 11000 Belgrade, Serbia (e-mail: dnedeljkovic@mas.bg.ac.rs).

Prof. Dr. Živana Jakovljević is with the Faculty of Mechanical Engineering, University of Belgrade, 16 Kraljice Marije, 11000 Belgrade, Serbia (e-mail: zjakovljevic@mas.bg.ac.rs).

 ISBN 978-86-7466-930-3

elements, and the attack is detected as the discrepancy between modeled and exhibited behavior of the system or as the discrepancy between modeled and data received through communication links. Depending on the approach for the model generation, IDS within ICS can be classified in two high level categories: design based and data based. In the design based approaches the model of the system/data is obtained in a mathematically formalized way using analytical models or different formal methods depending on the system type [7, 8].

Data based approaches, on the other hand, use the data obtained during system operation to create the model usually using different machine learning (ML) techniques. These approaches can be supervised, semi-supervised and unsupervised [9]. Supervised methods use labeled datasets containing data obtained during normal system operation and during system operation under attacks to generate detection mechanisms [10, 11]. Unsupervised methods, on the other hand, generate IDS using unlabeled data again containing the data obtained from the system performing with and without attack, and ML techniques find the structure within data themselves. The main shortcoming of the considered classes of methods is that they show low generalization properties when attacks not present during IDS creation are exhibited on the system.

Finally, semi-supervised methods use only the data obtained during normal system operation, i.e., from the system that was not subject to the attacks. They generate the model of the system behavior/communicated data during normal operation and the attack is recognized as the discrepancy between exhibited and the values estimated using the developed models. This class of approaches is most commonly utilized and shows better generalization to different kinds of attacks [12, 13, 14].

In our previous work [14] we have proposed a method for the development of IDS on communication links within ICS that is based on Convolutional Neural Networks (CNN). The proposed method, as well as the other data driven methods, requires significant amount of data from the real-world ICS that cannot be always easily obtained. In this paper we explore the possibility of augmentation of real-world data using Generative Adversarial Networks (GAN) and utilization of thus obtained dataset for IDS creation.

The reminder of the paper is structured as follows. Section 2 briefly describes GAN, whereas Section 3 refers to the utilized method for the development of IDS in ICS. Performance of IDS created using the data generated with a GAN and its comparison with IDS created based on real-world data are shown in Section 4 using an example of electro-pneumatic positioning system based on smart devices. Finally, in Section 5 we provide conclusions and future work guidelines.

II. Generative Adversarial Networks

Generative Adversarial Networks (GAN) represent a method for creation of generative model using adversarial process [15]. GAN consists of two players:

- Generator G that has the goal to generate data with the distribution close to the distribution of training data (Fig. 2.a), and
- Discriminator D with the goal to recognize if the data is created by generator or comes from the original dataset through classification of input as real or generated (fake) data (Fig. 2.b).

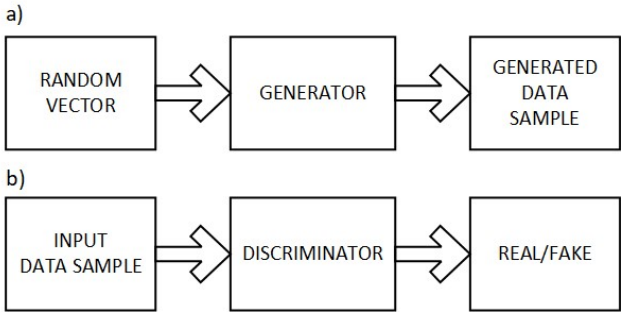

Fig. 2. Generative Adversarial Network: a) Generator, b) Discriminator

GAN training is a two player adversarial game in which generator generates fake data and tries to force discriminator to make a mistake and to recognize this data as real [16, 17]. As a rule, at the generator input is a vector of random numbers (vector of latent variables) and at its output is multidimensional vector that represents the generated data (Fig. 2.a). This data is at the input into discriminator, whereas at the discriminator output is a scalar which represents the probability that the input data is real and classifies the input accordingly (Fig. 2.b).

Generator and discriminator are trained simultaneously, where generator creates a batch of fake samples that are along with a batch of real samples from training dataset put to discriminator to classify them [16]. Based on the quality of discriminator's classification, the generator is updated to create "better" fake data and discriminator is updated to perform better classification. This adversary game repeats for a predefined number of iterations.

Generator and discriminator can be in the form of different ML based models. In our approach we will use deep neural networks (DNN), in particular CNNs and fully connected neural networks – Multilayer Perceptron (MLP).

III. CNN-based Method for the Development of IDS in ICS

The CNN-based method for the development of IDS in ICS that we have proposed in our previous work [14] belongs to the class of semi-supervised data driven methods and consists of the offline and online phase. During offline phase it generates the CNN based model of signals transmitted between IIoT devices using the data acquired during normal system operation. The model is based on auto-regression of the transmitted signals where the current value of the signal is estimated using a buffer of previously received v values. The

 ISBN 978-86-7466-930-3

main characteristic of this method is that it is designed to autonomously find the CNN with relatively small number of parameters that models the training data with good accuracy, opposite to alternative approaches that are as a rule based on manual trial and error.

IDS offline development consists of three main steps (Fig. 3). The first step represents signal preprocessing that performs FIR filtering, data structuring in ordered pairs prepared for training and data shuffling. CNN hyper-parameters (number of CNN layers, size and number of filters within them, number of pooling layers and their parameters, number of dense layers and number of neurons within them, etc.) are varied in the second step, and for each combination a CNN model is created. Finally, in the third step the generated model is selected as appropriate if it satisfies the following criteria:

1. The variance between real and estimated values should be similar for test and training data; this insures that the model is not prone to overfitting or underfitting.
2. The simulation of online performance of the IDS based on the developed model should show good performance in terms of false positive attacks detection; this insures the robustness of IDS to false attack detection online which is very important for smooth operation of ICS.

Once the model that meets given criteria is encountered, the offline procedure stops and this model is put to online detection of attacks. During online phase, the attack is detected based on the discrepancy between estimated and signal values received through communication links. If this discrepancy is higher than threshold automatically calculated from training data for z consecutive signal samples, the attack is detected.

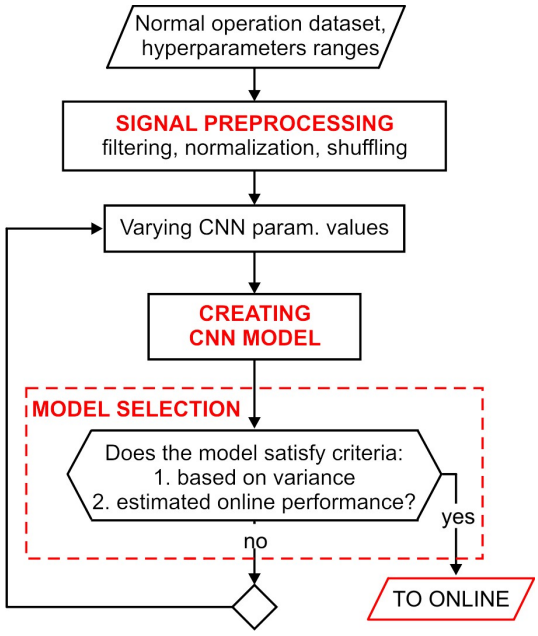

Fig. 3. Overview of the method for IDS development – offline phase

As can be observed from the presented short overview of the method, as all DNN based methods, it is highly dependent on the quality and quantity of the training data and it requires a large amount of this data at input. Acquisition of the data from the system in normal operation requires that the system is operated in isolated conditions without possibility for the attacks. Here the question arises if it is possible to operate the system long enough in such conditions to get sufficient amount of data. Another important issue that is present in Industry 4.0 factory is frequent reconfiguration of resources which leaves little room for experimenting with it. So the acquisition of appropriate amount of data from the system can be hardly feasible in some situations. For this reasons in the following section we explore the possibilities to use relatively small amount of data from real process and to augment it with data obtained using GAN.

IV. The Development of IDS for Electro-Pneumatic Positioning System based on Data Generated using GAN

In this paper we will develop IDS using GAN generated data for experimental electro-pneumatic positioning system – DisEPP (Fig. 4) that consists of:

1. Smart pneumatic cylinder, based on rodless cylinder driven by electro-pneumatic pressure regulator (EPR) that regulates pressure in 2-6 bar range on one and by mechanically controlled pressure regulator (MR) set to 4 bar on the other side, that is controlled by local controller LC1;
2. Smart encoder based on magnetic linear encoder controlled by LC2.

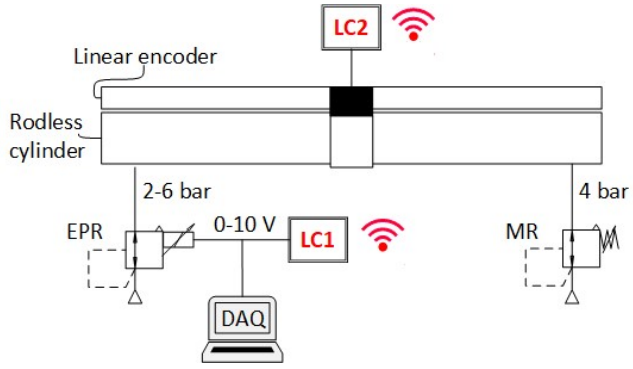

Fig. 4. A schematic representation of electro-pneumatic positioning system DisEPP

Both, LC1 and LC2 represent “mbed” devices based on ARM Cortex-M3 running at 96 MHz [18] augmented by IEEE 802.15.4-compliant wireless transceiver Microchip MRF24J40MA [19] that is used for communication between devices. The control task given in the form of desired piston positions is distributed between LC1 and LC2 in such way that: (i) LC2 has desired trajectory at input and calculates the corresponding pressure on electro-pneumatic regulator using PID and sensory signal; (ii) LC2 communicates PID output to

 ISBN 978-86-7466-930-3

LC1 using IEEE 802.15.4; (iii) LC1 converts received PID output to the 0-10 V range and puts it to electro-pneumatic pressure regulator which finally sets the pressure to 2-6 bars (proportional to the received voltage) and invokes the piston movement.

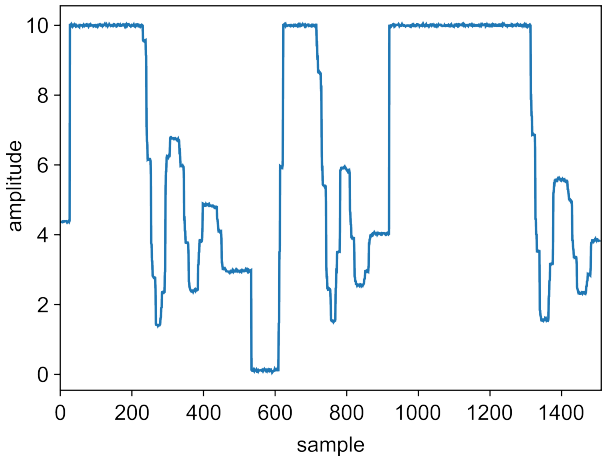

Fig. 5. An excerpt of signal acquired from DisEPP

IEEE 802.15.4 communication link between LC1 and LC2 represents vulnerable point for cyber-attacks and should be protected using IDS. The first step in IDS design using method from section III is generation of dataset representing signals obtained during normal system operation. For this purpose, we have acquired the voltage put to electro-pneumatic pressure regulator using National Instruments Data Acquisition (DAQ) system operating with 100 Hz sampling rate. A total of 399,000 samples x_i, $i \in [1, 399{,}000]$ were acquired and an excerpt of 1,500 samples is presented in Fig. 5.

A. Data Augmentation using GAN

To augment the data acquired from DisEPP we have developed a GAN with the following elements. The discriminator (Fig. 6) has the following architecture:

1. Two blocks of: (i) convolutional layer with 30 filters, each containing 10 samples, with ReLU (Rectified Linear Unit) activation function, followed by (ii) dropout layer with 0.2 dropout rate;
2. Flattening layer;
3. Output layer with one neuron.

At input of discriminator is a generated/real signal with length of 1,500 samples and estimation if this signal is generated or real (0/1) is at output.

The generator (Fig. 7), on the other hand, has the following architecture:

1. Input latent vector of 1,000 random numbers;
2. One dense layer with 100 neurons and sigmoid activation function;
3. Fully connected output layer with 1,500 neurons.

The generator generates 1,500 samples based on 1,000 random numbers.

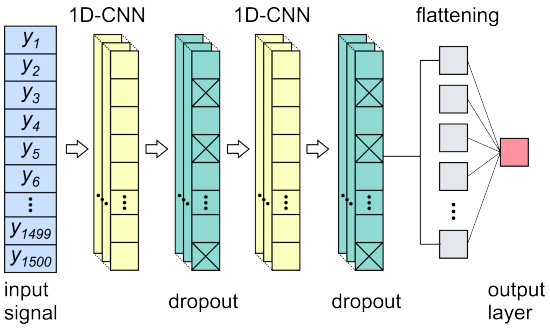

Fig. 6. The architecture of discriminator

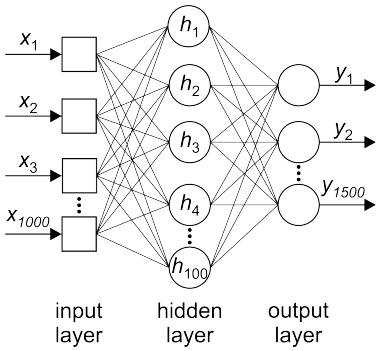

Fig. 7. The architecture of generator

For GAN training a batch of 2,000 real signals $\boldsymbol{s_i}$, $i \in [0, 1999]$, with length of 1,500 samples each, are extracted from training dataset in the following way:

$$\boldsymbol{s_i} = [x_{50i+1}, x_{50i+2}, \ldots x_{50i+1500}] \quad (1)$$

exploiting a total of 101,450 samples acquired from DisEPP. During training 500 epochs were employed.

Figure 8 presents an example of signal obtained using the trained generator. This signal is similar to the excerpt of signal obtained from real process (Fig. 5).

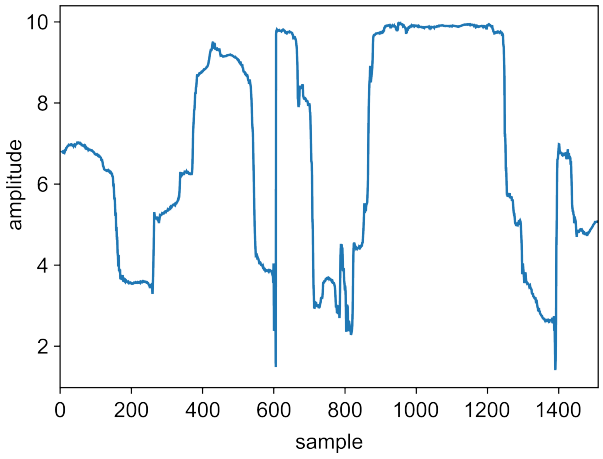

Fig. 8. An example of signal obtained using the generator

 ISBN 978-86-7466-930-3

B. *IDS creation using generated and acquired signals*

Following the procedure presented in Section III, two IDS were created: one using generated signals, and the other using real signals obtained from DisEPP. The same preprocessing procedure is applied to both signals, where the buffers of v=16 samples were used. The employed FIR filter in signal processing has pass band of [0, 0.11π], stop band of [0.35π, π], transition region in between. The filter is composed of 11 coefficients ([0.020243, 0.023017, 0.054189, 0.10397, 0.12557, 0.12344, 0.12557, 0.10397, 0.054189, 0.023017, 0.020243]) that are generated by the Parks-McClellan algorithm. After applying the filter, the signal is normalized by its maximum value. During models training, the whole datasets (signals structured into ordered pairs) are divided into training, validation and test part, with a share of 70/10/20%, respectively. Model training was performed through 5 epochs with Adam optimizer (learning rate of 0.001) and the mean squared error (MSE) cost function.

Following the procedure presented in Section III, two CNN models were developed:

1. based on signals generated using GAN where a total of 266 signals with 1,500 samples containing a total of 391,818 ordered pairs;
2. based on real signal from DisEPP that contains 399,000 samples corresponding to 398,973 ordered pairs.

For both signals (real and generated) the models with the same architecture were obtained. This architecture is composed of the following layers:

- 1D-CNN (4 filters, kernel size=2)
- 1D-CNN (8 filters, kernel size=2)
- Max pooling (pooling rate=2)
- 1D-CNN (16 filters, kernel size=2)
- 1D-CNN (16 filters, kernel size=2)
- Max pooling (pooling rate=2)
- Flattening
- Dense (30 neurons)
- Dense (1 neuron).

and it has a total of 2865 trainable parameters. The model was trained in Python v3.8.5 using a Spyder with TensorFlow v2.3.0 in the background.

In the online part of the algorithm, the detection threshold is calculated as a sum of the mean (μ) value and the triple standard deviation (σ) of discrepancies between received and estimated values over the testing data:

$$T=\mu+3\sigma \tag{2}$$

and it was T=0.00941 for the IDS based on generated and T=0.00956 for IDS based on real-world data.

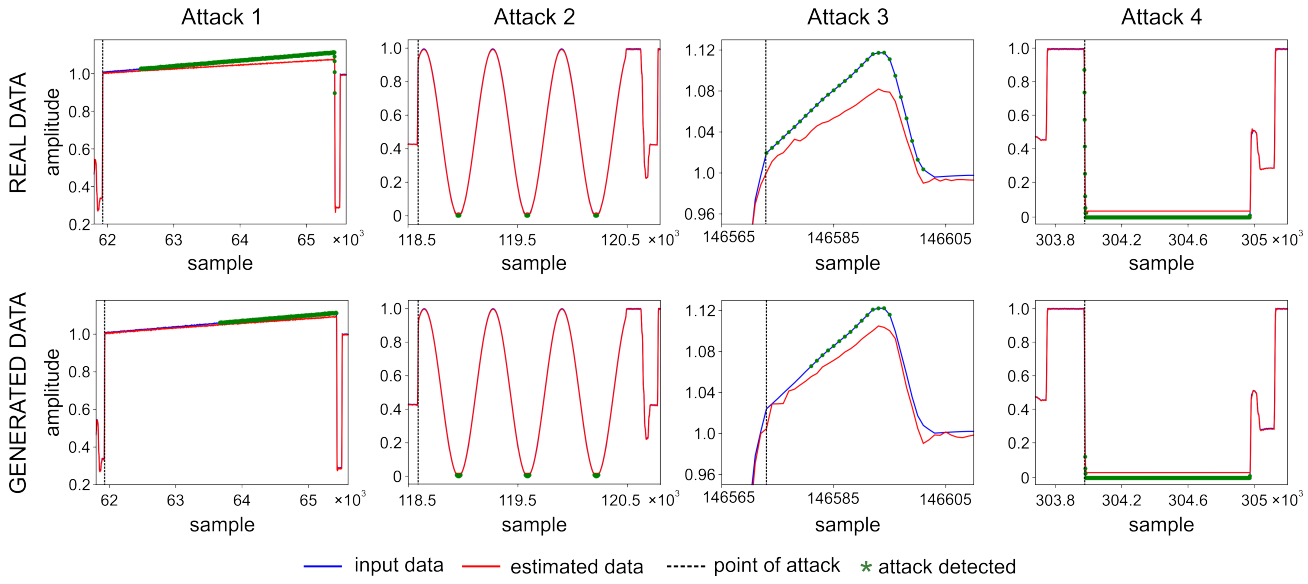

Fig. 9. Detected cyber-attacks

If the discrepancy between real and estimated value is higher than the threshold for z=15 consecutive signal samples, then the attack is detected.

To test the performance of the developed IDSs they were simulated using in Python using a number of the designed attacks. In this paper, we present and compare the performances of IDSs using four attacks with different shapes and duration. Attacks 1 and 3 increase signal value linearly by 0.00003 and 0.005 per sample, respectively, where in attack 1 random noise was introduced as well. Attack 2 utilizes sine function to generate signal value, whereas in attack 4 the signal value is set to 0 for predefined time period.

The results of attacks detection using IDSs obtained based on generated and real data are presented in Fig. 9. In this Figure input data and their estimation are shown in blue and red lines, respectively, the start of the attack is represented with a black dashed line, whereas the moments when the attack was detected are marked with green markers. Both

 ISBN 978-86-7466-930-3

models proved to be equally effective and successfully detected all attacks without false positives.

It can be noticed from the Fig. 9 that the model based on the real data detected attack 1 earlier than the model that used GAN-generated data for training. On the other hand, for attacks 2, 3, and 4, the difference between the detection moments is negligible.

V. Conclusion

In this paper, we have explored the possibilities for utilization of GAN based data augmentation in the design of IDS for ICS using an example of electro-pneumatic positioning system DisEPP based on smart devices. Using a limited number of real signal samples obtained from the DisEPP, an amount of data sufficient for deep learning based generation of IDS was generated. Using the previously developed attack detection method based on CNN, two IDS were created: one using generated signals, and the other using real signals obtained from DisEPP. To evaluate the performance of the created IDSs, a number of attacks have been created, of which four are presented in the paper. As presented in the paper, the IDS based on generated data was able to successfully detect all cyber-attacks without false positives. It presented the similar performances as IDS based on original data not only in terms of a number of detected attacks, but also in terms of attack detection latency, thus confirming that GAN augmented data can be successfully utilized for the generation of semi-supervised data based IDS in ICS.

In the future, we plan to extend our work to additional datasets that contain a higher number of signals and attacks. Furthermore, our research efforts will be directed to the application of different types of GAN and a comparison of their performances.

Acknowledgment

This research was supported by the Science Fund of the Republic of Serbia, grant No. 6523109, AI-MISSION 4.0 as well as by the Ministry of Education, Science and Technological Development of the Serbian Government under the contract No. 451-03-68/2022-14/200105.

References

[1] H. Kagermann, W. Wahlster, J. Helbig, *Recommendations for implementing the strategic initiative INDUSTRIE 4.0*, 2013. [Online]. Available: http://www.acatech.de

[2] M. R. Asghar, Q. Hu, S. Zeadally, "Cybersecurity in industrial control systems: Issues, technologies, and challenges," *Computer Networks*, vol. 165, article no. 106946, 2019.

[3] D. Upadhyay, S. Sampalli, "SCADA (Supervisory Control and Data Acquisition) systems: Vulnerability assessment and security recommendations," *Computers & Security*, vol. 89, article no. 101666, 2020.

[4] K. Stouffer, J. Falco, K. Scarfone, *Guide to industrial control systems (ICS) security*. NIST special publication, 2015.

[5] Y. Xu, Y. Yang, T. Li, J. Ju, Q. Wang, "Review on cyber vulnerabilities of communication protocols in industrial control systems," IEEE Conference on Energy Internet and Energy System Integration (EI2), pp. 1-6, IEEE, Beijing, China, Nov. 2017.

[6] Industrial Control Systems Cyber Emergency Response Team, Recommended Practice: Improving Industrial Control System Cybersecurity with Defense-in-Depth Strategies, 2016, Available: https://www.cisa.gov/uscert/sites/default/files/recommended_practices/NCCIC_ICS-CERT_Defense_in_Depth_2016_S508C.pdf, Accessed on: Mar. 2022.

[7] L. K. Carvalho, Y. C. Wu, R. Kwong, S. Lafortune, "Detection and mitigation of classes of attacks in supervisory control systems," *Automatica*, vol. 97, pp. 121-133, 2018.

[8] Z. Jakovljevic, V. Lesi, M. Pajic, "Attacks on Distributed Sequential Control in Manufacturing Automation," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 2, pp. 775-786, 2021.

[9] M. Elnour, N. Meskin, K. Khan, R. Jain, "A Dual-Isolation-Forests-Based Attack Detection Framework for Industrial Control Systems," *IEEE Access*, vol. 8, pp. 36639-36651, 2020.

[10] S. Sapkota, A. K. Mehdy, S. Reese, H. Mehrpouyan, "Falcon: Framework for anomaly detection in industrial control systems," *Electronics*, vol. 9, no. 8, article no. 1192, 2020.

[11] A. Al-Abassi, H. Karimipour, A. Dehghantanha, R. M. Parizi, "An ensemble deep learning-based cyber-attack detection in industrial control system," *IEEE Access*, vol. 8, pp. 83965-83973, 2020.

[12] G. Raman MR, N. Somu, A. Mathur, "A multilayer perceptron model for anomaly detection in water treatment plants," *International Journal of Critical Infrastructure Protection*, vol. 31, article no. 100393, 2020.

[13] M. Kravchik, A. Shabtai, "Detecting Cyber Attacks in Industrial Control Systems Using Convolutional Neural Networks," Proceedings of CPS-SPC 18 Conference, pp. 72-83, Toronto, Canada, Oct. 2018.

[14] D. Nedeljkovic, Z. Jakovljevic, "CNN based method for the development of cyber-attacks detection algorithms in industrial control systems," *Computers & Security*, vol. 114, article no. 102585, 2022.

[15] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, Y. Bengio, "Generative adversarial nets," *Advances in neural information processing systems*, vol. 27, 2014.

[16] J. Brownlee, *Generative Adversarial Networks with Python: Deep Learning Generative Models for Image Synthesis and Image Translation*. Machine Learning Mastery, 2019.

[17] F. Chollet, *Deep learning with python*. Manning, Shelter Island, NY, USA, Nov. 2017.

[18] NXP Semiconductors N.V. (2009, Feb.), "LPC1769/68/66/65/64/63 32-bit ARM Cortex-M3 microcontroller," [Online]. Available: https://www.nxp.com/docs/en/data-sheet/LPC1769_68_67_66_65_64_63.pdf, Accessed on: Mar. 2022.

[19] Microchip Technology Inc. (2008) "MRF24J40MA 2.4 GHz IEEE Std. 802.15.4TM RF Transceiver Module," [Online]. Available: http://ww1.microchip.com/downloads/en/DeviceDoc/70329b.pdf, Accessed on: Mar. 2022.

 ISBN 978-86-7466-930-3