Advances in Intelligent Systems Research, volume 174
Proceedings of the 8th Scientific Conference on Information Technologies for Intelligent Decision Making Support (ITIDS 2020)

Cyber-Physical Systems and Reliability Issues

Nafisa Yusupova
Department «Computational Mathematics and Cybernetics»
Ufa State Aviation Technical University
Ufa, Russia
yussupova@ugatu.ac.ru

Dmitry Rizvanov*
Department «Computational Mathematics and Cybernetics»
Ufa State Aviation Technical University
Ufa, Russia
ridmi@mail.ru

Dmitry Andrushko
Department «Computational Mathematics and Cybernetics»
Ufa State Aviation Technical University
Ufa, Russia
andrewrush@mpo14.ru

***Abstract*—Anomaly detection is a well researched concept used in many areas, including engineering systems design, where it helps detect errors and prevent failures. Traditional anomaly detection methods, based either on comparing the behavior of the real system with its model or on different signal processing methods, have been successfully applied for Fault Detection and Isolation (FDI) in mechatronic systems. Cyber-Physical Systems (CPS) are complex in both structural and behavioral terms. They consist of numerous heterogeneous components that generate large volumes of data, exchange information and form extremely complex patterns of behaviour. This makes it almost impossible to effectively set up and apply classical reliability assessment methods. The article discusses the basic models of machine learning and the possibility of their application to solve the problem of CPS reliability. It is proposed to use neural networks with long short-term memory (LSTM) to detect anomalies in the CPS. The neural network architecture with 3 hidden, input and output layers was designed. The experiment with testing data (Tennessee Eastman Process dataset) was conducted and analysis of the results was carried out.**

Keywords—cyber-physical systems, internet of things, classification, time-series, machine learning

I. INTRODUCTION

The concept of detecting anomalies is widely used in many areas, including the development of complex systems, because it allows you to detect errors at an early stage in the life cycle. Based on information about the current and previous state of the system, you can assess whether a failure will occur in the future. Traditional methods use modeling and mathematical apparatus of the signal processing theory [1].

One of the features cyberphysical systems (CPS) is the complexity of the structure and behavior of its individual elements. They consist of many heterogeneous components that generate a large amount of data and are difficult to interact with each other. With the development of CPS, the application of traditional methods for detecting anomalies is becoming more and more laborious.

Machine learning methods, in particular neural networks, are possible tool for solving the detecting anomalies problem. Neural network models are well suited for unstructured or weakly structured data [2], which in the long run allows the detection of anomalies in real-time.

II. STATE OF THE ART

The cyber-physical system in the general case is a combination of heterogeneous components that can interact with each other, self-organize and make autonomous decisions without human intervention. CPS is an integration of computing, network and physical processes. The main task in the design of CPS is the coordination of computational and physical components with each other [3].

CPS issues also affect related information technologies: management systems, data analysis, design, the Internet of things (IoT) and network connectivity. The development of CPS stimulates innovation in a number of areas with high reliability requirements: aeronautics, energy, healthcare, transport and others [4-7].

An example of cyber-physical systems is the automotive industry, which tends to appear autonomous cars that can interact with each other. Autonomy implies the ability to automate driver functions (self-parking and lane control), as well as fully independent driving. Interaction refers to the use of technologies to communicate with each other, the surrounding infrastructure (including via the Internet), on-board equipment, telecommunications (satellite monitoring of vehicles). An important part of CPS is the analysis of data obtained to optimize infrastructure, for example, smart traffic lights and traffic cameras - a special case of CPS.

The principles of CPS follow from the concept of inter-machine interaction (M2M), which is based on the idea of an autonomous device that interacts directly with another autonomous device. In this case, autonomy is understood as the ability of a node to generate and transmit information to another node without human intervention [8]. The form of information exchange depends on the specific implementation of the system. It is quite possible that a particular M2M device does not use widespread services and network topologies. This excludes Internet devices that are used as cloud and network storage. The M2M system can exchange data over channels that are not based on the IP protocol, for example, via a serial port or an arbitrary protocol.

The main components of CPS are sensors, computing and network equipment, software, actuators, and physical parts. Each component is susceptible to various types of malfunctions.

Fault (fault) - an abnormal condition or defect that can manifest itself as an error (error), which propagates through the system and ultimately leads to its failure (failure) [9].

For example, a sensor malfunction (sensor error) leads to incorrect measurement values (data error) that propagate over the network to the controller, which affects the calculation of the drive setpoint (another data error), and ultimately results in the system being unable to perform the required functions, i.e. failure [6, 10].

Random and systematic malfunctions are distinguished. Accidental malfunctions occur at any time, usually due to physical damage, wear. Systematic failures are due to design errors.

According to GOST R IEC 61508-2-2012 [11], which is identical to the international standard IEC 61508 [12], hardware components can have both random and systematic malfunctions. Programs are subject only to systematic malfunctions, because they are not prone to wear over time and usually occur due to an incorrect sequence of input data.

By duration, hardware faults are divided into short-term, intermittent, and permanent. Faults of the first type appear for a short time, intermittent faults tend to disappear and reappear, permanent faults remain in the system until they are eliminated.

Sensors and sensor networks play a crucial role in CPS: they provide information about the actual physical state of the system and the environment.

For example, in an autonomous vehicle, information coming from odometry, inertial and ultrasonic sensors, as well as from more complex lidars, radars and navigation cameras, provide the necessary input data for control, navigation and positioning algorithms.

For sensors, the following measurement errors are characteristic:

- drift: the sensor output continues to increase or decrease (usually linearly);
- offset: the output signal has a (constant) difference from the correct value, which is also called offset;
- noise: the variance of the output signal of the sensor increases significantly compared to the normal value;
- freezing: the output is "stuck" at a fixed value, also called a "jam".

Computers, controllers, and embedded cards are vulnerable to failure as a result of a single event (Single Event Upset, SEU), that is, an unintentional change in the state of a bit, which can occur even due to a single ionizing particle due to exposure to electromagnetic fields or radiation. The likelihood of such a malfunction increases due to a decrease in the technical process of integrated circuit elements.

Bit-flips is a special case of short-term errors that can occur in systems with high reliability requirements, for example, in the space industry or the automotive industry.

Inverting bits can affect the values in RAM memory cells and processor registers. The RAM stores the instructions and data of the entire program, and the processor registers are temporary storages in which the data currently being processed is stored. In other words, processor registers load data from memory and the processor processes it.

In microprocessors, inverting discharges can lead to synchronization errors, command flow, and data [13]. Data errors are associated with incorrect variable values as a result of inverting a bit that affects a cell or register. They are more common than synchronization or command flow errors.

Failure models are an abstraction of real physical defects in computing. An example is a constant fault (stuck-at) and the random inversion of one or more bits. In the first case, the bit is constantly fixed either at zero (stuck-at-zero) or at one (stuck-at-one). In the second case, a short-term change in the state of the bit occurs, which can be changed later.

Classic failure models are not enough to analyze all physical and information processes, so with the development of modern technologies there is a need for new models of faults: pulses in the transient process, uncertainty, delays, cable break, short circuit, open circuit, contact closure, etc.

The complexity and heterogeneity of CPS can lead to errors when transmitting data over the network: packet loss, delays. The first error occurs if the packet does not reach the receiver. In the event of a delay, the packet does not reach the receiver within the maximum allowed time interval.

Hence the main practical problems of the CPS:

- errors in data transmission over a wireless network;
- data processing errors;
- anomalies;
- noisy data;
- data loss;
- uncertainty;
- data variance;
- communication delays;
- time synchronization errors and others.

III. OVERVIEW OF EXISTING ANOMALY DETECTION METHODS

Anomalies are patterns in data that do not correspond to a specific concept of normal behavior. Data and synchronization errors in CPS signals are anomalies. Distinguish between point, context and collective anomalies [14, 15]. A point anomaly is a single instance of data that can be considered abnormal with respect to other data, for example, an extremely high or low sensor value compared to other values. This is the simplest type of anomaly and the most popular among anomaly detection methods. A contextual anomaly is an instance of data that is abnormal in certain conditions (context). The concept of context is determined by the structure of the data set. For example, sensor measurements may be accompanied by some metadata, such as geotags and timestamps. A data instance is an anomaly in this context, but an identical data instance can be considered normal in another context. In any time series data, time is a context attribute. Collective Anomaly: A collection of related data instances is abnormal to the entire dataset. Individual instances of data in a collective anomaly cannot be anomalies per se, but their combined occurrence in the aggregate is anomalous.

In recent years, deep learning has become one of the most popular machine learning techniques with unprecedented results in various fields of application. As part of a wider family of machine learning, deep learning can provide good performance and flexibility, using a multi-layer system of non-linear filters to extract traits with transformations. Deep learning is superior to traditional machine learning with increasing data volume [16, 17]. The main reason for the success of deep learning is its ability to receive high-level ideas that are relevant to the task. These representations are automatically extracted from the source data without the need to manually determine the characteristics and use the experience of subject matter experts.

Since traditional algorithms are not able to process complex data structures, their efficiency in detecting anomalies is not optimal both in inconsistent (for example, images) and sequential data sets. In addition, as the amount of data increases, traditional methods become almost impossible to use with the high level of data granularity needed to detect anomalies. Methods for the deep detection of anomalies (approx. Literal translation, did not find analogues in the domestic literature) (DAD) are necessary for large-scale detection of anomalies. DAD methods extract significant hierarchical characteristics from the source data. The ability to automatically extract features eliminates the need to manually set them by subject matter experts, so it is used for end-to-end processing of a raw data set, for example, in text and speech recognition tasks.

Deep learning anomaly detection algorithms are becoming more popular and are used to solve a wide range of tasks, for example, fraud detection, anomaly detection in medicine, sensor networks and the Internet of things. Studies have shown that deep learning completely surpasses traditional methods in the problems of detecting anomalies as the data scale grows [18]. A detailed review of deep learning methods for detecting anomalies is given in [19]. The problem of detecting anomalies for both traditional algorithms and algorithms based on deep learning is that the boundary between normal and abnormal (erroneous) behavior is often blurred and constantly changing.

There are some challenges to using deep learning models to detect anomalies in time series. First, we lack a well-defined picture indicating the occurrence of an anomaly. Secondly, noise in the input data seriously affects the performance of the algorithms. Third, as the length of the time series data increases, computational complexity also increases. Fourth, time series data are usually unsteady, non-linear and dynamically changing.

Recurrent Neural Networks (RNNs) can use the internal state (memory) to process input sequences. They are able to demonstrate dynamic behavior over time and retrieve features in the time sequence of data. Which makes it possible to use to detect anomalies in the data of time series. However, the disadvantage of RNNs is that they cannot distinguish between long-term dependencies with increasing time interval. To solve this problem, neural networks with long short-term memory (LSTM) were proposed, which are a special type of RNN consisting of a memory cell that stores information about previous periods of time. LSTM is a deep learning system that avoids the problem of a fading (or explosive) gradient. LSTM prevents the disappearance or explosion of backward propagating errors. As a result, LSTM can solve problems that require taking into account all the events that occurred over thousands or even millions of time intervals. For serial and temporal data, LSTM and RNN have become the preferred deep learning models because of the ability to distinguish between long-term dependencies.

Classification-based approaches use a neural network to recognize the normal and erroneous conditions of a system. This approach requires training with the teacher, a sufficient number of marked copies of the data, both normal and erroneous.

In prediction-based approaches, the current and previous values are used to predict the next several steps of the time-series. The following real signal value is compared with the predicted values for error detection. This method is widely used when erroneous instances are difficult to obtain, provided that the normal time series can be predicted for a certain number of steps forward. Unlike the first approach, it even makes it possible to mitigate temporary errors by replacing real erroneous values with predicted ones.

IV. STATEMENT OF THE RESEARCH PROBLEM AND DESCRIPTION OF THE INPUT DATA

The problem of detecting anomalies in CPS systems can be formulated as a multiple classification problem. Let a multidimensional time series from a labeled training sample be given. A multidimensional time series consists of several variables (features), for example, an accelerometer that provides three-dimensional data in each unit of time for each of the three axes

It is required to develop a classifier that can recognize the type of anomalies, that is, determine whether the current element of the control sample belongs to a specific class of anomalies.

As anomalies, the article considers signal errors in the CPS, which are time series data recorded continuously in time.

The choice of neural network architecture in the methods of deep detection of anomalies primarily depends on the nature of the input data. Input data can be classified as sequential (e.g. sensor values) or inconsistent (e.g. images).

The work uses a data set obtained by simulating chemical processes in industry (Tennessee Eastman Process or TEP dataset sample) [20, 21]. The data set consists of four parts: training and testing for normal (fault-free) and abnormal (faulty) processes. Training sets contain 500 time measurements in 25 hours of simulation. Test set kits contain 960 time measurements over 48 hours of simulation.

Each tuple in the selection contains the following column variables:

- in the first column (faultNumber) types of errors are indicated, values vary from 0 to 20. A zero error number means no errors, values from 1 to 20 represent various types of errors in TEP.
- the second column (SimulationRun) indicates how many times the TEP was simulated to get all the data.

In the training and test samples, the number of runs varies from 1 to 500 for each type of malfunction.

- the third column (sample) shows how many times the TEP variables were changed in one simulation. The value varies from 1 to 500 for the training and from 1 to 960 for the test sample. TEP variables (columns 4 through 55) were generated every 3 minutes for 25 hours and 48 hours for training and test samples, respectively.
- columns 4-44 (from xmeas_1 to xmeas_41) contain the measured TEP values.
- columns 45-55 (from xmv_1 to xmv_11) contain the controlled values of TEP.

V. THE EXPERIMENT AND ANALYSIS OF THE RESULTS

For the experiment, the following methodology was used:

1. Formation of validation, training and test samples.

2. Pre-processing of the obtained samples, including normalization.

3. Definition of training parameters.

4. Neural network training on a training set.

5. Verification of the results by training the neural network on a test sample.

6. Analysis and interpretation of the results.

Validation data was extracted from the training set to validate it. Validation data make up 20% of the training sample. The use of validation data allows us to assess the conformity of the model and the training sample, which is necessary when selecting model hyperparameters.

The obtained data set, consisting the validation data, training and test samples, contains changes of 52 signals during 500 identical time intervals. Each signal needs to determine the correct type of error, which is a classification task.

Before training the LTSM network, it is necessary to pre-process the data so that it is stored in an array of matrices, where each element is a set of 52 signals in one simulation. Each matrix stores a set of signals for each TEP simulation and can be either a normal or an anomalous process. Each set of signals indicates a specific type of error in the range from 0 to 20.

Also, during the preprocessing, data normalization was performed. Normalization is a method that scales the numerical values in a dataset to a common scale without distorting the differences in the range of values. This method is used so that a variable with an anomalous value does not affect the final result of the training. It also converts numerical values in a higher interval to a smaller interval (usually from –1 to 1) without losing any relevant information needed for training.

The resulting samples contain 400 error-free (fault-free) and 6800 faulty (faulty) simulations. The training fault-tree (Fig. 1) and faulty (Fig. 2) samples are shown below. For clarity, only 10 out of 52 signals are displayed.

The input layer of the neural network sequence Input Layer is the same as the number of signals (52). The network has three hidden layers with dimensions 52, 52 and 26, respectively.

To solve the problem of retraining, the dropout method between the layers of the LTSM network was used. A fully-connected layer was also added to solve classification problems, the dimension of which coincides with the number of output classes (18). The softmax layer added to the end of the neural network determines the probability of belonging to a particular class.

The training took place in the Matlab R2019b Trial mathematical modeling package (with the Deep Learning Toolbox and Parallel Computing Toolbox expansion packs) using the NVIDIA GeForce MX150 GPU, which supports CUDA hardware acceleration. Since this example processes a large amount of data, the use of a GPU significantly speeds up the learning time. Below are the learning outcomes.

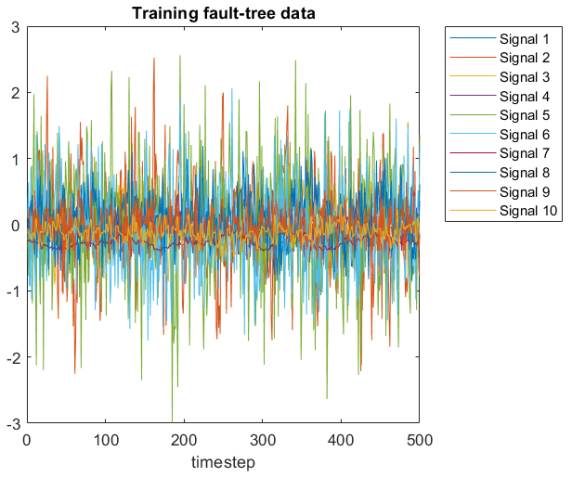

Fig. 1. Training fault-tree sample

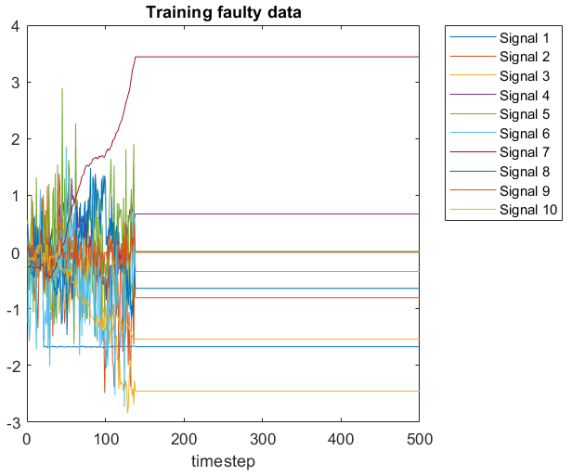

Fig. 2. Training faulty sample

Based on the results of checking the neural network with a test sample, its accuracy was determined - the number of matches of the classification results with the actual values of the types of faults divided by the total size of the test sample

(0.9974). High accuracy shows that the neural network has successfully classified most of the elements of the test sample.

The matrix of inaccuracies in (Fig. 3) shows the effectiveness of the classification. It has numerical values mainly on the main diagonal. A trained network is effective and correctly classifies more than 99% of signals.

VI. Conclusion

With the development of cyberphysical systems, the problem of ensuring reliability becomes more and more urgent, since the components of cyberphysical systems are subject to various kinds of anomalies.

As the amount of data increases, traditional methods become almost impossible to use with the high level of data granularity needed to detect anomalies. Machine learning methods are possible tools for detecting anomalies in CPS.

The article proposes the architecture of the LTSM network with 3 hidden, input and output layers, which allows us to classify anomalies in the functioning of the components of the cyberphysical system. The results of the experiment showed an anomalies classification accuracy at 99.74%.

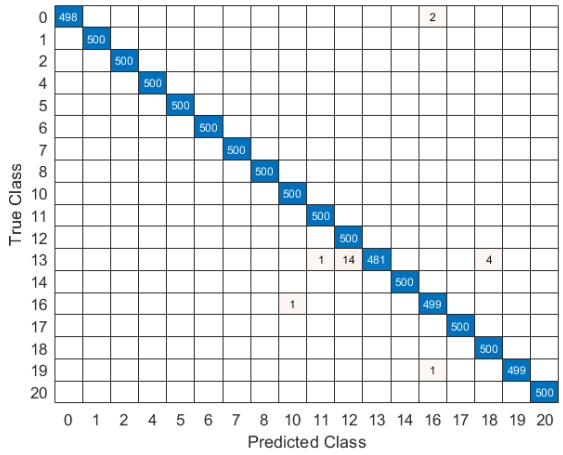

Fig. 3. Inaccuracy matrix

Acknowledgement

The research is partially supported by grants RFBR 18-07-00193-a and 19-07-00895-a. The results were obtained within the framework of project № FEUE-2020-0007 according to the state assignment.

References

[1] Dorokhov A. N. Ensuring the reliability of complex technical systems. SPb .: Doe, 2011 .-- 349 p. (in Russian)

[2] Yusupova N. I. Processing of poorly structured information based on artificial intelligence methods [monograph] / N. I. Yusupova, D. R. Bogdanova, M. V. Boyko; Ufa State Aviation Technical University (USATU). - Moscow: Innovative Engineering, 2016. (in Russian).

[3] National Science Foundation. Cyber-Physical Systems (CPS). [Electronic resource] // URL: https://www.nsf.gov/funding/pgm_summ.jsp?pims_id=503286 (accessed: 12/27/2019).

[4] Shakhmametova, Gouzel R .; Yusupova, Nafisa I. Intelligent Technologies Integration in the Task of Unaccented Trajectories Search in Robotics // 18th International-Federation-of-Automatic-Control (IFAC) Conference on Technology, Culture and International Stability (TECIS) Volume 51, Issue 30, 2018 , Pages 538-543.

[5] Yussupova, N., Rizvanov, D. Decision-Making Support in Resource Management in Manufacturing Scheduling // 18th International-Federation-of-Automatic-Control (IFAC) Conference on Technology, Culture and International Stability (TECIS) Volume 51, Issue 30, 2018, Pages 544-547.

[6] Mutzke, T., Ding, K., Morozov, A., Janschek, K., Braun, J. Model-based Analysis of Timing Errors for Reliable Design of Mechatronic Medical Devices, Proceedings of 3rd International Conference on Control and Fault -Tolerant Systems, Barcelona, Catalonia, 2016.

[7] Gabdulkhakova, A. An agent-based solution to the resource allocation problem in emergency situations / A. Gabdulkhakova, B. König-Ries, D. Rizvanov // Proc. 9th IEEE European Conf. on Web Services (ECOWS 2011), 14–16 Sept. 2011, Lugano, Switzerland. Pp. 151-157.

[8] Kovács, G., Yussupova, N., Rizvanov, D. Resource management simulation using multi-agent approach and semantic constraints // Pollack Periodica Volume 12, Issue 1, 2017, Pages 45-58.

[9] Avizienis Algirdas. Dependability and Its Threats: A Taxonomy / Algirdas Avižienis, Jean-Claude Laprie, Brian Randell, Building the Information Society, Springer, Boston US, 2004.

[10] Fabarisov, N. Yusupova, K. Ding, A. Morozov, K. Janschek. The Efficiency Comparison of The Prism and Storm Probabilistic Model Checkers for Error Propagation Analysis Tasks / INTERNATIONAL SCIENTIFIC JOURNAL "INDUSTRY 4.0", Vol. 3 (2018), Issue 5, pg (s) 229-231.

[11] GOST R IEC 61508-2-2012. Functional safety of electrical, electronic, programmable electronic safety related systems. Part 2. System requirements. (in Russian).

[12] IEC 61508. Functional Safety of Electrical / Electronic / Programmable Electronic Safety-related Systems.

[13] Fabarisov T., Yussupova N., Ding K., Morozov A., Janschek K .: Analytical Toolset for Model-based Stochastic Error Propagation Analysis: Extension and Optimization Towards Industrial Requirements. Proceedings of the 19th international workshop on computer science and information technologies, Germany, Baden-Baden, 2017, Volume 1, pp. 66-70.

[14] V. Chandola, A. Banerjee, and V. Kumar. Anomaly detection: A survey. ACM computing surveys (CSUR), 41 (3): 15, 2009.

[15] V. Chandola, A. Banerjee, and V. Kumar. Outlier detection: A survey. ACM Computing Surveys, 2007.

[16] A. C. Bahnsen. Building ai applications using deep learning. https://blog.easysol.net/ building-ai-applications /, 2016.

[17] H.-K. Peng and R. Marculescu. Multi-scale compositionality: identifying the compositional structures of social dynamics using deep learning. PloS one, 10 (4): e0118309, 2015.

[18] A. Javaid, Q. Niyaz, W. Sun, and M. Alam. A deep learning approach for network intrusion detection system. In Proceedings of the 9th EAI International Conference on Bio-inspired Information and Communications Technologies (formerly BIONETICS), pages 21–26. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2016.

[19] R. Chalapathy and S. Chawla. Deep learning for anomaly detection: A survey, 2019.

[20] Rieth, C. A., B. D. Amsel, R. Tran., And B. Maia. "Additional Tennessee Eastman Process Simulation Data for Anomaly Detection Evaluation." Harvard Dataverse, Version 1, 2017. https://doi.org/10.7910/DVN/6C3JR1.

[21] Heo, S., and J. H. Lee. "Fault Detection and Classification Using Artificial Neural Networks." Department of Chemical and Biomolecular Engineering, Korea Advanced Institute of Science and Technology.