Copyright © 2019 for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0)

Bio-inspired Approach to Self-Regulation for Industrial Dynamic Network Infrastructure

Daria Lavrova
Higher School of Cybersecurity and Information Security
Peter the Great St. Petersburg Polytechnic University
Saint-Petersburg, Russia
lavrova@ibks.spbstu.ru

Elizaveta Zaitceva
Higher School of Cybersecurity and Information Security
Peter the Great St. Petersburg Polytechnic University
Saint-Petersburg, Russia
eaz@ibks.spbstu.ru

Petr Zegzhda
Higher School of Cybersecurity and Information Security
Peter the Great St. Petersburg Polytechnic University
Saint-Petersburg, Russia
zeg@ibks.spbstu.ru

***Abstract* — A bio-inspired approach for self-regulation of modern industrial network infrastructure is proposed. The approach is based on the analogy between the target function of an industrial system and the DNA. The target function describes the set of functions necessary for the operation of industrial system, including a description of their relations and the order of execution. The proposed functional representation of the target function uniquely characterizes the industrial system, just as DNA characterizes the organism and contains biological information important for its building and maintaining. Then the task of restoring the target function after the attack is similar to the task of DNA sequencing. The key difference is that the functional representation of the target function is known in advance, and this greatly simplifies the task. Proposed approach using both self-regulation scenarios and mathematical apparatus of de Bruijn graphs and intersection graphs used in bioinformatics for DNA sequencing. The approach reduces the time for network self-regulation required when detecting security threats.**

Keywords—industrial network; self-regulation; de Bruijn graph; cyber threat; intersection graph; information security; cybersecurity; cyber-physical system.

CEUR-WS.org/Vol-2603/short8.pdf

I. INTRODUCTION

The development of industrial systems has led to a shift from automated production to the concept of Digital Production. In Digital Production systems, the information and physical components are closely interconnected; they work within a single industrial circuit [1-3].

At the same time, the problem of ensuring the security of industrial systems, the specificity of which is autonomy from humans, mutual components control of each other and their availability to communicate using the Internet, is becoming increasingly important [4]. This specificity opens up wide opportunities for remote destructive impact on the components of industrial systems, as a result of which physical processes can be disturbed. This can cause harm to the environment, life and human health. In such conditions, the task of preventing computer attacks on industrial systems is especially urgent, an important step in solving which is to counter attacks [5].

In this paper, we propose a bio-inspired approach for industrial network infrastructure self-regulation. The purpose of this approach is to exclude the conditions of successful cyber-attack implementation due to reconfiguration of the network structure.

II. RELATED WORKS

The study of the problem of developing approaches to automatic self-regulation of the network infrastructure of complex large-scale systems showed that many solutions are aimed at restoring the system after a failure (including that caused by a cyber-attack).

In [6], the authors propose an approach for automatic proactive response to cybersecurity incidents, which uses data obtained from open sources; analytical models of attacks, events, countermeasures and dependencies between services; hierarchical built-in set of heterogeneous security metrics. The choice of metrics is based on their effectiveness, defined as the difference in risk levels before and after the implementation of the metric. Risk, in turn, is determined by the presence of vulnerabilities in the system and the cost of resources. The main problem of this approach is the use of data presented in open sources. Thus, the method is focused on already known attacks and may prove to be powerless when attackers invent new methods of influencing the system.

A small number of studies are devoted to creating approaches that allow not only detecting cyber-attacks, but also counteracting them by neutralizing them or correcting system behavior [7-11]. The study [7] is especially noted, in which the authors propose the correction of destructive effects based on the control method, using the Lyapunov model. This method is designed to ensure system stability. To identify potential computer attacks, the authors use machine learning methods that are widely used for clustering and regression, as well as

The study was carried out as part of the scholarship of the President of the Russian Federation to young scientists and graduate students SP- 1932.2019.5.

methods based on neural networks. The use of control based on the Lyapunov model allows a neutralizing effect on the subsystem when a cyber-attack is detected. The disadvantage of this method as applied to its integration with PS is the rather high complexity of mathematical transformations and calculations, in particular, the construction of the Lyapunov model. We should also highlight the works devoted to the implementation of the bio-inspired homeostatic concept [12,13]. This concept implies maintaining a constant state of the environment under the conditions of destructive influences, provided by the implementation of the homeostatic control loop, which carries out automatic self-regulation of the system. The approach [13] can be used for self-regulation of the network infrastructure of small-scale industrial systems. This approach is based on the development of self-regulation scenarios and the application of a specific scenario, depending on what destructive effect the system has on the system. Thus, we can conclude that most approaches to self-regulation of complex systems (including industrial ones) are based on the use of behavior models and self-regulation patterns. These approaches, on the one hand, will be effective in case of system failures, but on the other hand, it will not It will be effective in case of massive cyberattacks that immediately affect a large number of system components.

The approach proposed in this paper is aimed at self-regulation of industrial systems with a flexible dynamic network infrastructure. It provides a system response both to the negative impact caused by cyberattacks and to failures caused by technical malfunctions. The approach uses a graph representation of the network infrastructure and a functional representation of the target function.

III. Modeling of Network Infrastructure and Target Function

Network infrastructure of industrial system is represented as an oriented graph (Fig. 1). The set of graph vertices V characterizes all components of the industrial system that are capable of network interaction. The set of arcs E reflects all possible inter-component connections, which manifest as data exchange.

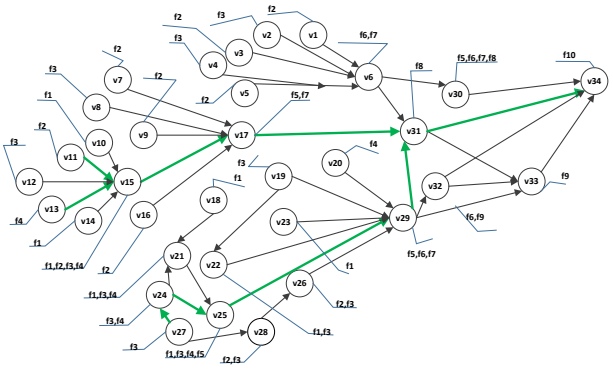

Fig. 1. Example of a graph representation for network infrastructure and target function (bold line).

Each component of the system modeled by the vertex v_i is characterized by a set of functions that it is able to implement. Each arc e_{ij} of the graph corresponds to a certain characteristic ω, which may have different meanings depending on the type of industrial system (channel bandwidth, data transfer rate, etc.).

The target function of an industrial system is presented simultaneously as a set of routes on a graph (where each route characterizes a specific process performed by the system) and as a set of functional sequences: $F=\{F_1, F_2, \ldots, F_n\}$, where each system process corresponds to a functional sequence.

The following types of relationships between functions are introduced into the model:

- Sequentially performed functions - one function uses the results of others, it is not required to perform this function immediately after the completion of the previous ones. Parentheses are used to indicate this relationship: $f_j(f_k)$ - first, the function f_k is executed, then the function f_j is applied to its result.
- Strictly sequentially performed functions - one function uses the results of others, it is required to perform this function immediately after the previous ones are completed. To indicate such a relationship, square brackets are used: $f_j[f_k]$ - the function f_k is first executed, then, immediately after its completion, the function f_j is applied to its result.
- Parallel functions - must be performed simultaneously. To indicate such a relationship, the sign * is used, the operation is commutative: $f_j * f_k$.
- Functions performed in random order. To indicate such a relationship, the + sign is used, the operation is commutative: $f_j + f_k + f_m$.

Then {} is used as an analogue of ordinary brackets.

For functions, a decomposition operation can be performed that implements the representation of some complex function f_i as a sequence of several, more computationally simple, functions.

IV. Analogy Between Target Function and DNA

Both target function of an industrial system and DNA can be represented as a sequence of elements: in the case of an target function, the elements are functions, and in the case of DNA, nucleotides. Under cyber-attacks, one or more components can lose their ability to perform certain functions. In this case, the target function “breaks up” into separate parts, which will need to be connected together again. To do this, it is required to reconfigure the network infrastructure. To restore the functional sequence that determines the target function, it is proposed to apply a mathematical apparatus used in DNA sequencing - the search for overlaps and matching parts of the sequence.

A DNA chain is a sequence of four types of nucleotides: A (adenine), T (thymine), G (guanine), C (cytosine). Modern

technologies allow reading reads - sequences of several hundred nucleotides in length from random places; the key step is to combine reads based on their overlapping sections. For this, special programs are used, most of which use de Bruijn and overlap graphs [14]. The de Bruijn graph is a directed graph whose vertices are rows of length *(k − 1)*, and the edges are rows of length *k*. The overlap graph is a weighted oriented graph whose vertices are rows (which can be of different lengths). An edge between a pair of vertices is drawn if the corresponding lines overlap.

Examples for target function which could be constructed using by both graphs are presented at Fig. 2, 3. Let the target function be represented by the following sequence: $F=f_1f_3f_4f_5f_8f_{11}$.

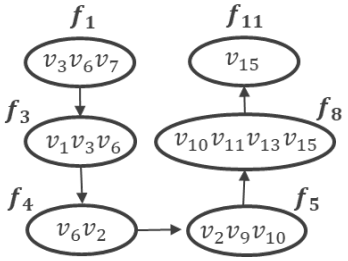

Fig. 2. Target function representation using de Bruijn graph.

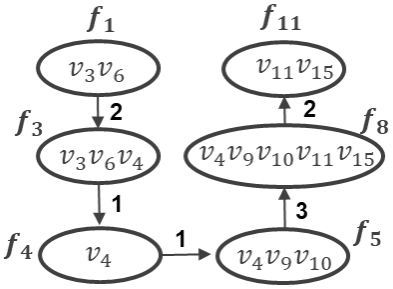

Fig. 3. Target function representation using overlap graph.

Thus, the main differences between the graph de Bruijn and the overlap graph in terms of graph structure are as follows:

- When searching for overlappings in the de Bruijn graph, a fixed length of the prefix and suffix equal to *k-1* is used.
- De Bruijn graph is characterized by a fixed length of lines characterizing each vertex (length *k*).

Also proposed an analogy between information functions of industrial systems and DNA fragments (Table 1).

TABLE I. DNA AND TARGET FUNCTION

Construction	DNA and target function	
	DNA	***Target function***
Read	DNA fragment of small length	The function of an industrial system with low computational complexity
Contig	DNA fragment of medium length	The computationally complex function of an industrial system
Scaffold	Orderly set of contigs	A chain of computationally complex functions interconnected in the target function

To implement the method, for each function that is part of the target function, is constructed a set of vertices that are able to implement this function. Thus, for each function, a cluster of devices is formed that can replace each other in the event of failure or compromise (Fig. 4).

The proposed approach to self-regulation also uses pre-formed self-regulation scenarios based on representing the network infrastructure of the industrial system in the form of a graph. Scenarios contain rules for replacing vertices and arcs in a graph, as well as restrictions on the construction of new routes that reflect the performance of the target function. In particular, such routes should not include compromised system components or components with suspicious behavior.

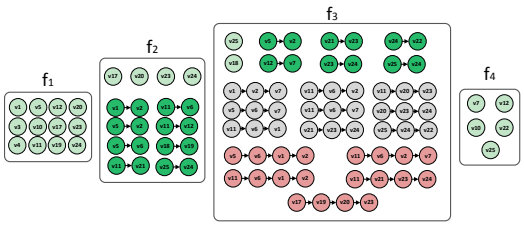

Fig. 4. Cluster of devices for each function of target function

The method is performed in three main steps:

- Updating data on the structure of the system in accordance with the identified anomalies and threats to cybersecurity. At this step, the removal of incorrectly functioning vertices from the clusters and their relationships is performed.
- Automatic selection of the method of self-regulation (use of scripts, de Bruijn graphs or overlap graphs).
- Application of the selected method of self-regulation, as a result of which the restoration of the target function is performed.

TABLE II. SELF-REGULATION APPROACHES

Type of security breach	Target function recovery
Violation of one read function or one contig function	Using self-regulation scenarious
Violation of several read functions that are not interconnected, several contig functions that are not interconnected, or several functions (which include both read functions and contig functions) that are not interconnected.	Using only self-regulation scenarious or only de Bruijn graphs and overlap graphs

Type of security breach	Target function recovery
Violation of several functions (both reads and contigs) related to each other or a combination of various types of violations	Using self-regulation scenarious, de Bruijn graphs and overlap graphs

To confirm the assumption described in Table 2, an experiment was conducted consisting in simulating cyber-attacks and estimating the time of self-regulation, performed in different ways.

The Fig. 5 shows the results of an experiment in which 3 interconnected vertices were removed from the graph characterizing the industrial network. Average time for self-regulation using scenarios was 0.02 seconds, for overlap graphs – 0.003 seconds.

The Fig. 6 shows the results of an experiment in which one vertex was deleted, then 3 interconnected vertices in another part of the graph were deleted and 2 successive vertices were compromised. The time spent on self-regulation turned out to be approximately the same for both approaches (0.004 seconds for scenarios and 0.003 seconds for complex approach), however, the use of an integrated approach gave a greater gain in time.

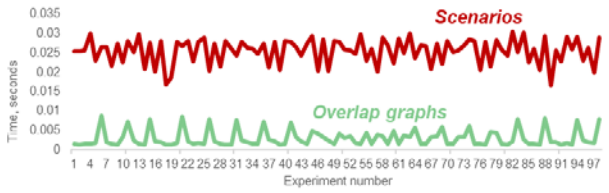

Fig. 5. Results of the first experiment.

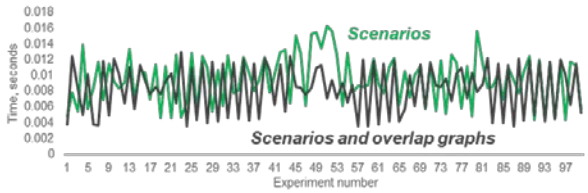

Fig. 6. Results of the second experiment.

The results of the experiments confirmed the assumption that when removing interconnected vertices, de Bruijn graphs and overlap graphs should be used, and for massive attacks involving vertices in different parts of the graph, an integrated approach should be used.

V. Experimental Results

To conduct experimental research, a bench simulating an automatic industrial water treatment system was used [15]. Water treatment includes six subprocesses P1-P6 (Fig. 7):

- P1: collection and storage of untreated water.
- P2: pre-treatment of water.
- P3: primary water treatment involving the use of ultrafiltration and backwash technologies.
- P4: secondary treatment (dechlorination).
- P5: reverse osmosis.
- P6: transfer of purified water, backwash and purification.

Target function of a part of water treatment system: $F=\{\{\{f_8(f_6(f_7(f_4)))* f_5*f_2(f_7)\}(f_3(f_3(f_1))))\}* f_4 \}(f_3(f_2(f_1)))$, where:

- f_1 - control of water circulation systems.
- f_2 - parameter adjustment.
- f_3 - access to the database.
- f_4 - command sending and reconfiguration.
- f_5 - adjusting the volume of water in the tanks.
- f_6 - sending a command about moving water.
- f_7 - change system parameters.
- f_8 - resetting the tank.

The following types of destructive influences were modeled: denial of service attacks on the components of the experimental bench, attacks consisting in modifying data from the components of the experimental bench, illegitimate changes to the structure of the experimental bench. Consider one of the denial of service attacks, as a result of which three interconnected system components fail (in terms of the graph model, three vertices of the graph are removed).

Fig. 8 shows the changes in the target function of the system, as a result of which the functions f_1, f_3 and f_4 ceased to be executed in the target function.

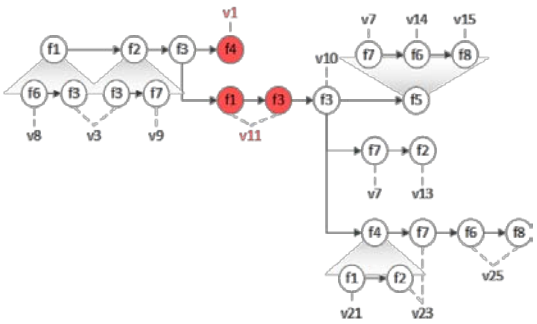

Fig. 8. Changes in target function during cyber attack.

For system self-regulation, de Bruijn graphs and known function decompositions were used:

- $f_1=f_3(f_6)$.
- $f_2=f_7(f_3)$.
- $f_5=f_7(f_6(f_8))$.

- $f_4=f_2(f_1)$.

The graph obtained as a result of self-regulation is shown in Fig. 9.

Black edges are edges constructed according to the principles of constructing an overlap graph. The dashed edges are based on the neighborhood of the vertices in the original graph. The peaks reachable from the starting peak are marked in gray. The vertices from which the final vertex is reachable are marked in black.

For the considered attack, the time of self-regulation was 25% of the time of its implementation. Thus, for this cyber-attack, it was possible to exclude the conditions for its implementation. The restored target function is shown in Fig. 10.

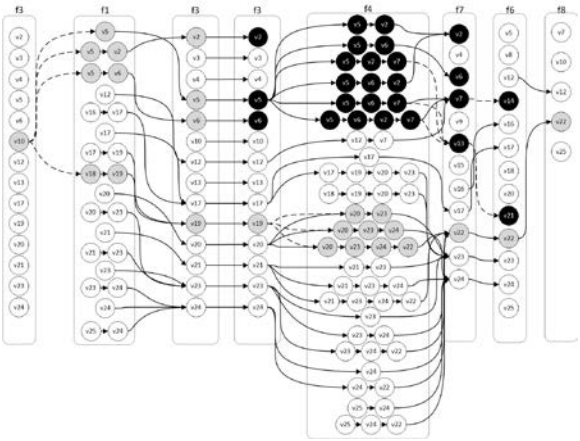

Fig. 9. Graph obtained as a result of self-regulation.

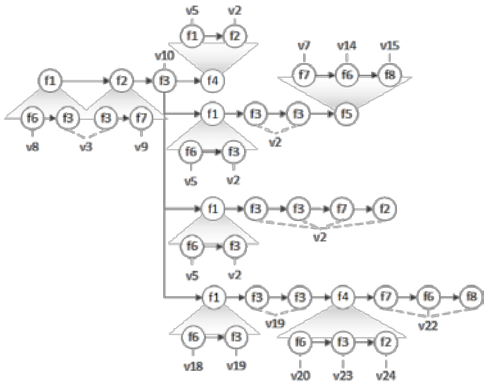

Fig. 10. The restored target function.

For other attacks, the system's self-regulation time was 12-38% of their implementation time.

Conclusion

The specifics of systems that combine information technology and physical process control devices have made insufficient the use of classical methods of ensuring information security. The work is devoted to a detailed consideration of one of the stages of preventing cyber-attacks on industrial systems, which consists in the self-regulation of their network structure.

In this paper, the similarity between target function restoring and DNA sequencing is investigated. It is proposed to use the principles of constructing de Bruin graphs and overlap graphs to restore system performance. The use of such graphs will reduce assembly time due to faster "coupling" of the restored sections of the target function.

This approach was proposed for the first time, an analysis of related works showed that most studies use behavioral models and pre-formed self-regulation patterns for automatic self-regulation. The combined use of de Bruijn graphs and overlap graphs together with self-regulation scenarios will effectively resist massive cyber-attacks, which is confirmed by the results of experimental studies.

References

[1] M.J. de C Henshaw, "Systems of systems, cyber-physical systems, the internet-of-things... whatever next?," Insight, vol. 19, num. 3, pp. 51-54 2. 2016. DOI: 10.1002/inst.12109.

[2] E. Pavlenko, D. Zegzhda, "Sustainability of cyber-physical systems in the context of targeted destructive influences," IEEE Industrial Cyber-Physical Systems (ICPS), pp. 830-834, May 2018. DOI: 10.1109/ICPHYS.2018.8390814.

[3] N. Sadiku, Y. Wang, S. Cui, M. Musa, "Cyber-physical systems: a literature review," European Scientific Journal, vol. 13, No. 36, pp. 52-58, 2017. DOI: 10.1142/S2424862217500129.

[4] O. Givehchi, K. Landsdorf, P. Simoens, A. W. Colombo, "Interoperability for industrial cyber-physical systems: An approach for legacy systems," IEEE Transactions on Industrial Informatics, vol. 13, No, 6, pp. 3370-3378, 2017. DOI: 10.1109/TII.2017.2740434.

[5] D. Lavrova, M. Poltavtseva, A. Shtyrkina, "Security analysis of cyber-physical systems network infrastructure," IEEE Industrial Cyber-Physical Systems (ICPS), pp. 818-823, May 2018. DOI: 10.1109/ICPHYS.2018.8390812.

[6] E. Doynikova, I. Kotenko, "The Multi-Layer Graph Based Technique for Proactive Automatic Response Against Cyber Attacks," 26th Euromicro International Conference on Parallel, Distributed and Network-based Processing (PDP), pp. 470-477, 21 March 2018. DOI: 10.1109/PDP2018.2018.00081.

[7] Z. Wu, F. Albalawi, J. Zhang, Z. Zhang, H. Durand, P. Christofides, "Detecting and Handling Cyber-attacks in Model Predictive Control of Chemical Processes," Mathematics, vol. 6, No. 10, pp. 173, October 2018. DOI: 10.3390/math6100173.

[8] A. Markov, A. Barabanov, V. Tsirlov, "Information Security Controls against Cross-Site Request Forgery Attacks on Software Applications of Automated Systems," Journal of Physics: Conference Series, vol. 1015, No. 4, pp. 1-7, 2018. DOI: 10.1088/1742-6596/1015/4/042034.

[9] A. Markov, A. Dorofeev, V. Tsirlov, "Social media in identifying threats to ensure safe life in a modern city," International Conference on Digital Transformation and Global Society, pp. 441-449, 2016. DOI: 10.1007/978-3-319-49700-6_44.

[10] S. Chennareddy, A. Agrawal, A. Karuppiah, "Modular self-reconfigurable robotic systems: a survey on hardware architectures," Journal of Robotics, 2017. DOI: 10.1155/2017/5013532.

[11] I. Kotenko, I. Saenko, O. Lauta, "Analytical modeling and assessment of cyber resilience on the base of stochastic networks conversion," IEEE International Workshop on Resilient Networks Design and Modeling (RNDM), pp. 1-8, August 2018. DOI: 10.1109/RNDM.2018.8489830.

[12] I. Gerostathopoulos, D. Skoda, F. Plasil, T. Bures, A. Knauss, "Architectural homeostasis in self-adaptive software-intensive cyber-physical systems," European Conference on Software Architecture, pp. 113-128, November 2016. DOI: 10.1007/978-3-319-48992-6_8.

[13] D. P. Zegzhda, E. Y. Pavlenko, “Cyber-physical system homeostatic security management,” Automatic Control and Computer Sciences, vol. 8, pp. 805-816, 2017. DOI: 10.3103/S0146411617080260.

[14] A. Sergushichev, A. Aleksandrov, S. Kazakov, F. Tsarev, A. Shalyto, “Combining de Bruijn graphs, overlap graphs and microassembly for de novo genome assembly,” Izv. Saratov Univ. (N. S.), Ser. Math. Mech. Inform., vol. 13, iss. 2, pp. 51-57, 2013. DOI: 10.18500/1816-9791-2013-13-2-2-51-57.

[15] J. Goh, S. Adepu, K. Junejo, A. Mathur, “A dataset to support research in the design of secure water treatment systems,” International Conference on Critical Information Infrastructures Security, pp. 88-99, October 2016. DOI: 10.1007/978-3-319-71368-7_8.