Communication Papers of the Federated Conference on Computer Science and Information Systems pp. 35–42 DOI: 10.15439/2020F194 ISSN 2300-5963 ACSIS, Vol. 23

# Direct communication of edge elements in the Industrial Internet of Things

Piotr Hajder*, Mirosław Hajder†, Mateusz Liput† and Mariusz Nycz‡
*AGH University of Science and Technology, Krakow, Poland, email: phajder@agh.edu.pl
†University of Information Technology and Management, Rzeszow, Poland
email: {miroslaw.hajder,mateuszliput123}@gmail.com
‡Rzeszow University of Technology, Rzeszow, Poland, email: mnycz@prz.edu.pl

***Abstract*—In this paper the results concerning changes introduced into the architecture of connections of an industrial computer network built on the basis of solutions characteristic of the Industrial Internet of Things were described. We introduce the new solutions improving such properties of industrial networks as: reliability, information security and usage of computational resources available in edge elements. The basic goal of the research was ensuring continuous functioning of the industrial computer network. Isolation of critical elements of the technological process can have far-reaching consequences, including local threats to workers' health and life, and even global technological disasters. Continuity of operation as a consequence of the reliability of the components of an information system is one of its most desirable properties.**

**The offered solution is based on the use of multi-channel reconfigured optical bus that is activated when the basic communication channels fail. In addition, the use of the described solutions for the build of a parallel, heterogeneous computing system based on elements of the Arduino and Raspberry Pi and system servers has been described and evaluated.**

## I. Introduction

THE paper presents the results in the area of improving the effectiveness of Industrial Internet of Things (IIoT) used in the industrial information system (IIS). Presented results are based on a real system from medium and large industrial enterprises of the metallurgical, electromechanical and aviation industries.

During the research, our attention focused on improving three key IIoT properties for the enterprise: reliability along with derived characteristics, information security, and insufficiency of computational and memory resources. Information security is a guarantee of eliminating IIS unsanctioned access, use, disclosure, deformation, modification, investigation, recording or destruction of information [1], [2], [3], [4].

Reliability is a property of technical objects to maintain over time, within set limits, the values of all parameters necessary to perform the required functions in specific modes and conditions of use. The concept of dependability is associated with a number of other often cited system and network characteristics. Most often they are: fault tolerance; reliability; survivability, whose terms can be found in the literature [5], [6], [7]. They all indirectly determine the degree of continuity of IIS, key characteristics of IIS [8], [9]. Starting from the definition presented earlier, information security can be a measure of its insensitivity to cyber criminals' attacks, which may also result in a loss of acting continuity. Although IIS is not an attractive object for traditional cyber criminals, such attacks performed for political reasons will occur more often and their negative consequences will increase. The solutions proposed in work allow improve the level of security in different way than currently used. In critical situations, IIS should have at its disposal minimal, attack-resistant processing resources that allow it to safely monitor its operations in critical situations, or terminate technical systems that threaten security. Starting the research, it was assumed that IIS will be build on the basis of industrial Internet of Things, which edge elements was built on the basis of Arduino or Raspberry platform [1], [10]. Thus, these elements can be used to build a heterogeneous, parallel processing system, largely insensitive to external attacks. The presented work consists of three sections. Section II presents the essence of industrial information systems using IIoT, based on the available literature, special features were distinguished, and potential threats of this type of systems were discussed. Section III discusses the proposed solution for connecting IIS edge elements, stressing the benefits and potential dangers. Section IV describes a research experiment whose aim was to confirm the effectiveness of the proposed solutions. In *the Summary* the analysis of the results of theoretical and empirical research is presented and discusses the current and future uses of the designed architecture. The work ends with *the bibliography* containing the works of other authors as well as the authors' own publications.

## II. Industrial computer networks with IoT

### A. Evolution of industrial computer networks

In order to illustrate the evolution of interoperability of industrial network components, Fig. 1 shows its hierarchical model. The classic model presented there consist of the 4 layers, which the highest two use traditional communication channels based on Ethernet technology. Changes massively appear at the bottom-up automation level, at which interstitial communication takes place using industrial communication standards. We currently know over 50 types of industrial networks and data transmission protocols covered by the standard Fieldbus term [2], [11], [12]. Among of them the most known are: ProfiNet, HART, Modbus, Profibus, DeviceNet, CAN,

The work was realized as a part of fundamental research financed by the Ministry of Science and Higher Education, grant no. 16.16.110.663 - task 18

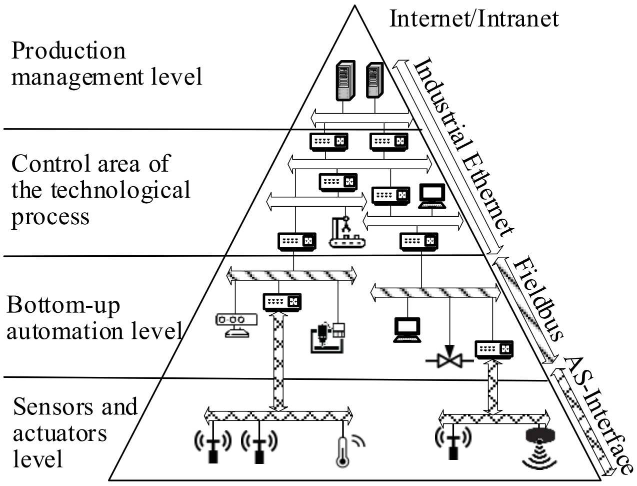
Internet/Intranet
Production management level
Control area of the technological process
Bottom-up automation level
Sensors and actuators level
Industrial Ethernet
Fieldbus
AS-Interface

Fig. 1. Hierarchical model of an industrial computer network

CANopen, Lon-Works, FoxCom, ControlNet, SDS, Seriplex, BACnet, FIP, ASI, Industrial Ether-net, Wordfip, Foundation Fieldbus, Inter-bus, BitBus and others [2], [11]. Only some of them are widely used. Although a special FDT (Field Device Tool) interface has been designed to ensure interoperability of devices, technologies, data transmission protocols of different standards, their versions, generations and manufacturers, the need for communication between various system components remains a problem. Changes in the manner of communicating elements of the industrial computer network should first cover the *bottom-up automation level* and concern the use of methods and means of communication characteristic of the IoT network in this layer. However, the deepest modifications will include *sensors and actuators level*, in which, in place of passive measuring sensors, IoT compliant devices will be used, with a wide range of communication options. One of the most significant disadvantages of industrial computer networks operating in accordance with the architecture of Fig. 1 are the difficulties in communication controlling at *sensors and actuators level*. By equipping the lowest level of the model with intelligent measuring sensors, based on IIoT solutions, this defect will be removed and the control will be carried out using sensors based on the Arduino and Raspberry platforms. The use of hardware and software traffic analyzers located in all network segments at risk of attacks, both with wired and wireless communication is the first stage of the proposed changes in the architecture of the industrial computer network in Fig 1. For security reasons, sensors should not be connected to any of the automation, electronics or industrial metrology components used in the system. Thus, the operation of the monitoring subsystem in any negative way does not translate into the operation of the enterprise's integrated information service system. Unfortunately, in a number of cases this solution can be troublesome or even impossible to implement. An acceptable solution is the use of a further reconfigured connection network. For better illustration of the changes made in the traditional industrial network architecture, the bottom two layers of the new network are presented in Fig. 2. Additional software and hardware traffic analyzers $A$ have been added to sensors and actuators level. They were made using Raspberry Pi or Arduino platforms equipped with software and additional communication interfaces. The main goal for analyzers is to detect traffic anomalies in tracked network segments. Depending on the interface, wired or wireless network segments can be analyzed using the following protocols: Ethernet, Fieldbus AS-Interface, both on sensors and actuators and bottom-up automation levels. The proposed solution based on autonomous traffic analyzers, besides its advantages (significant improvement of information security and availability of computational resources), in small installations is characterized by high implementation costs. In large enterprise networks, these costs (around 250 euro per analyzer) are negligible. Therefore, as a part of the research, the traffic analysis technology based on IIoT components, for which traffic monitoring is not their main function, has been positively verified. The introduction to the lowest level of the industrial network model of software and hardware traffic analyzers is the first of the modifications made in the classic industrial computer network. The second change applied involves the implementation of an additional multi-bus communication network used exclusively by traffic analyzers. Separating the bus from the rest of the network ensures its insensitivity to possible attacks. The bus is a communication element that can be used in critical moments, e.g. for emergency securing of technological operations. From the point of view of modern graph theory [13], [14], [15], [16], [17] and hierarchical systems [18], [19], the modified network is a hierarchical star whose leaves have been connected by means of an additional horizontal communication channel.

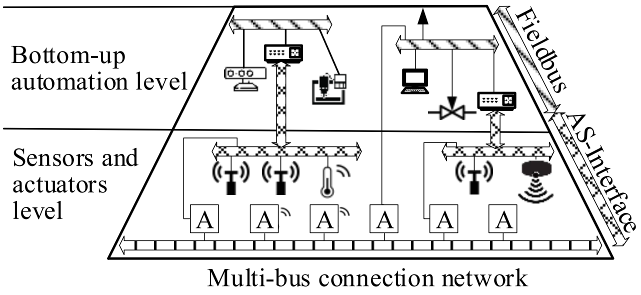
Bottom-up automation level
Sensors and actuators level
Fieldbus
AS-Interface
A
A
A
A
A
A
Multi-bus connection network

Fig. 2. Hierarchical model of the industrial network. A - software and hardware network analyzer

### B. Methods for describing bus systems

The most important feature of the communication bus (Fig. 2) is the possibility of distributed reconfiguration. This means that the selection of a specific communication channel will be made directly by traffic analyzers $A$. The organization of connections can be selected so that the edge communication parameters imposed by the designer, such as: maximum transmission delay, integration of only indicated network nodes, network life at the indicated level will be met at any time. This requires, among others mathematical description of bus network connections. Mathematical description of bus systems is presented in [9], [20], [21]. In order to improvements in the machine analysis and synthesis of the bus connection network, an algebraic topology notation based on an algebra of connected finite non-directed graphs has been proposed. Graph algebra is defined in the Definition 1.

**Definition 1.** *The pair $A = (D, \Omega)$ will be called the universal graph algebra over the universe $U$, if $D$ is a set of graphs with vertices from the set $U$, and the signature $\Omega$ enables the zero operation $\Lambda$, binary operations of adding a vertex, adding an edge, removing a vertex and removing edges.*

First, let's define the theorem, which can be used to describe the topology of selected networks.

**Theorem 1.** *The minimal elements of the $AG = (A, \Omega)$ algebra will be trees of the form $G_u^0$, where: $G_u^0 = (V = \{u\}, E = \emptyset)$ - an empty tree composed of the vertex u.*

Using the Theorem 1 we will analyze the trees:

1) $T_{u,v} = (V = \{u, v\}, E = \{(u, v)\})$;
2) $T_{u,v}^w = (V = \{u, v, w\}, E = \{(u, w), (w, v)\})$.

Using algebra, the above trees can be represented by the minimum elements:

1) $T_{u,v} = (V = \{u, v\}, E = \{(u, v)\}) = w_{ik}(G_u^0 \cup G_v^0, u, v) = T_{u,v} = G_u^0 * G_v^0 = G_u^0 *_f G_v^0(f(u) = v)$;
2) $T_{u,v}^w = (V = \{u, w, v\}, E = \{(u, w), (w, v)\}) = T_{u,w} \cup T_{w,v} = w_{ik}(T_u^0 \cup T_w^0, u, w) \cup w_{ik}(T_w^0 \cup T_v^0, w, v) = (T_u^0 * T_w^0) \cup (T_v^0 * T_w^0)$.

where: $w_{ik}$ - the operation of adding an edge into connected graph; * - the operation of combining two connected graphs. In addition, the result of combining two graphs will be a connected graph, if even one or two of them does not meet connectivity condition. In particular, for a connected graph $G$, graphs corresponding to the expressions $(T_u^0 \cup T_v^0) * G$ and $(T_u^0 \cup T_v^0) * (T_u^0 \cup T_v^0) * (T_w^0 \cup T_s^0)$, where: $u, v, w, s$ in pairs different vertices, will be connected graphs. The algebraic expression describing the single channel bus in graph algebra notation has the following form: $(T_{S_1}^0 \cup \ldots \cup T_{S_r}^0 \cup T_{K_1}^0 \cup \ldots T_{K_n}^0) * T_B^0$. Technical solutions developed on the basis of research are based solely on the multi-channel bus. Suppose the complete multi-bus (i.e. each service provider and recipient is connected to each of the buses) consists of $m$ channels, $S_r$ recipients and $K_n$ service providers. Then its physical form and its notation in the form of a tripartite graph have the form presented in Fig. 3. It can be assumed from a technical point of view, that the $B_1, \ldots, B_m$ buses are logical channels functioning in the one common physical channel. The algebraic expression describing the above network has the following form:

$$(T_{S_1}^0 \cup \ldots \cup T_{S_r}^0 \cup T_{k_1}^0 \cup \ldots \cup T_{k_n}^0) * T_B^0$$

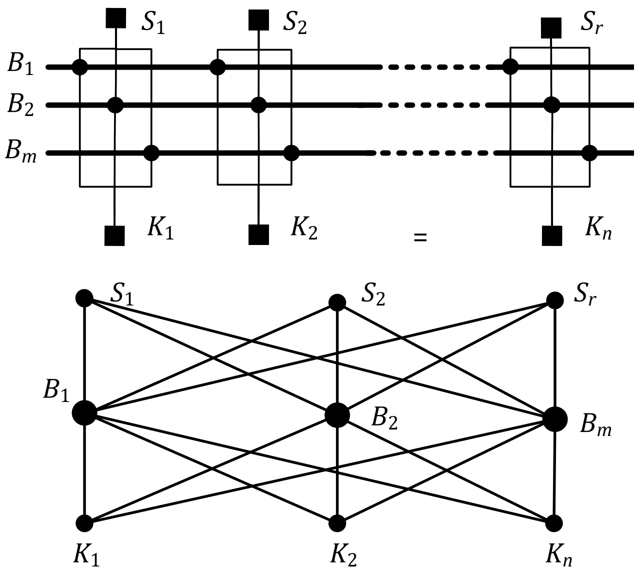

Fig. 3. The network with multi-channel bus presented as tripartite graph

### C. Additional functions of modified network

Having a set of independent computational nodes and reconfigured connections in one system creates an excellent opportunity to construct a scalable computational unit. Particular interest in this type of solutions in IIS result from the following conditions:

1) The concentration of cyber-attacks on the top two layers of the network model. They have the resources necessary to safely support production technologies. Attacks on the lower layers of the model are very rare;
2) In order to maximize the level of security it is appropriate to separate computational diagnostic components outside the areas of the industrial network model available to cybercriminals.

In the case of the cyber-attack on IIS resources, degradation of computational and communication resources used in the security subsystem occurs. If the threat detection system is based solely on polynomial combinatorial algorithms or the analysis of limited data, the problem of insufficient computational resources rather does not appear. It becomes valid at the time of application of time-complex algorithms, e.g. providing machine learning or bio-inspired analysis. If the detection system acts only as a part of the top layer, degradation of its resources is highly possible. That is why, in some enterprises, independent computational resources with low usage in a stable mode of operation are constructed on the basis of the edge network components, designed solely to operate the detection system. Their load increases drastically when threats appear. The solution offered assumes the separation of scalable computational resources, independent of other components of the upper IIS layer. For this purpose, it was decided to prioritize the procedures for parallelizing processing in a heterogeneous environment with multi-channel optical connections. Communication is based on the repeatedly

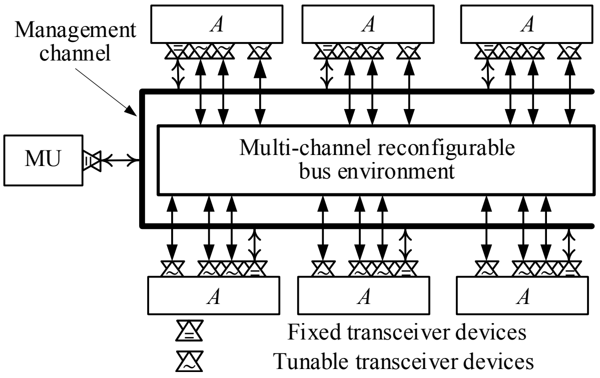
Management channel
A
MU
Multi-channel reconfigurable bus environment
Fixed transceiver devices
Tunable transceiver devices

Fig. 4. The concept of the system: MU - management node, controlling bus system

folded bus with the possibility of its physical division and virtual reconfiguration of the connection network. The first level of parallelization is based on a separate security sever and uses CUDA technology [22], [23]. At the second level, in the event of a computational power deficit, a heterogeneous computational system is dynamically constructed using the available resources of analyzers based on dedicated hardware of IIoT components.

### D. Bus communication systems in IIS

The basic concept of a parallel reconfigurable computational system with multi-bus connections is shown in Fig. 4. It was assumed that all computational functions will be performed in the system based on the resources of the management node $MU$ and $A$ traffic analyzers. For security reasons, the computational servers of the upper layers are not used. Each of the analyzers has been equipped with at least one fixed and one variable single-channel communication interface. If it is possible, then the parameters of each of the fixed channels are unique in the entire structure, thanks to which there is no limit to setting up networks with any connection architecture. A deviation from this rule is the management channel to which all computational nodes are connected. This channels always uses broadcast and is only intended for sending information about the configuration of transceivers. The use of a device that supports a fixed communication channel is also expedient from an economic point of view - the price of these devices is many times lower than tunable devices. Each of the $A$ analyzers is equipped with a set (2 to 4) of fixed or tunable transceiver devices. They are connected in various ways to logical buses (e.g. wave channels) $B_1, \ldots, B_4$, thus responding to different user needs. The use of independent optical fibers (physical buses) instead of logical buses significantly changes the properties of the above organizations. If the system from Fig. 5a is equipped with integrated transceivers and logical buses $B_1, \ldots, B_4$ it can be used to improve the failure resistance of transceivers. In addition, if many analyzers compete for access to logical buses, the amount of information provided or retrieved from the node may be multiplied by using routing addressed to this type of architecture. If $B_1, B_2$ buses are physical channels, the system from Fig. 5a improves fault tolerance of this type of channel. Past experience shows that damage to communication channels usually occur at the physical level, i.e. optical fibers or transceivers. Similarly, the other architectures from Fig. 5 can be analyzed. All the analyzers operate at sensors and actuators level. If the transmitting and receiving devices can be separated, and instead of the traditional bus the folded bus is used, the following effective methods of minimizing communication delays appear, by selecting the appropriate part of the bus, where the information signal is delivered or supplied. Each of the analyzers has its own computational power, which can be used in the system, both in a group and independently for each device.Computational servers will be made available in the system to increase computational efficiency. They can be independent devices or components of an industrial computer network deployed at the production management level.

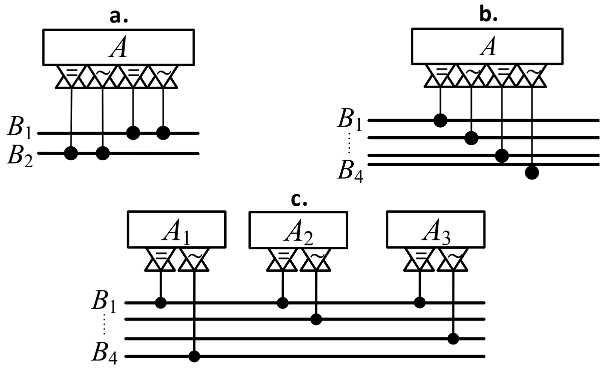

Fig. 5. Connections between node and the virtual bus: a. multiple complete; b. single complete; c. partial

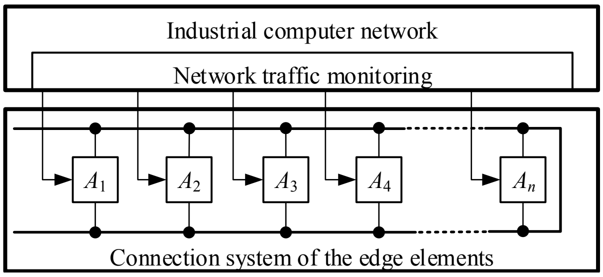
Industrial computer network
Network traffic monitoring
$A_1$
$A_2$
$A_3$
$A_4$
$A_n$
Connection system of the edge elements

Fig. 6. Functional components of the industrial computer network with horizontal connection of edge elements

## III. Connection architecture of the edge elements

### A. The concept of the system organization

Let's analyze a fragment of the architecture of an industrial computer network using IIoT with horizontal connection of edge elements based on multi-bus communication with passive joining of users. The functional components of such a system are shown in Fig. 6. Further considerations will apply only to the edge connection system located in the lower part of

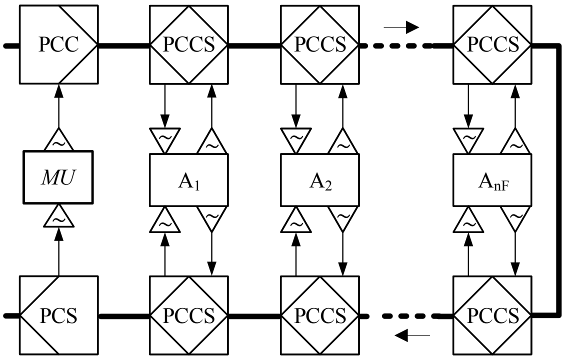

Fig. 7. Connection of edge elements by means of a multi-channel simplex folded bus with tunable transceiver elements with a symmetrical organization without load balancing

Fig. 6. Each of the $n$ analyzers is connected to the lower and upper part of the folded multi-channel bus. Both statements are used only to perform communication operations as a part of a computational system built based on edge elements. The number of connections is limited only by the possibilities of the edge element. In addition, each of the analyzers receives a signal for analysis from the industrial computer network to fulfill its basic functions. The power of edge elements (e.g. based on Raspberry Pi 4) is so high that they can fulfill the function of analyzers and computational elements solving other tasks. In Fig. 6, for a known reason, the management bus and the server controlling the bus set are omitted. There are many alternative ways to improve the communication efficiency of the multi-bus systems. The range of possible methods and measures to improve the quality of inter-node connections in multi-channel industrial networks is very wide. However, not all available solutions are equally attractive. Since the priority of the work being performed is to minimize the costs of IIS construction and operation, it was decided to use a passive bus in which the organization of logical communication channels is performed by means of transceivers directly in the connected node. For similar reasons, it was decided to use simplex bus channels. Fault tolerance, reliability and survivability considerations have decided to use multiple folded physical channels. Their use is particularly beneficial from the point of view of the availability of the communication subsystem. The networks based on them maintain connectivity even with repeated damage to the physical network. For similar reasons, it was decided to use symmetrical communication channels without load balancing. Passive connection architecture plays a decisive role in organizing the connections of edge elements. Other choices are of secondary importance. The connection system architecture meeting the above criteria is shown in Fig. 7. Each of the analyzers (computational nodes) has been equipped with two tunable transmitters and two tunable receivers synchronized with them in pairs. Unlike previously discussed solutions, the management node was also equipped with tunable transceiver devices. The application of this solution resulted from the desire to unify the system.

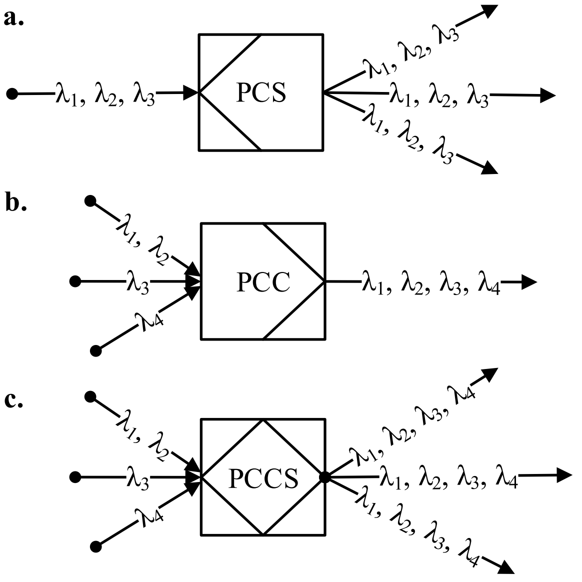

Fig. 8. Passive couplers: a. Physical channel separator (PCS); b. Physical channel connector (PCC); c. Physical channel connector-splitter (PCCS)

In practice, we also use other solutions to organize the management channel, including: a pre-imposed logical channel with a fixed transceiver and receiver, and the organization of the management channel in an independent physical channel. The communication components of the system in Fig. 7 are shown in Fig. 8. As $\lambda$ we denote independent communication channels. All of the following elements are passive and their purchase does not generate high costs. They are available on the market.

### B. Determining acceptable parameters of the system

Let's estimate acceptable values of the system parameters shown in Fig. 8. The minimum number of logical transmission channels necessary to build a connected communication system is 2. The first of the channels is used as a management, the second connects computational nodes (analyzers - network edge nodes). Both channels operate in broadcast mode. The maximum number of $K^{lk}_{max}$ required to build a complete topology with direct duplex connections is:

$$K^{lk}_{max} = n_F^2 - n_F + 1 \tag{1}$$

where: $n_F$ - number of computational nodes of the system. Minimum number of receiving and transmitting devices of the $i$-th node of connected system, $k^R_{i_{min}}$ and $k^T_{i_{min}}$ respectively, are identical and equals 2 ($k^R_{imin_{in}} = k^T_{i_{min}} = 2$). Maximum number of $k^R_{imin}$ and $k^T_{i_{min}}$ are also even and equal $n_F$. The total minimum numbers of receiving and transmitting devices, $K^R_{min}$ and $K^T_{min}$ respectively, are defined as:

$$K^R_{min} = \sum_{i=1}^{n_F} k^R_{i_{min}} + 1 = 2n_F + 1 \tag{2}$$

$$K^T_{min} = \sum_{i=1}^{n_F} k^T_{i_{min}} + 1 = 2n_F + 1 \tag{3}$$

Let's estimate system delays without routing. The value of the maximum transmission delay $\tau_{max}$ depends not only on the technical characteristics (e.g. geographical size) of the system, but also on the occupancy level of the transceiver and communication bus. Let's consider a simple example: nodes $A_1$ and $A_2$ equipped with sets of transceiver elements will communicate. If any upper receiving element $A_2$ is free, the distance over which the signal will be transmitted is minimal (part of the upper bus fragment) and the number of passive splitting elements passed is 2. However, if the upper receivers of node $A_2$ are occupied, the information signal will only be received by the lower element $A_2$. The signal will cover almost the entire bus length, including $2n_F - 1$ passive PCCS components. This should not happen if the system provides for optimization of transmission performed by the management node. The system's answer will be blocking the communication channel. Determining the maximum delay value will start from the wort case, when directions of the physical and logical channels are opposite and the length of the communication channel will be maximum. Let there be unloaded logical channels and unoccupied transceiver devices in the analyzed system. The maximum delay will occur on the path between nodes $A_2$ and $A_1$ because the information channel will be created between the upper interface $A_2$ and the lower $A_1$. In systems without routing, the described situation is acceptable and in no way results from the occupation of any receiving or transmitting elements. In the case of a homogeneous connection system, in which the distance between the computational nodes and the time parameters of the components are identical, the delay value $\tau_{max}$ is described by the following expression:

$$\tau_{max} = t^{B_{2\to1}} + 2(n_F - 1)t^{PCCS} + t_\nu^R + t_\nu^T \tag{4}$$

where: $t^{B_{2\to1}}$ - transmission delay of the physical channel between $A_2$ (device is connected to the upper part of the bus) and $A_1$ (device is integrated with the lower part of the bus); $t^{PCCS}$ - delay of the integrating PCCS component; $t_\nu^R, t_\nu^T$ - time delay of receiving and transmitting devices, respectively. In the most unfavorable case, the length of the used physical channel fragment is comparable to its total length. Therefore, $t^{B_{2\to1}}$, where $t^B$ - signal transmission delay between bus ends. In addition, for large $n_F$ values, the delays of the transceiver devices can be neglected. Then, expression (4) can be written in a simplified form:

$$\tau_{max} \approx t^B + 2n_F t^{PCCS} \tag{5}$$

We will determine the maximum delay in sending information through a channel in a routed communication network. We will use the previously proposed routing algorithm that transfers the creation of the transmission channel between the upper and lower parts of the folded bus. In a routed system, the maximum delay occurs when passing information over the channel connection $A_1$ to $A_{n_F}$ or $A_{n_F}$ with $A_1$. In this case, the expression (4) can be written as:

$$\tau_{max}^r = t^{B_{1\to n_F}} + n_F t^{PCCS} + t_\nu^R + t_\nu^T \tag{6}$$

where: $\tau_{max}^r$ - maximal transmission delay in the architecture with routing; $t^{B_{1\to n_F}}$ - signal transmission delay through the physical bus between nodes $A_1$ and $A_{n_F}$. If the upper and lower part of the folded bus are symmetric, then $t^{B_{1\to n_F}} \approx 0.5t^{B_{2\to1}}$. Considering the expressions (4) and (6), the following condition can be accepted:

$$\tau_{max}^r = 0.5\tau_{max} \tag{7}$$

In the computational systems with symmetric architecture of the transceiver devices, minimal value of the transmission delay does not depend on whether the system uses routing and for types (with or without routing) is determined by the expression:

$$\tau_{min} = \tau_{min}^r = t^{B_{1\to2}} + 2t^{PCCS} + t_\nu^R + t_\nu^T \tag{8}$$

where: $\tau_{min}^r$ - minimum value of the transmission delay for the architecture with routing; $t^{B_{1\to2}}$ - delay of the signal transmitting through the physical bus between neighboring nodes, in particular between $A_1$ and $A_2$. The last, significant time parameter of the computational system connection efficiency is the average value of the transmission delay $\tau_{avg}$. We will specify its value for organizations without routing. If the distribution requests for the set of transmission channels is homogeneous, the value $\tau_{avg}$ can be defined as the arithmetic mean of the minimum and maximum delay of signal transmission through the channel, i.e.:

$$\tau_{avg} = 0.5(\tau_{min} + \tau_{max}) \tag{9}$$

Considering the expressions (4) and (8), the formula (9) can be written in the following way:

$$\begin{aligned}\tau_{avg} &= 0.5(t^{B_{2\to1}} + (2n_F - 1)t^{PCCS} + t_\nu^R + t_\nu^T + \\ &+ t^{B_{1\to2}} + 2t^{PCCS} + t_\nu^R + t_\nu^T = \\ &= 0.5(t^{B_{2\to1}} + t^{B_{1\to2}} + (2n_F + 1)t^{PCCS} + 2t_\nu^R + 2t_\nu^T)\end{aligned}$$

Note that $t^{B_{2\to1}} \approx t^B$ and $t^{B_{1\to2}} \approx t^B/2n_F$. Therefore, if $n_F \gg 2$ the above expression can be written as follows:

$$\tau_{avg} \approx 0.5\tau_{max} \tag{10}$$

Based on the expression (9) we will determine the average delay $\tau_{avg}^r$ for an organization with routing. Using the expressions (6) and (8), delay value can be estimated:

$$\begin{aligned}\tau_{avg}^r &= 0.5(t^{B_{1\to n_F}} + n_F t^{PCCS} + t_\nu^R + t_\nu^T + t^{B_{1\to2}} + \\ &+ 2t^{PCCS} + t_\nu^R + t_\nu^T) = \\ &= 0.5(t^{B_{1\to n_F}} + t^{B_{1\to2}} + (n_F + 2)t^{PCCS} + 2t_\nu^R + 2t_\nu^T)\end{aligned}$$

If we assume that $t^{B_{1\to n_F}} \approx 0.5t^B$ and $t^{B_{1\to2}} \approx t^B/2n_F$, then for $n_F \gg 2$, $t^{B_{1\to n_F}} \gg t^{B_{1\to2}}$. Then:

$$\tau_{avg} \approx 0.5\tau_{max}^r \approx 0.25\tau_{max} \tag{11}$$

Analysis of transmission time characteristics shows that routing improves most of them. At the same time, along with the reduction in the length of the transmission channel, the signal attenuation also decreases, which minimizes the requirements, and thus reduces the overall cost of the system construction. The basic conclusion of the above analysis is the desirability of extensive use of the system with routing.

### C. *Description of the system functioning*

The device developed at the current stage is intended to improve the security of the industrial information system. Built-in autonomous analyzers record all manifestations of network activity. The analyzers also make a preliminary detection of any anomalies, this process is continued in the management server. Each of the analyzers can work in one of several modes. In basic mode, the analyzer tracks the traffic in the assigned wired or wireless network segment. All analyzers are connected via two communication channels. The first of them - broadcasting - controls communication between the analyzers and the management server, while the second (direct - dynamically set up) is the right channel for transferring information between security system objects. The acquired data on traffic anomalies in the segments are stored in the monitoring unit (edge element) and periodically sent to the server(s) of the security system. For this purpose, a two-point virtual channel connecting the selected monitoring node with the appropriate server is dynamically established. The management channel participates in the connection setup procedure. Dynamic set up of the transmission channels is also beneficial from a security point of view. In this way, the number of entries to the system available to the intruder is minimized. The number of virtual channels operating simultaneously depends only on the number of transceivers installed in the servers. It is allowed to connect all analyzers and servers with a set of broadcast channels. Pre-processing of traffic information in the segment is performed directly in the analyzers. For this purpose, each monitoring node stores the symptoms of threats, which are in fact traffic patterns, indicating the appearance of a threat. The list of symptoms is not constant, it is created autonomously by computational servers based on data sent by nodes using a set of analytical algorithms using AI methods (machine learning, big data algorithms) and biologically motivated tools (immunological and evolutionary methods). Patterns are periodically sent to monitoring nodes. If traffic similar to one of the patterns appears in the segment, the monitoring node requests to set up a temporary two-point communication channel connecting it with the selected computational server. The channel set-up consists in determining the number of the common communication channel for the monitoring node and the server. Until the dangerous symptom stops, the traffic is analyzed in real time on the security server. Further decisions are made about how to deal with the potential threat. According to Fig. 5, the server can simultaneously support multiple virtual networks, it can also process information from multiple analyzers. Thanks to this, the security system can work in the real time. If the detected anomalies in the assessment of the security server may threaten the security of IIS, the functioning of the system is limited and in extreme cases the system is stopped. The algorithm of functioning of the threat detection system is presented in Fig. 9. The described procedure illustrates one of many possible works performed by the edge elements thanks to their connection via a bus communication channel. With time, as the edge element resources increase, the range of such activities will be practically unlimited.

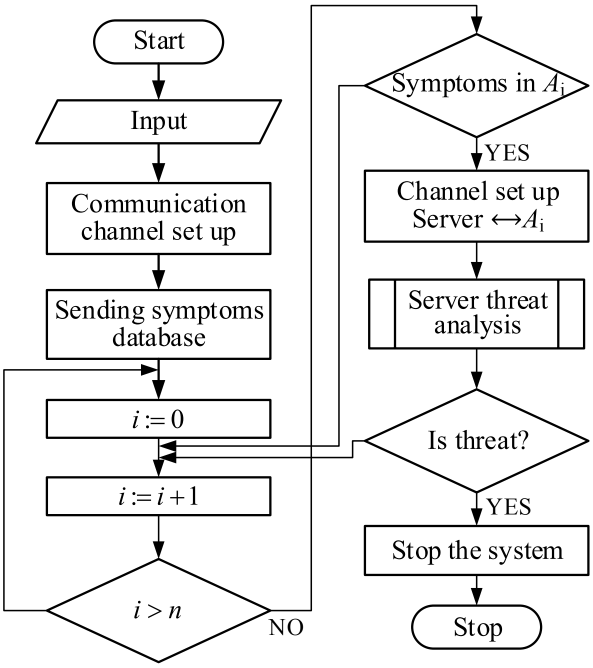

Fig. 9. Stages of functioning of the system of connecting edge elements

## IV. Experimental results

In order to process results from the research, we used IBM SPSS Modeller, OriginPro, Process Explorer and software-hardware prototype of designed device. It consisted of two steps. The first of them performed the procedure of filling the symptom base with attack patterns. Filling was outlined until balanced accuracy on the training and test set exceeded 0.986. During the process, normalization of numeric parameter values and conversion to binary values of text parameters, as well as classification of database records, was performed. In the second step, using the software testing sequence generator imitating the output of the measurement sensors and the machine time measurement tool, the attack detection time was determined for the given accuracy and completeness of attack detection. The minimum value of all basic detection parameters (correct classification, precision, completeness and metric) was set at a minimum of 80%. Teardrop, smurf, satan, portsweep, pod, normal,nmap neptune, ipsweep and back attacks were examined. The study analyzed the effectiveness of four architectures: CPU, CPU + 8 RPi, CPU + GPU and CPU + GPU + 8 RPi. Each of the experiments was performed 1000 times, the published results were subjected to processing characteristics of empirical data. The results of the experiment are shown in Fig. 10. Previous work has shown the desirability of using edge elements to increase the computational power available in industrial computer networks. With relatively small investments, it can get performance comparable with many times more expensive commercial solutions. Currently,

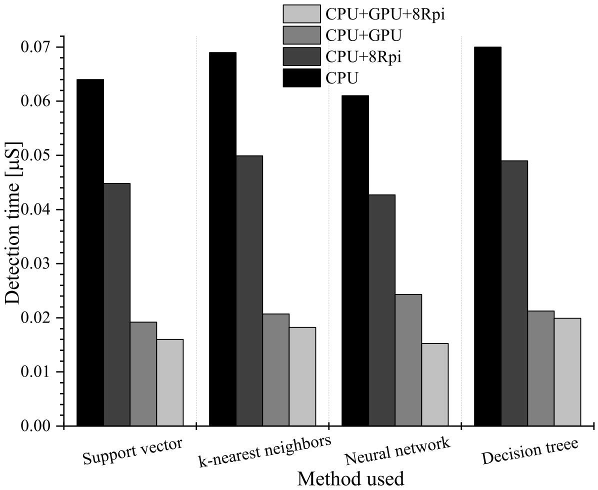

Fig. 10. Attack detection times for various computational system configurations and detection methods

the designed solution in real conditions is being tested, in which threats will be detected. Previous work with the test system utilized traffic load and computational tasks from a test event generator configured to bring tasks and loads similar to those found in an industrial computer network.

## V. Summary and further works

The paper presented the method of horizontal joining of edge elements of an industrial computer network. Due to the wide application of passive transmission network based on folded buses, the solution is characterized by low costs and high resistance to physical channels damage. The solution has a very wide development potential, resulting, among others with:

1) Possibilities of repeated folding of the communication bus. If the number of interfaces allowing to build transceivers increases, it will be possible to effectively use folding, increasing the system's resilience and minimizing communication delays due to routing;
2) Possibilities of splitting or grouping buses. Users can also be grouped [9]. Thanks to this, there are wide possibilities of organizing network traffic ensuring load balancing of communication channels and minimizing delays;
3) Continuous development of the system's hardware base. Arduino and Raspberry devices are constantly evolving, providing new technical possibilities. First of all, improving the hardware base will result in an increase in the number of independent communication channels connecting the processing element to the bus system.

Further work will focus on increasing the size of the prototype and developing tools for computer-aided design of such systems. The functionalities offered will be combined with edge elements, currently focused on the analysis of information security threats monitored by industrial computer network analyzers. Currently available computational resources are used to detect threats based on collected traffic information. A number of modern methods are used for this purpose, including machine learning and intelligent data analysis. In the future, it is planned to develop methods to support the resources of industrial computer networks in solving time-consuming tasks, such as combinatorial optimization and graph algorithms.

## References

[1] A. Dehghantanga and K. K. R. Choo, *Handbook of Big Data and IoT Security*. Springer, 2019.

[2] C. Alcaraz, *Security and Privacy Trends in the Industrial Internet of Things*. Springer, 2019.

[3] M. Alazab and M. J. Tang, *Deep Learning Applications for Cyber Security*. Springer Nature, 2019.

[4] M. Collins, *Network Security Through Data Analysis. Building Situational Awareness*. Sebastopol: O'Reilly, 2013.

[5] M. J. Zuo, *Optimal reliability modeling: principles and applications*. Hoboken: John Wiley & Sons, 2003.

[6] M. L. Shooman, *Reliability of Computer Systems and Networks: Fault Tolerance Analysis and Design*. New York: John Wiley & Sons, 2002.

[7] P. Stavroulakis, *Reliability, survivability and quality of large scale telecommunication systems*. Chichester: John Wiley & Sons, 2003.

[8] S. Bhattacharjee, *Practical Industrial Internet of Things Security*. Birmingham: Packt, 2018.

[9] P. Hajder and Łukasz Rauch, "Reconfiguration of the multi-channel communication system with hierarchical structure and distributed passive switching," in *Computational Science – ICCS 2019, 19th International Conference*. Faro, Portugal: Springer, 2019, pp. 502–516.

[10] A. Banafa, *Secure and Smart Internet of Things (IoT). Using Blockchain and AI*. Gistrup: River Publishers, 2018.

[11] A. Kott and E. J. Colbert, *Cyber-security of SCADA and Other Industrial Control Systems*. Cham: Springer Nature, 2016.

[12] F. Paganelli and G. J. van Viet, *Resilience and Reliability on AWS*. Sebastopol: O'Reilly Media, 2013.

[13] B. Bollobas, *Modern Graph Theory*. New York: Springer, 1998.

[14] J. Bang-Jensen and G. Z. Gutin, *Digraphs: Theory, Algorithms and Applications*. London: Springer, 2010.

[15] M. D. Rahman, *Basic Graph Theory*. Cham: Springer International Publishing AG, 2017.

[16] P. Mathis, *Graphs and Networks*. New York: Wiley & Sons, 2010.

[17] R. Diestel, *Graph Theory*. New York: Springer, 2010.

[18] N. J. Smith and A. P. Sage, "An introduction to hierarchical systems theory," *Computers & Electrical Engineering*, vol. 1, pp. 55–71, 1973.

[19] M. D. Mesarovic, D. Macko, and Y. Takahara, *Theory of hierarchical multilevel systems*. New York: Academic Press, 1970.

[20] M. Hajder and J. Kolbusz, *Formalizacja projektowania architektury systemów informacyjnych*. Rzeszow: Wyższa Szkoła Informatyki i Zarządzania, 2015.

[21] M. Hajder, M. Nycz, and J. Kolbusz, "Grafowe reprezentacje obiektów technicznych," in *Innowacyjna gmina. Informatyka w jednostkach samorządu terytorialnego*, M. Hajder, Ed. Rzeszów: Wyższa Szkoła Informatyki i Zarządzania, 2014, pp. 163–172.

[22] L. Lladis, Y. Kim, S. sarafijanovic, and V. Sarafijanovic, "Performance evaluation of a tape library system," in *IEEE 24th International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems (MASCOTS)*, London, 2016.

[23] T. Sterling, M. Anderson, and M. Brodowicz, *High Performance Computing. Modern Systems and Practices*. Cambridge, MA, USA: Morgan Kaufmann, 2018.