Intelligent Secure Data Aggregation in WSNs

Olena Semenova[1], Natalia Kryvinska[2], Serhii Baraban[3], Maksym Prytula[1],
and Volodymyr Martyniuk[1]

[1] *Vinnytsia National Technical University, Vinnytsia, Ukraine,*
[2] *Comenius University in Bratislava, Bratislava, Slovakia,*
[3] *Poznan University of Technology, Poznan, Poland*

https://doi.org/10.26636/jtit.2025.3.2220

Abstract — The paper discusses the problem of secure data aggregation in wireless sensor networks (WSNs) – a procedure that is of critical importance for reducing energy consumption, minimizing transmission overhead, and thus prolonging network lifetime. Due to the limited computational and energy resources of WSN nodes, traditional aggregation methods often fail to perform effectively in dynamic heterogeneous environments. With such a context taken into consideration, this study emphasizes the potential of artificial intelligence techniques, such as neural networks, genetic algorithms, and fuzzy logic, to enable adaptive aggregation approaches tailored to environmental and network-specific parameters. Furthermore, the integration of fuzzy logic, genetic algorithms, and artificial neural networks into a hybrid system leverages the strengths of each approach, resulting in enhanced adaptability and accuracy of the aggregation process. As part of the investigation, a fuzzy inference system (FIS) model was developed that incorporates attributes such as energy, current load, distance to the base station, and trust level. The model was implemented in Matlab using the Fuzzy Logic Designer toolbox. To further improve system performance, a genetic algorithm was applied to optimize membership functions. In the final phase, the model was transformed into an adaptive neurofuzzy inference system (ANFIS) which was trained using simulated data within Matlab. The simulation results demonstrate that the proposed hybrid approach ensures flexible, robust and energy-efficient control of the data aggregation process under dynamically changing conditions in which WSNs operate.

Keywords — artificial intelligence, data aggregation, fuzzy logic, security, wireless sensor networks

1. Introduction

Wireless sensor networks (WSNs) play an important role in a wide range of applications, including industrial manufacturing, smart cities, automotive, healthcare and environmental monitoring [1]. In these environments, autonomous sensor nodes are distributed to monitor various conditions. The sensors gather data and transmit it to a central node for processing and further analysis.

Scalability, self-organization, and adaptability are among the characteristics of WSNs that make them effective tools for processing data in real-time, particularly in potentially hazardous or hard-to-reach situations. Data aggregation and routing are two essential processes in WSNs that contribute to improving energy efficiency, extending network lifetime, and minimizing communication overhead [2].

Data aggregation refers to the process of collecting useful data. In WSNs, appropriate data aggregation procedures are required to preserve limited resources. The primary objective of aggregation algorithms is to collect data in a manner that optimizes energy efficiency, thereby extending the network's lifespan.

As WSNs are characterized by restricted computational capabilities, limited memory, and finite battery capacity, all this complicates the development process. Moreover, in some cases, this may result in applications that are tightly integrated with network protocols [3]. Furthermore, data aggregation is employed to address overlapping in data routing. When data from various sensor nodes converge at the same node on their return to the sink, they are aggregated as if they pertain to the same data.

In general, data aggregation methods relied upon in WSNs can be classified into four categories: cluster-based, tree-based, in-network, and centralized data aggregation [4].

In the cluster-based approach, the network is segmented into clusters. Each cluster is made up of a group of sensor nodes, with one node designated as the cluster head. The cluster head is responsible for data aggregation, where the collected data is combined and subsequently sent to the sink.

These clusters operate in two distinct phases: during the initial (setup) phase the cluster selection process occurs and clusters are established. The steady phase follows, in which the cluster is functioning. Throughout the steady phase, all nodes within the cluster, including the cluster head, continuously sense their environment for specific data in a regular way.

All member nodes transmit the detected data to the cluster head which aggregates the information and forwards it to the sink. This strategy minimizes bandwidth usage by decreasing the number of packets that need to be transmitted. Furthermore, the data aggregation process relied upon in this method not only reduces the number of packets sent directly to the sink, but also lowers energy consumption due to the shorter transmission distances. However, it suffers from a drawback, namely increased latency. Cluster-based data

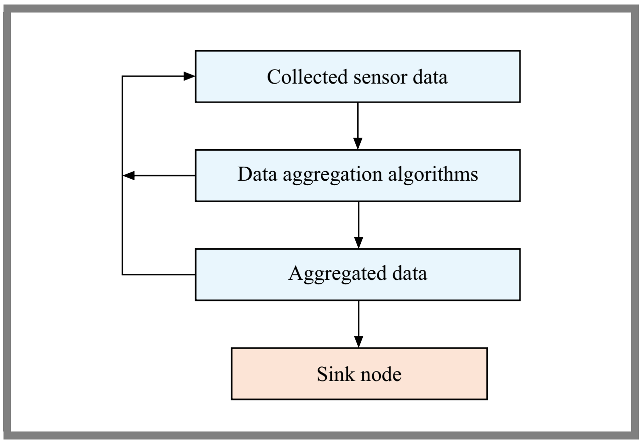


Fig. 1. Diagram presenting the general data aggregation algorithm.

aggregation techniques include LEACH, HEED, SEP, and PAgIoT.

In the tree-based approach, aggregation trees are established in such a way that every data transmission requires the formation of spanning trees. In this framework, the base station functions as the root of the tree, whereas sensor nodes act as leaves. Data are collected by the leaves and transmitted towards the root with the parent nodes consolidating the data throughout the networks.

In-network aggregation represents a comprehensive method for collecting and processing data at intermediary nodes, in addition to facilitating the routing of information through multi-hop networks. Its primary objective is to minimize the consumption of energy required to perform the process. This method may either decrease the size of the data, leading to a reduction in the amount of data that needs to be transmitted subsequently, or maintain the width by combining all received packets.

In the centralized approach, all sensors transmit the collected data as data packets to a central node or base station via the shortest available route. The function of the aggregator or header node is to compile the data received from the other nodes, after which the consolidated data are sent as a single packet.

Figure 1 shows the general data aggregation algorithm through different aggregation techniques [5]. The algorithm utilizes sensor data from the sensor nodes and aggregates them using several aggregation algorithms, including the centralized approach, low energy adaptive clustering hierarchy (LEACH), and tiny aggregation (TAG), among others. These aggregated data are then transmitted to the sink node by selecting the most efficient path.

However, various efficient aggregation protocols do not meet the resource constraints. Moreover, numerous security threats exist, including but not limited to snooping attacks, wormhole attacks, black hole attacks, packet replication attacks, denial-of-service (DoS) attacks, and distributed denial-of-service (DDoS) attacks. In many cases, the process of optimizing performance of WSNs concerns more than one of the metrics, thus necessitating the application of multi-objective optimization [6].

2. Problem Definition

During the data aggregation process, several challenges must be addressed. It is evident that it is difficult to overcome all these challenges simultaneously. The most significant challenges include the following [7]:

- **Data redundancy**. Sensor nodes often detect similar types of data and even the same events, leading the sink node to collect redundant information. This results in a waste of time, energy, and other resources.
- **Delay**. In some cases, data from more distant nodes arrive late at the sink or root node, causing the aggregation process to commence later than intended. Additionally, aggregations at intermediate levels can further increase the delay.
- **Accuracy**. There are two primary types of accuracy-related issues. Firstly, the aggregator function serves as an approximation mechanism. Therefore, some precision is inevitably lost during the data forwarding process. Secondly, there may be a compromised node that transmits false or inappropriate data to the aggregator node. The aggregator node does not guarantee the correctness of these data and proceeds to process them.
- **Traffic load**. In specific situations, the aggregator node may become overloaded. This occurs when load balancing is not effectively implemented or when clusters are of unequal sizes.
- **Aggregation freshness**. Data from similar frames should be aggregated, while the use of outdated stored data or the aggregation of data from multiple frames across different time periods should be avoided, as such an approach compromises freshness.
- **Security**. As wireless sensor networks are often implemented in hostile environments, security issues, particularly involving data confidentiality and integrity, become crucial.

Therefore, when a malicious node infiltrates the network, ensuring the delivery of packets to the base station becomes a challenge. Given the resource limitations inherent in WSNs, it is essential to guarantee packet delivery while minimizing energy consumption. Transmission of redundant data within the network accelerates the depletion of energy in a node. This leads to network partitioning which, in turn, results in increased energy consumption and a consequent reduction in the overall lifetime of the network.

Therefore, in conjunction with data aggregation, it is essential to ensure the security of successful data transmission to the base station through the efficient use of available resource parameters.

To address a variety of challenges related to energy efficiency, coverage maximization, and security provision in WSNs, artificial intelligence (AI) can be applied effectively in wireless sensor networks by enabling smart decision making [8].

AI denotes the ability of a system to perform tasks that require human-like intelligence, emulating human thought processes or concepts. It is regarded a significant domain within computer science that aims to enhance machine intelligence. The predominant techniques employed in AI include search algorithms, learning methodologies, fuzzy systems, knowledge representation, and reasoning processes [9].

AI finds application in addressing numerous intricate issues across diverse fields such as security, finance, healthcare, and transportation, leveraging its proficiency in managing incomplete and noisy data, tackling nonlinear problems, and demonstrating suitability for prediction and accelerated post-training generalization. The different AI techniques used to tackle WSN-related challenges comprise fuzzy logic, artificial neural networks, evolutionary computation, nature-inspired approaches, swarm intelligence, deep learning, reinforcement learning, and hybrid models [10].

3. Literature Review

This section discusses various publications associated with the application of AI used for data aggregation in WSNs.

Study [11] proposed a fuzzy-based secure data aggregation protocol which improves the lifetime of the network, maximizes the packet delivery ratio, and minimizes the end-to-end delay. Paper [12] introduces a fuzzy-based data aggregation technique to ensure energy efficiency in wireless sensor networks. A similarity-aware data aggregation process using a fuzzy c-means approach is discussed in [13].

Current secure data aggregation protocols represent a trade-off between security and the shortest path. To address this problem, the protocols mentioned in [15] are developed by integrating the distributed k-means algorithm and the fuzzy c-means algorithm. In work [15], a data aggregation algorithm was constructed based on a self-organizing feature mapping neural network.

In [16], a machine learning-based approach for an energy efficient data aggregation model in WSNs was described. The security feature can also be incorporated into the proposed model. Study [17] presents four algorithms for data aggregation. Three of them are based on backpropagation neural networks.

To mitigate energy consumption and ensure data aggregation in WSNs, study [18] presents a cluster-based data aggregation routing approach with a genetic search algorithm. This method aims to reduce energy use. In [19], a novel hybrid LEACH algorithm was introduced for data aggregation based on a genetic algorithm to optimize WSN parameters, including energy consumption.

As different AI-based techniques for data aggregation in WSNs have their own pros and cons [20], a combination of these approaches may be beneficial, as it leverages the complementary strengths of the different models, thus overcoming the limitations of individual methods.

Here, the authors propose using a hybrid approach for data aggregation in wireless sensor networks that combines three AI technologies: fuzzy logic, artificial neural networks, and genetic algorithms. This integration allows one to account for the uncertainty and incompleteness of the sensor data as well as dynamic changes in the network and the surrounding environment. The proposed approach ensures a high level of adaptability and reliability while making data aggregation-related decisions.

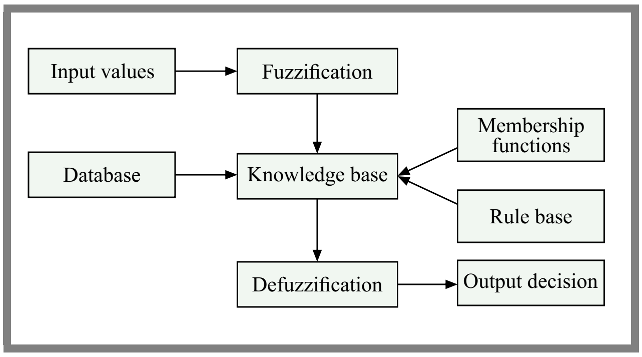


Fig. 2. Diagram presenting the proposed FIS architecture.

4. Methodology

In this section, a fuzzy inference system (FIS) intended for data aggregation in WSNs will be developed. A common FIS consists of four functional units: fuzzification, rule base, decision making, and defuzzification (Fig. 2).

The fuzzification unit transforms crisp inputs into linguistic variables. A rule base consists of a collection of fuzzy if-then rules. The decision-making unit conducts inference based on the fuzzy if-then rules. The defuzzification unit converts the fuzzy results generated by the inference system into precise outputs [21].

To develop an FIS, input in the form of linguistic variables along with their corresponding terms must be established. Then, the membership functions associated with these inputs are to be defined. The inputs include all conceivable states of the process being controlled, whereas the output represents all potential control actions. Subsequently, a rule base must be specified, consisting of a set of if-then fuzzy rules to characterize the controlled states. Finally, to evaluate the performance of the FIS, a simulation is performed.

The inputs of the (FIS) proposed in this study include the following: energy level of the node, distance from the base station or the sink node, load, and node trust level. Its output variable is the aggregation priority of the node.

The energy level of a sensor node is the remaining battery capacity which directly influences its suitability for performing energy-intensive tasks, such as data aggregation. Nodes with higher energy levels are preferred to act as aggregators to prolong the overall network lifetime. From a security perspective, maintaining a minimum energy threshold is critical, as low-energy nodes are more vulnerable to exhaustion attacks and may be less reliable in executing secure aggregation protocols.

For the energy input variable E such linguistic terms as “low” and “high” are applied. These may be regarded intuitive categories offering a general assessment and facilitating decision making. Thus, we obtain the following:

$$E \rightarrow \left\{\mu_E^{low}(E),\, \mu_E^{high}(E)\right\}\,. \tag{1}$$

The distance between a sensor node and the base station or the sink node impacts energy consumption and transmission latency. Therefore, nodes located closer to the sink are generally better suited for aggregation due to reduced communication costs and lower risk of packet loss. From a security standpoint, longer distances may increase the exposure of transmitted data to interception or manipulation, thereby necessitating stronger encryption or trust mechanisms.

For the distance input variable D such linguistic terms as "near" and "far" are applied as:

$$D \rightarrow \left\{\mu_D^{near}(D),\, \mu_D^{far}(D)\right\}\,. \tag{2}$$

The traffic load of a sensor node corresponds to the volume of data packets it processes or forwards over a given time interval, which significantly impacts its efficiency in data aggregation operations. High traffic load can lead to increased latency, buffer overflows, and reduced aggregation accuracy due to packet collisions or losses. From a security point of view, excessive traffic may signal potential threats, such as flooding or spoofing attacks.

For the load input variable L such linguistic terms as "small" and "large" are applied in the following way:

$$L \rightarrow \left\{\mu_L^{small}(L),\, \mu_L^{lar}(L)\right\}\,. \tag{3}$$

Node trust is a quantified reliability of a sensor node. High-trust nodes are prioritized to serve as aggregators in order to ensure that the fused data is accurate, timely, and free from manipulation. From a security perspective, incorporating trust evaluation helps mitigate the risks posed by compromised or malicious nodes, leading to enhanced robustness of the aggregation process.

For the trust input variable T such linguistic terms as "low" and "high" are applied as follows:

$$T \rightarrow \left\{\mu_T^{low}(T),\, \mu_T^{high}(T)\right\}\,. \tag{4}$$

Aggregation priority can be regarded as a level of preference assigned to a given sensor node when it comes to performing data aggregation tasks within the wireless network. From a security point of view, this priority should be determined by evaluating factors such as node trustworthiness, energy availability, and exposure to potential threats, ensuring that only reliable and resilient nodes are selected.

Prioritizing secure nodes for aggregation reduces the likelihood of data tampering, spoofing, or compromised fusion, thus enhancing the overall integrity and confidentiality of aggregated information. For the aggregation priority input variable P, we propose to apply 8 linguistic terms: "no priority", "very weak", "weak", "medium weak", "medium", "medium strong", "strong", "very strong". Thus, we obtain the following:

$$\begin{aligned} P \rightarrow \{&\mu_p^{no}(P),\, \mu_p^{wv}(P),\, \mu_p^{v}(P),\, \mu_p^{mv}(P),\\ &\mu_p^{m}(P),\, \mu_p^{ms}(P),\, \mu_p^{s}(P),\, \mu_p^{vs}(P)\}\,. \end{aligned} \tag{5}$$

Fuzzification is the process of transforming crisp input values into degrees of membership by mapping them onto predefined fuzzy sets through membership functions. The trapezoid shape has been selected for membership functions of input values because they are quite simple to use, easy to comprehend, and can help in smooth transitions between different phrases. Thus, the membership functions for the inputs are defined as follows:

For the energy input E, we have:

$$\mu_{low}(E) = \begin{cases} 1 & \text{if} \quad E \leqslant 0.4 \\ \dfrac{0.8-E}{0.8-0.4} & \text{if } 0.4 < E \leqslant 0.8 \\ 0 & \text{if} \quad E > 0.8 \end{cases}\,, \tag{6}$$

$$\mu_{high}(E) = \begin{cases} 0 & \text{if} \quad E \leqslant 0.4 \\ \dfrac{E-0.4}{0.8-0.4} & \text{if } 0.4 < E \leqslant 0.8 \\ 1 & \text{if} \quad E > 0.8 \end{cases}\,, \tag{7}$$

For the distance input D, we have:

$$\mu_{near}(D) = \begin{cases} 1 & \text{if} \quad D \leqslant 0.3 \\ \dfrac{0.7-D}{0.7-0.3} & \text{if } 0.3 < D \leqslant 0.7 \\ 0 & \text{if} \quad D > 0.7 \end{cases}\,, \tag{8}$$

$$\mu_{far}(D) = \begin{cases} 0 & \text{if} \quad D \leqslant 0.3 \\ \dfrac{D-0.3}{0.7-0.3} & \text{if } 0.3 < D \leqslant 0.7 \\ 1 & \text{if} \quad D > 0.7 \end{cases}\,, \tag{9}$$

For the load input L, we have:

$$\mu_{small}(L) = \begin{cases} 1 & \text{if} \quad L \leqslant 0.2 \\ \dfrac{0.6-L}{0.6-0.2} & \text{if } 0.2 < L \leqslant 0.6 \\ 0 & \text{if} \quad L > 0.6 \end{cases}\,, \tag{10}$$

$$\mu_{lar}(L) = \begin{cases} 0 & \text{if} \quad L \leqslant 0.2 \\ \dfrac{L-0.2}{0.6-0.2} & \text{if } 0.2 < L \leqslant 0.6 \\ 1 & \text{if} \quad L > 0.6 \end{cases}\,, \tag{11}$$

For the trust input T, we have:

$$\mu_{low}(T) = \begin{cases} 1 & \text{if} \quad T \leqslant 0.25 \\ \dfrac{0.75-T}{0.75-0.25} & \text{if } 0.25 < T \leqslant 0.75 \\ 0 & \text{if} \quad T > 0.75 \end{cases}\,, \tag{12}$$

$$\mu_{high}(T) = \begin{cases} 0 & \text{if} \quad T \leqslant 0.25 \\ \dfrac{T-0.25}{0.75-0.25} & \text{if } 0.25 < T \leqslant 0.75 \\ 1 & \text{if} \quad T > 0.75 \end{cases}\,, \tag{13}$$

The thresholds used to define the membership functions were analytically selected by the authors. To enhance their accuracy and validity, they will be corrected by means of genetic optimization and neural training.

In this investigation, we propose to utilize a Sugeno-type fuzzy inference system, as it offers computational efficiency and ease of mathematical analysis due to its use of "crisp" singleton outputs and linear functions, making it highly suitable for real-time and embedded applications. Unlike Mamdani FIS, in which a defuzzification phase involves complex centroid calculations, Sugeno FIS produces output through weighted averages, thus resulting in faster and more precise decision-making. Compared to Tsukamoto FIS, which restricts output membership functions to be monotonic and performs rule-by-rule defuzzification, Sugeno FIS provides greater flexibility.

Therefore, the P membership functions of the aggregation priority output are singletons. They are defined as:

$$\mu_{No}(P) = \begin{cases} 1 & \text{if } P = 0 \\ 0 & \text{otherwise} \end{cases} , \tag{14}$$

$$\mu_{vw}(P) = \begin{cases} 1 & \text{if } P = 0.14 \\ 0 & \text{otherwise} \end{cases} , \tag{15}$$

$$\mu_{w}(P) = \begin{cases} 1 & \text{if } P = 0.29 \\ 0 & \text{otherwise} \end{cases} , \tag{16}$$

$$\mu_{mw}(P) = \begin{cases} 1 & \text{if } P = 0.43 \\ 0 & \text{otherwise} \end{cases} , \tag{17}$$

$$\mu_{m}(P) = \begin{cases} 1 & \text{if } P = 0.57 \\ 0 & \text{otherwise} \end{cases} , \tag{18}$$

$$\mu_{sm}(P) = \begin{cases} 1 & \text{if } P = 0.71 \\ 0 & \text{otherwise} \end{cases} , \tag{19}$$

$$\mu_{s}(P) = \begin{cases} 1 & \text{if } P = 0.86 \\ 0 & \text{otherwise} \end{cases} , \tag{20}$$

$$\mu_{vs}(P) = \begin{cases} 1 & \text{if } P = 0.1 \\ 0 & \text{otherwise} \end{cases} , \tag{21}$$

The inference engine in a fuzzy inference system applies logical reasoning to map fuzzified inputs to corresponding fuzzy outputs based on a predefined set of fuzzy rules. It determines the degree to which each rule is activated, and combines their outcomes to generate an aggregated response, thus representing the system's behavior.

In general, a fuzzy rule for our FIS is as follows:

$$\begin{aligned} R^i : \text{if } E = A_E^i \text{ and } D = A_D^i \text{ and } L = L_L^i \\ \text{and } T = A_T^i \text{ and then } P = z^i \ . \end{aligned} \tag{22}$$

The firing strength of a rule is:

$$w^i = \mu_E^{A_E^i}(E) \cdot \mu_D^{A_D^i}(D) \cdot \mu_L^{L_L^i}(L) \cdot \mu_T^{A_T^i}(T) \ . \tag{23}$$

The suggested FIS for data aggregation operates according to an 8-rule base shown in Tab. 1. These rules were deduced after the preliminary calculation performed by the authors. To enhance their accuracy and validity, they will be corrected by neural training. If needed, they can also be corrected while applying the genetic optimization.

Defuzzification is the process of converting the fuzzy reasoning result into a crisp numerical output. Sugeno FIS performs defuzzification through a weighted average of singleton outputs, where weights correspond to the firing strengths of the rules. For this case, we get:

$$P = \frac{\sum\limits_{i=1}^{8} w^i \cdot z^i}{\sum\limits_{i=1}^{8} w^i} \ . \tag{24}$$

However, FISs often lack the adaptive learning capability and they may not be able to adjust to new circumstances, as the rules and membership functions are quite rigid. Overall, this makes FISs less efficient in the dynamically changing environment of modern wireless heterogeneous networks. That is why, in many technical applications, fuzzy inference systems are being integrated with a neural network or genetic algorithms. Inspired by biological neurons, neural networks are computer models that can recognize patterns and relations in real-world data [22].

In this investigation, the authors propose to utilize an adaptive neurofuzzy inference system (ANFIS). ANFIS is classified as a hybrid artificial intelligent system whose characteristics place it between neural network and FIS. Therefore, it combines the benefits of fuzzy logic and those of neural networks, enabling it to learn and adapt its rules to increase accuracy compared to conventional FIS [23].

ANFIS is a more appropriate technique for real-world issues where data patterns are not always easily captured by classical fuzzy rules, since it can handle, unlike regular FIS, complicated and non-linear interactions between input and output values. Mapping between the specified input values and the intended output values is performed using ANFIS training.

Genetic algorithms are evolutionary optimization techniques inspired by natural processes [24]. The combination of a genetic algorithm with the FIS involves optimizing fuzzy rule parameters and membership functions to improve system efficiency. This approach works especially well for complicated issues, allowing the optimized FIS to attain greater accuracy.

The genetic algorithm systematically produces populations of parameter sets, implements selection, crossover, and mutation operations, and progresses towards optimal solutions. It processes encoded parameters of the membership functions and applied evolutionary operators to minimize an objective function and to iteratively search for the optimal solution. This iterative process persists until the convergence criterion

Tab. 1. Fuzzy rules.

	E	D	L	T	P
1	L	F	S	L	VW
2	H	F	S	L	W
3	L	F	L	L	No
4	H	N	L	L	MW
5	L	N	L	H	M
6	H	N	S	H	VS
7	L	F	S	H	MS
8	H	N	L	H	S

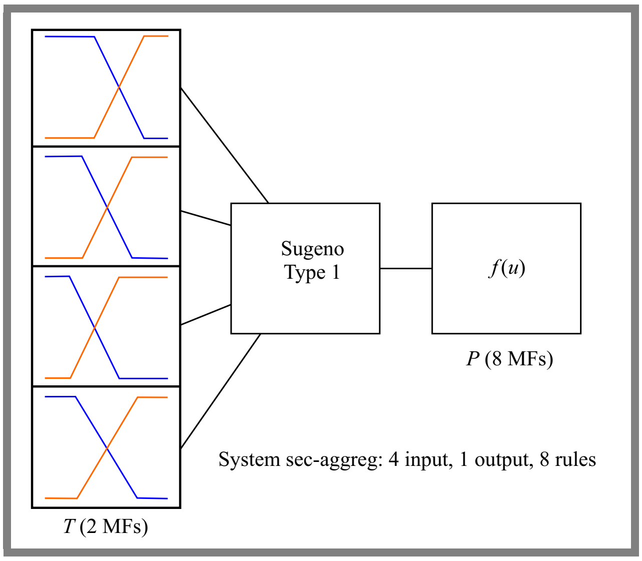


Fig. 3. Fuzzy inference system developed in Matlab.

is satisfied, at which point the most effective parameter set is chosen for the ultimate implementation of the fuzzy inference system.

The objective function is defined as:

$$J(\theta) = \frac{1}{N} \sum_{i=1}^{N} \left[y_i(\theta) - \hat{y}_i \right]^2 , \tag{25}$$

where y_i is an actual output from the FIS for the i-th training sample, $\hat{y}_i$ is a desired output for the i-th training sample, N is a total number of training samples, and θ is a vector of membership function parameters.

5. Simulation

Matlab software can be utilized effectively to validate the functionality of the developed FIS for data aggregation in WSNs. In general, Matlab serves as a comprehensive platform for the visualization and fine-tuning of membership functions, rule bases, and output surfaces, thereby facilitating the development of more accurate and reliable decision-making systems. Simulation of FIS offers key advantages, including rapid prototyping, systematic performance evaluation, and streamlined experimentation. Furthermore, Matlab provides the integration of FIS with machine learning techniques such as neural networks and with optimization toolboxes, which enhance the adaptability and efficiency of fuzzy inference systems, enabling their refinement and deployment in complex real-world scenarios.

The first step was to specify the membership functions for the inputs and the output. Four inputs and one output variables were specified. Figure 3 illustrates the interface of the suggested FIS for data aggregation. Here, the FIS editor outlines the main information about the designed fuzzy inference system.

Then, the rule base was assigned. Next, we assigned the input values and ran the simulation process to produce outputs to check the operability of the developed FIS.

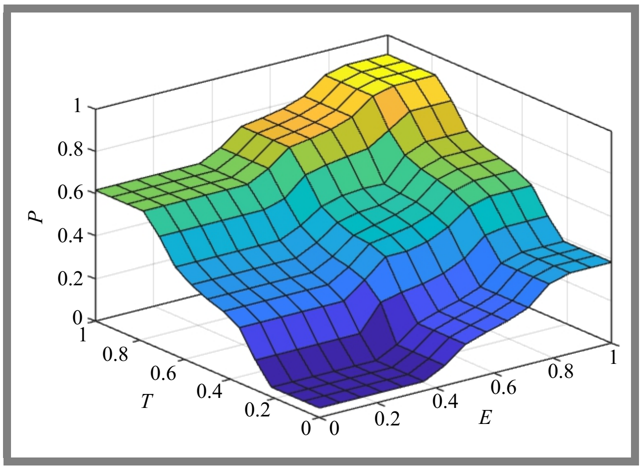
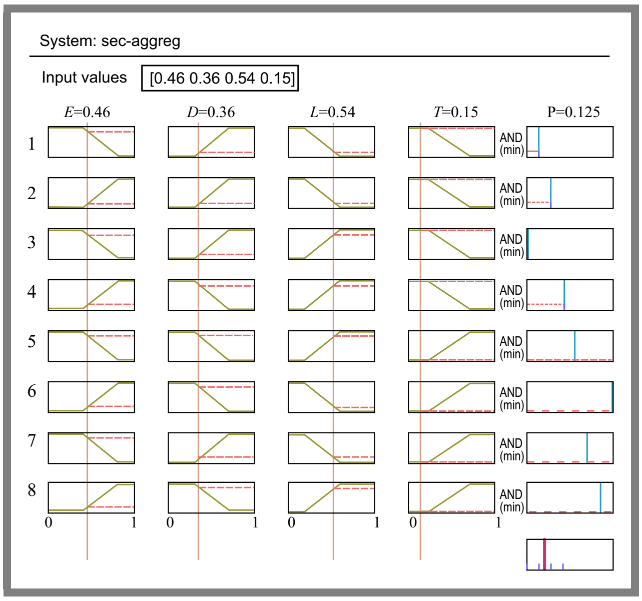


Fig. 4. Visualization of the control surface.

The control surface of an FIS provides a three-dimensional visualization of the system's output behavior based on two selected input variables. In the model developed, the control surface is defined with the x-axis representing the energy level E, the y-axis representing the trust level of node T, and the z-axis representing the resulting aggregation priority P, as shown in Fig. 4. This surface plot illustrates how variations in energy and trust jointly influence the aggregation priority, offering an intuitive understanding of the system's decision-making logic and enabling the evaluation of its responsiveness to changes in critical node parameters.

In the first case, energy value E was 0.46, distance value D was 0.36, load value L was 0.54, and the trust value equaled 0.15. According to Fig. 5, it yielded the aggregation priority of the node equal to 0.215. This means that this sensor node has a very low priority when it comes to choosing it as the aggregation node.

In the second case, energy value E was 0.76, distance value D was 0.22, load value L was 0.77, and the trust value equaled 0.62. According to Fig. 6, it yielded the aggregation priority


System: sec-aggreg

Input values [0.46 0.36 0.54 0.15]

Fig. 5. Simulation results for the first case.

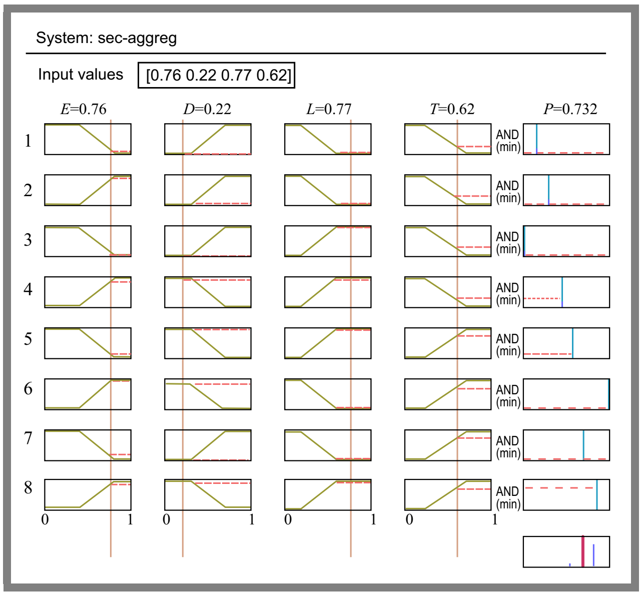


Fig. 6. Simulation results for the second case.

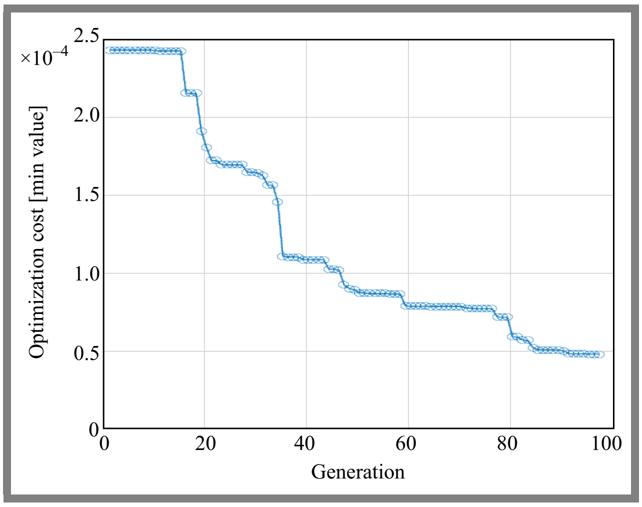


Fig. 7. FIS optimization results showing the training convergence process.

of the node equal to 0.732. This means that this sensor node has quite a high priority when it comes to being selected as the aggregation node.

Next, in this investigation, we utilized a genetic algorithm to optimize the parameters of the FIS developed in Matlab. Overall, the optimization procedure seeks to improve FIS performance by adjusting rule weights and membership functions. As genetic algorithms can handle non-linear search spaces, they are frequently employed as optimization tools. Matlab provides an appropriate environment to combine fuzzy logic systems with various optimization techniques. This is expected to result in increased system stability and control accuracy.

The optimization process involves the establishment of an objective function that assesses the efficacy of a fuzzy inference system according to particular criteria, such as error minimization.

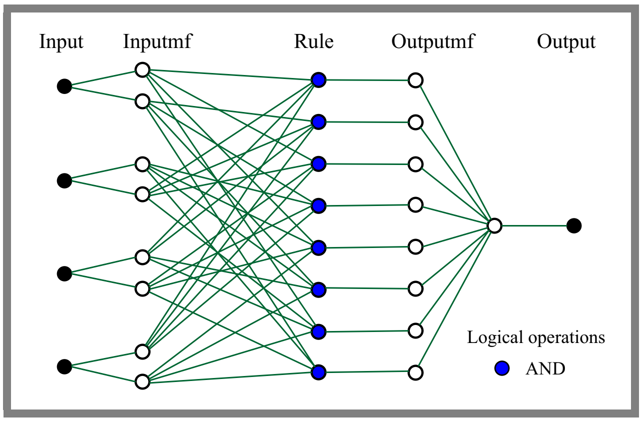


Fig. 8. Structure of the developed ANFIS.

A standard genetic algorithm is employed to optimize the parameters of the FIS. For our case, the optimization specifically targeted membership function parameters, while the rule base structure remained unchanged to preserve expert-defined logic. To perform genetic optimization, a test sample of input and output parameters was utilized.

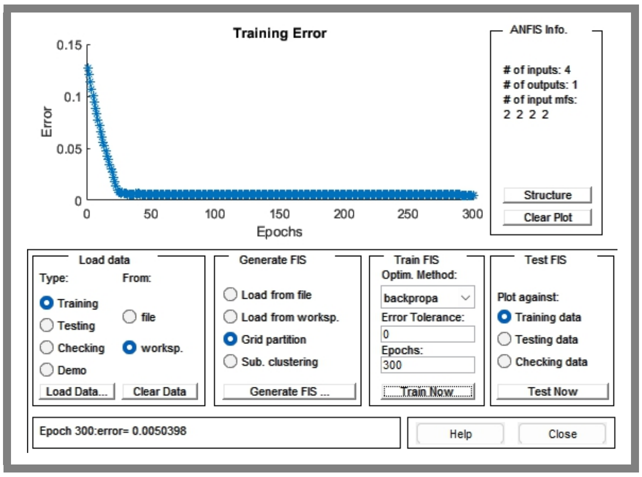
The genetic algorithm converged after 97 iterations, resulting in a refined set of membership functions (Fig. 7). Here, the convergence of the genetic algorithm indicates that the optimization process has successfully reached a stable solution where successive generations no longer produce significant improvements and that the error between the FIS output and the target output is minimized.

This implies that the parameters of the membership functions have been effectively adjusted, thereby enhancing the accuracy of the FIS. Therefore, the simulation result confirmed that the genetic algorithm effectively tuned the parameters of the membership functions.

To further enhance the system's adaptability, the developed FIS was converted into an adaptive neurofuzzy inference system (ANFIS) using the ANFIS edit tool. This transformation enabled the integration of neural network learning capabilities with the interpretability of fuzzy logic, thus allowing the system to automatically adjust its parameters based on training data.

The structure of the generated ANFIS is shown in Fig. 8. The ANFIS model was trained using the backpropagation optimization method which adjusts the parameters of the membership functions by minimizing the error between the predicted and target outputs through gradient descent. Unlike the hybrid method, this approach relies solely on backpropagation to iteratively update both the premise and the consequent parameters, based on the computed error signals. Although computationally more intensive, this method offers full control over the learning process and ensures that all parameters are optimized in a unified framework driven by error minimization.

The ANFIS was trained considering the data from the WSN-DS dataset, specialized for intrusion detection in WSNs. Training the ANFIS optimizes its parameters by iteratively adjusting them to minimize the difference between the predicted and target outputs. Two types of parameters are op-



# of inputs: 4
# of outputs: 1
# of input mfs:
2 2 2 2

Type: ● Training; From: ● worksp.

● Grid partition

Optim. Method: backpropa

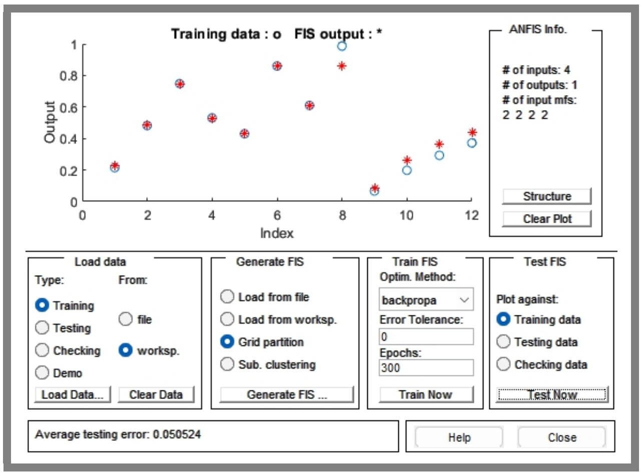
Error Tolerance: 0

Epochs: 300

Plot against: ● Training data

Epoch 300:error= 0.0050398

Fig. 9. Trend of errors in the trained ANFIS.



# of inputs: 4
# of outputs: 1
# of input mfs:
2 2 2 2

Type: ● Training; From: ● worksp.

● Grid partition

Optim. Method: backpropa

Error Tolerance: 0

Epochs: 300

Plot against: ● Training data

Average testing error: 0.050524

Fig. 10. Testing errors.

timized, i.e. the shape of the membership functions and linear function coefficients in the fuzzy rules.

The training persisted for 300 epochs until the error reached a relatively low level. Error assessment was conducted using the root mean square error method. Figure 9 shows this procedure, illustrating the decrease in RMSE in successive epochs, thereby signifying the effective training of the ANFIS model. Figure 10 presents the testing error represented by asterisks and the training error indicated by dots.

A decrease in training error signifies a progressive improvement in parameter adjustment. This decrease indicates that the ANFIS has effectively learned from the training data, resulting in optimized membership functions and rule parameters.

The simulation results support the relevance of the application of the proposed hybrid optimized fuzzy inference system in WSNs with security issues.

The combination of fuzzy logic with genetic algorithm optimization methods and neural network learning processes emphasizes the significant potential of the proposed approach for being applied in highly dynamic and resource-limited wireless sensor network environments.

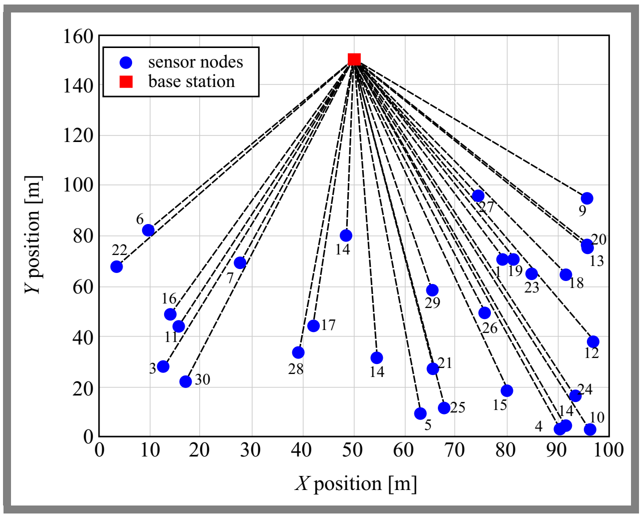


Fig. 11. Structure of the wireless sensor network tested.

6. Comparative Analysis

To perform a comparative analysis of the proposed data aggregation technique with a classic approach, a simulation in Matlab was performed for the wireless sensor network. The simulated WSN consists of 30 sensor nodes which are distributed within a 100 m × 100 m area. A base station is located in the monitored region at coordinates (50, 150). This corresponds to a practical deployment scenario, since the base station is located in a more accessible location for data collection and processing. The simulated WSN is shown in Fig. 11.

Here, each sensor node is characterized by four parameters: residual energy, distance to the base station, traffic load, and trust level. To model security threats, 20% of the nodes were randomly designated as malicious with low trust values of 0.1 ... 0.3, representing potential threats such as packet dropping or data manipulation.

This simulated topology served as a basis for comparing two data aggregation approaches: the proposed intelligent secure aggregation and a simple energy-based aggregation. The latter is a base approach that selects aggregator nodes solely based on their residual energy levels, ignoring other factors.

The simulation results shown in Fig. 12 demonstrate how the intelligent secure aggregation method performs compared to the simple energy-based aggregation approach.

Figure 12a refers to intelligent secure aggregation. The aggregator nodes (red stars) are well-distributed, avoiding malicious nodes (black circles). The color gradient shows the priority of the aggregation, with the selected nodes having high scores. Figure 12b refers to the simple aggregation method. Here, the aggregator nodes (green stars) may overlap with malicious nodes. Color shows the energy levels only, and the selection ignores such factors as trust and load levels.

Figure 13 illustrates the comparison of data aggregation methods in the form of bar graphs. Thus, the visualization of results confirms that by avoiding malicious nodes and balancing the load, the proposed intelligent scheme distributes aggregation

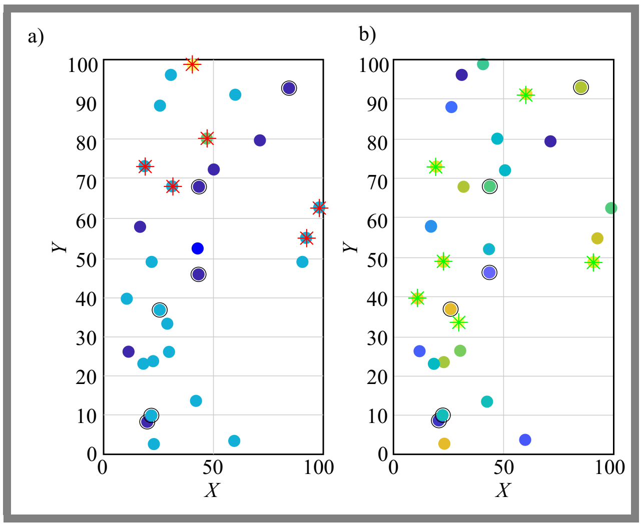


Fig. 12. Comparison of: a) fuzzy-based secure aggregation and b) simple energy-based aggregation.

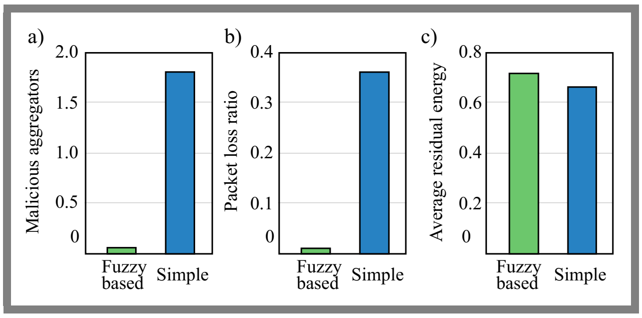


Fig. 13. Comparison of data aggregation methods in terms of: a) security (malicious node selection), b) reliability (packet loss) and c) energy efficiency.

tasks more evenly, reduces packet loss, and extends network lifetime.

The simulation results demonstrate a significant advantage of the intelligent approach. On average, the proposed method selected 0.06 malicious aggregator nodes per run, leading to a packet loss of 1%, while the simple method selected 1.80 malicious aggregators, causing approximately 36% of packets to be lost. Furthermore, the intelligent approach exhibited better energy efficiency, with an average residual energy of 0.71, compared to 0.66 in the simple aggregation scheme.

7. Conclusions

Data aggregation contributes to minimizing the volume of message transmissions within a wireless sensor network, thereby lowering overall energy consumption. To address this challenge, the proposed system implements an AI approach for selecting an optimal node based on its proximity to the sink, its available resources, and its trust (security). The selection of an energy-efficient node minimizes energy usage across the wireless sensor network, thus contributing to an extended network lifetime. The designated node is responsible for collecting and aggregating data from all member nodes within the cluster. Since a malicious node cannot be chosen as data aggregators, secure data aggregation is ensured in the WSN.

The fuzzy inference system developed in the Matlab environment was designed to assess the suitability of a given sensor node for participation in the data aggregation process within a wireless sensor network. The FIS operates based on a real-time evaluation of four key parameters: residual energy, distance from the base station, node load, and trust level.

These parameters serve as input to the FIS, thus enabling both adaptive and context-aware decision making under dynamic conditions of a wireless network. To improve the accuracy of the system, the membership functions of the FIS were optimized using a genetic algorithm, while preserving the original rule base. This evolutionary optimization approach allowed for fine-tuning of the fuzzy model to better reflect the non-linearities of WSN parameters.

Subsequently, the developed FIS was transformed into an ANFIS which can learn from data and autonomously adjust its parameters in response to changes in the network. The ANFIS model was trained in Matlab using the backpropagation optimization method, which iteratively minimized the output error by adjusting the parameters of the membership functions through gradient descent. This learning capability enhanced the responsiveness and robustness of the solution.

The authors claim that this study offer a contribution in the form of the considered methodology for developing genetic neuro-fuzzy inference systems which can be further utilized for designing real hybrid intelligent solutions to be implemented, for various purposes, in wireless sensor networks.

The authors admit, however, that this study lacks experimental testing which will be the subject of continued investigations.

In addition, other network parameters can be taken into account to enhance the potential of the developed FIS. In future inquiries, the data aggregation mechanism used in WSNs will be improved by implementing other AI techniques as well.

References

[1] L. Obaid *et al.*, "Challenges of Wireless Sensor Networks and Their Solutions", *International Journal of Computers and Informatics*, vol. 3, pp. 102–129, 2024 (https://doi.org/10.59992/IJCI.2024.v3n10p3).

[2] N. Kaur and D. Vetrithangam, "Routing and Data Aggregation Techniques in Wireless Sensor Networks: Previous Research and Future Scope", *Studies in Autonomic, Data-driven and Industrial Computing*, pp. 705–718, 2024 (https://doi.org/10.1007/978-981-99-5435-3_51).

[3] D.N. Ajobiewe, "Data Aggregation in Wireless Sensor Networks: Emerging Research Areas", *Journal of Mathematical Sciences and Computational Mathematics*, vol. 3, pp. 88–101, 2021.

[4] S.A. Abdulzahra and A.K.M. Al-Qurabat, "Data Aggregation Mechanisms in Wireless Sensor Networks of IoT: A Survey", *International Journal of Computing and Digital Systems*, vol. 13, pp. 1–15, 2023.

[5] I.D.I. Saeedi and A.K.M. Al-Qurabat, "A Systematic Review of Data Aggregation Techniques in Wireless Sensor Networks", *Journal of Physics: Conference Series*, vol. 1818, art. no. 012194, 2021 (https://doi.org/10.1088/1742-6596/1818/1/012194).

[6] D. Kandris and E. Anastasiadis, "Advanced Wireless Sensor Networks: Applications, Challenges and Research Trends", *Electronics*, vol. 13,

art. no. 2268, 2024 (https://doi.org/10.3390/electronics13122268).

[7] N.R. Roy and P. Chandra, "Analysis of Data Aggregation Techniques in WSN", *Advances in Intelligent Systems and Computing*, vol. 1059, pp. 571–581, 2019 (https://doi.org/10.1007/978-981-15-0324-5_48).

[8] K.K. Sarma, "Application of Soft Computing Tools in Wireless Communication – A Review", in: *Signals and Communication Technology*, Springer, India, pp. 197–207, 2015 (https://doi.org/10.1007/978-81-322-2407-5_16).

[9] W. Osamy *et al.*, "Recent Studies Utilizing Artificial Intelligence Techniques for Solving Data Collection, Aggregation and Dissemination Challenges in Wireless Sensor Networks: A Review", *Electronics*, vol. 11, art. no. 313, 2022 (https://doi.org/10.3390/electronics11030313).

[10] R.V. Kulkarni, A. Forster, and G.K. Venayagamoorthy, "Computational Intelligence in Wireless Sensor Networks: A Survey", *IEEE Communications Surveys & Tutorials*, vol. 13, pp. 68–96, 2011 (https://doi.org/10.1109/surv.2011.040310.00002).

[11] S. Reshma, K. Shaila, and K.R. Venugopal, "Maximizing Network Lifetime using Fuzzy Based Secure Data Aggregation Protocol (FSDAP) in a Wireless Sensor Networks", *International Journal of Recent Technology and Engineering*, vol. 8, pp. 5989–6001, 2019 (https://doi.org/10.35940/ijrte.C4559.118419).

[12] S. Bhushan *et al.*, "FAJIT: A Fuzzy-based Data Aggregation Technique for Energy Efficiency in Wireless Sensor Network", *Complex and Intelligent Systems*, vol. 7, pp. 997–1007, 2021 (https://doi.org/10.1007/s40747-020-00258-w).

[13] R. Wan *et al.*, "Similarity-aware Data Aggregation Using Fuzzy C-means Approach for Wireless Sensor Networks", *Journal on Wireless Communications and Networking*, vol. 2019, art. no. 59, 2019 (https://doi.org/10.1186/s13638-019-1374-8).

[14] J. Qin, W. Fu, H. Gao, and W.X. Zheng, "Distributed K-means Algorithm and Fuzzy C-means Algorithm for Sensor Networks Based on Multiagent Consensus Theory", *IEEE Transactions on Cybernetics*, vol. 47, pp. 772–783, 2017 (https://doi.org/10.1109/TCYB.2016.2526683).

[15] H. Zhou and K. Yu, "A Novel Wireless Sensor Network Data Aggregation Algorithm Based on Self-organizing Feature Mapping Neutral Network", *Ingénierie Des Systèmes d'Information*, vol. 24, pp. 119–123, 2019 (https://doi.org/10.18280/isi.240118).

[16] N. Kaur and D. Vetrithangam, "Energy Efficient Data Aggregation in Wireless Sensor Networks Using Meta Heuristic Based Feed Forward Back Propagation Neural Network Approach", *Journal of Machine and Computing*, vol. 4, pp. 651–660, 2024 (https://doi.org/10.53759/7669/jmc202404062).

[17] F. Khorasani and H.R. Naji, "Energy Efficient Data Aggregation in Wireless Sensor Networks Using Neural Networks", *International Journal of Sensor Networks*, vol. 24, art. no. 26, 2017 (https://doi.org/10.1504/IJSNET.2017.084207).

[18] R. Kowsalya and B.R. Jeetha, "CDARGA: Cluster-based Data Aggregation with Genetic Routing Algorithm in Wireless Sensor Networks", *International Journal of Recent Technology and Engineering*, vol. 8, pp. 2976–2982, 2020 (https://doi.org/10.35940/ijrte.F8443.038620).

[19] S. Sharmin, I. Ahmedy, R.M. Noor, and H. Ismail, "Using Hybrid Genetic Algorithm for Data Aggregation in Wireless Sensor Networks", *18th International Conference on Ubiquitous Information Management and Communication (IMCOM)*, Kuala Lumpur, Malaysia, 2024 (https://doi.org/10.1109/IMCOM60618.2024.10418358).

[20] H. Kumar and P.K. Singh, "Comparison and Analysis on Artificial Intelligence Based Data Aggregation Techniques in Wireless Sensor Networks", *Procedia Computer Science*, vol. 132, pp. 498–506, 2018 (https://doi.org/10.1016/j.procs.2018.05.002).

[21] R. Saatchi, "Fuzzy Logic Concepts, Developments and Implementation", *Information*, vol. 15, art. no. 656, 2024 (https://doi.org/10.3390/info15100656).

[22] M. Islam, G. Chen, and S. Jin, "An Overview of Neural Network", *American Journal of Neural Networks and Applications*, vol. 5, pp. 7–11, 2019 (https://doi.org/10.11648/j.ajnna.20190501.12).

[23] T.S. Ogedengbe *et al.*, "An Overview of Neural Networks, Fuzzy Systems and Neuro-fuzzy Systems", *AIP Conference Proceedings*, vol. 3007, art. no. 100017, 2024 (https://doi.org/10.1063/5.0197104).

[24] R.R. Mohsin, "Genetic Algorithm: A Study Survey", *Iraqi Journal of Science*, vol. 63, pp. 1215–1231, 2022 (https://doi.org/10.24996/ijs.2022.63.3.2).

Olena Semenova, Ph.D.
Department of Infocommunication Systems and Technologies
https://orcid.org/0000-0001-5312-9148
E-mail: semenova.o.o@vntu.edu.ua
Vinnytsia National Technical University, Vinnytsia, Ukraine
https://vntu.edu.ua

Natalia Kryvinska, Ph.D.
Department of Information Management and Business Systems
https://orcid.org/0000-0003-3678-9229
E-mail: natalia.kryvinska@fm.uniba.sk
Comenius University in Bratislava, Bratislava, Slovakia
https://uniba.sk

Serhii Baraban, Ph.D.
Department of Data Processing Technologies
https://orcid.org/0000-0001-9535-1644
E-mail: serhii.baraban@put.poznan.pl
Poznan University of Technology, Poznan, Poland
https://put.poznan.pl

Maksym Prytula, Ph.D.
Department of Information Radioelectronic Technologies and Systems
https://orcid.org/0000-0003-1577-5215
E-mail: prytula@vntu.edu.ua
Vinnytsia National Technical University, Vinnytsia, Ukraine
https://vntu.edu.ua

Volodymyr Martyniuk, M.Sc.
Department of Infocommunication Systems and Technologies
https://orcid.org/0009-0006-8421-0348
E-mail: vm4ukr@gmail.com
Vinnytsia National Technical University, Vinnytsia, Ukraine
https://vntu.edu.ua