BULLETIN OF THE POLISH ACADEMY OF SCIENCES
TECHNICAL SCIENCES, Vol. 72(1), 2024, Article number: e147919
DOI: 10.24425/bpasts.2023.147919

SPECIAL SECTION

Multifunctional clustering based on the LEACH algorithm for edge-cloud continuum ecosystem

Andrzej PASZKIEWICZ[1]*, **Cezary ĆWIKŁA**[2], **Marek BOLANOWSKI**[1], **Maria GANZHA**[3], **Marcin PAPRZYCKI**[3], and **Michal HODOŇ**[4]

[1] Department of Complex Systems, Rzeszow University of Technology, Al. Powstańców Warszawy 12, Rzeszów 35-959, Poland
[2] Rzeszow University of Technology, Al. Powstańców Warszawy 12, Rzeszów 35-959, Poland
[3] Systems Research Institute Polish Academy of Sciences, Newelska 6, Warszawa 01-447, Poland
[4] Department of Technical Cybernetics, University of Žilina, Univerzitná 8215/1, 010 26 Žilina, Slovakia

Abstract. This paper introduces a novel approach to building network cluster structures, based on the modified LEACH algorithm. The proposed solution takes into account the multitasking of the network infrastructure, resulting from various functions performed by individual nodes. Therefore, instead of a single head, dedicated to a given cluster, a set of heads is selected, the number of which corresponds to the number of performed functions. Outcomes of simulations, comparing the classical and the multifunctional approach, are presented. The obtained results confirm that both algorithms deliver similar levels of energy consumption, as well as efficiency in terms of the number of individual nodes discharged.

Keywords: LEACH, IoT; cluster; sensors; phase transition.

1. INTRODUCTION

Over the years, data collection and processing systems have been constantly evolving. For instance, with the development of the Internet of Things (IoT) networks, it was necessary to create new methods and means of data processing, new architectures, and communication protocols. Consequently, edge computing has been proposed, to augment cloud-based systems [1–5].

However, taking into account the possibility of distributed data processing at various levels of (and in various locations within) the IoT ecosystem, work began on a novel approach, called *edge-cloud continuum* [6, 7]. It assumes that different functionalities can be implemented at different system levels (and/or in various locations). In this way, the efficiency and scalability of distributed IoT systems can be increased. It is easy to observe that this approach can be seen as a part of complex systems theory [8, 9], which captures interactions between multiple interacting subsystems. Thanks to this, the value of a complex system is greater than the sum of the values of its independent components.

Complex systems usually exhibit a modular organization, associated with strong relationships existing between groups of vertices. This may be the result of having common features, or roles, and functions performed in the system. This property is called community structure, or clustering. Clustering can take place at various levels of a distributed system, as well as between levels. Moreover, in the case of exceptionally large, geographically distributed, loosely coupled systems, clustering can be a "local phenomenon", concerning a set of closely connected nodes. Indeed, there are many examples of distributed networks. These can be sensor networks applied in industrial systems [10], weather control systems [11], measurement systems used in agricultural production [12], as well as general-purpose sensor systems that perform various functions, depending on current needs, thanks to the ability to connect multiple heterogeneous sensors to a given local node. Wireless body area networks (WBAN) systems [13], which ensure data collection and transmission within the network of sensors monitoring vital functions, efficiency, etc. of the human body, are also extremely interesting.

From the perspective of the communication infrastructure, dedicated to the above-mentioned systems, an important aspect is the development of solutions that can adapt to the changing conditions, in order to provide the requested functionality. An example of such a mechanism is the LEACH (low-energy adaptive clustering hierarchy) algorithm, which was developed for information routing in homogeneous sensor networks [14]. Its main purpose is to reduce the total amount of information, sent to the base station. Therefore, clusters are created, within which *heads* are determined, i.e. nodes aggregating network traffic, which is then redirected to the base station. Obviously, this concept is in line with the assumptions of the edge-cloud continuum computing where information is processed as close as possible to the data source to realize user-defined workflows. With this approach, locally collected data can be pre-processed, filtered, cleaned, normalized, as well as compressed, sampled, etc. Obviously, the scope of undertaken activities depends on the computing power and power resources of individual nodes, among others.

*e-mail: andrzejp@prz.edu.pl

Manuscript submitted 2023-02-25, revised 2023-09-07, initially accepted for publication 2023-10-08, published in February 2024.

Taking into account the resources of wireless network nodes, which are often extremely limited, the LEACH algorithm assumes cyclical (in rounds) rotation of nodes performing the information flow aggregation. Here, the guiding assumption is that this can balance the power consumption of the individual nodes, thus extending the usability of the whole ecosystem. Over time, many improvements to the LEACH algorithm have been proposed. However, the question of multitasking of individual nodes, and the possibility of them simultaneously belonging to different clusters has been omitted. Nevertheless, it should be noted that, as can be seen in [15], the use of modular gateway nodes, like the GWEN (gateway edge node), may lead to edge-cloud continuum deployments, in which individual nodes may be assembled to deliver a precisely defined set of functions. Moreover, such nodes may actually belong to different clusters. For instance, on a construction site, they may interact with multiple beacons to provide geo-localization-related functions, while in another context they may be involved in processing and routing data devoted to worker health monitoring. Taking this into account, the purpose of this contribution is to extend the LEACH-based approach to deliver similar advantages, while recognizing the potential need to support functional heterogeneity. Thus, a new solution was developed to facilitate multitasking and multifunctional operation of individual nodes. As part of the validation of the proposed approach, a simulation-based study was conducted to determine the impact of functional changes on the energy efficiency of the nodes.

2. RELATED WORK

There are many different approaches to clustering, including those based on K-Means algorithms, complex systems theory, fuzzy logic clustering, and many others. However, this paper focuses on the developments related to the LEACH algorithm.

The basic LEACH algorithm provides routing in homogeneous sensor networks [14]. For this purpose, it uses clusters responsible for aggregating data, sent from the local nodes. Another approach is represented by the LEACH-C algorithm [16], which is based on the centralized supervision. The base station collects information about the energy level of individual nodes. Only nodes with above-average energy, in a given round, are eligible to be cluster heads. Based on this criterion, the base station creates clusters using simulated annealing to minimize the amount of energy needed to send data to the cluster head (by the individual nodes). The opposite approach is represented by the LEACH-B algorithm [17]. In this algorithm, individual nodes have knowledge of the energy levels of the other nodes, and the cluster head is selected on this basis. The algorithm also assumes a fixed number of heads, based on the desired percentage of heads and the total number of nodes. If the number of heads falls below a certain value, then the regular node with the highest energy has the highest probability of being promoted to the cluster head. Another improvement of the LEACH algorithm is Energy-LEACH, which streamlines the cluster head selection procedure [18]. The main metric is the node residual energy, which determines whether a node will become the cluster head after the first round. Compared to LEACH, this approach provides longer network life and greater energy savings. One of the fundamental problems of many communication mechanisms in distributed sensor networks, including the LEACH family of algorithms, is the faster energy depletion of the nodes that aggregate the data, which is then sent to the base station. For this reason, a new version of the LEACH algorithm, called TL-LEACH, was proposed in [19]. The key change is the use of two-level clusters to transmit data to the base station, which results in a reduction in the transmission distance. Thanks to this, fewer nodes must transmit data to the base station over longer distances, which is especially important in networks characterized by high node density. Here, it is noteworthy that in this way the edge-cloud continuum concept can be implemented relatively effectively as, for instance, part of the local calculations can be performed at the second level of the cluster heads. An algorithm similar to TL-LEACH is MH-LEACH [20], which improves the communication between the cluster head and the base station by using multi-hop transmission. In the LEACH algorithm, each cluster head communicates directly with the base station, regardless of the distance between them, so if the distance is large, it will consume more energy. Hence, MH-LEACH assumes the optimal path, which on the way from the cluster head to the base station contains other cluster heads as relay stations to transfer data through them. The next interesting algorithm is ACHTH-LEACH [21], which improves the performance of LEACH by using greedy k-means. All nodes that are recognized as located near the base station form one cluster, while the greedy k-means algorithm is used for the remaining nodes to form clusters while ensuring that the number of nodes in the clusters is similar. To evenly distribute the energy among the nodes, the node with the highest remaining energy becomes the head of the cluster. This algorithm uses two-hop transmission to avoid unnecessary power consumption. Addressing slightly different concerns, in order to ensure clustering in large wireless sensor networks, the size of which is much larger than the transmission range of a single sensor, the MELEACH-L algorithm was developed [22]. It is a low-power multi-channel routing algorithm that manages channel allocation between neighboring clusters and cooperation between cluster heads during data collection by controlling the size of each cluster and by separating cluster heads from the backbone nodes. There are also LEACH DCHS and LEACH-DHCS-CM algorithms, which periodically format the cluster in a steady state phase. The latter was shown as especially resulting in the reduction of energy consumption when transmitting the same amount of data. Separately, an approach in which, instead of competition for being the head of the cluster based on random numbers, a random time interval is used was proposed in [23]. In this case, the nodes that have the shortest time interval become cluster heads. Another approach is ALEACH, which uses a distributed algorithm to create clusters, thanks to which nodes make autonomous decisions without any centralized control [24]. In this approach, the number of nodes per cluster is highly variable, and the amount of data each node can send to the cluster head varies with the number of nodes in the cluster. Another approach is to change the structure of the network by using fiber optic links. The hybrid sensor network topology proposed in the OLEACH algorithm [25] consists of distributed sensor links, located in the center and two separate wireless sensor networks with randomly placed nodes. Its applications include military, industrial, and energy links and tunnels. Another algorithm, with a high level of security, is Armor-LEACH [26]. It ensures authenticity, confidentiality, integrity, and optimal cluster size, reduces scheduling complexity, and evenly distributes power consumption among nodes. Due to the specific requirements, in mobility-oriented environments, an improvement in the form of the LEACH-Mobile algorithm has been suggested [27]. This solution pays particular attention to the mobility factor, which is of the greatest importance when choosing a cluster head. The opposite approach to mobile LEACH is Recluster-LEACH [28]. This algorithm improves the functioning of the LEACH algorithm by improving the mechanism of cluster head selection in the first phase, the fusion of clusters in high-density areas and nodes, using multi-hop routing. There is also the MR-LEACH algorithm [29]. The motivation for its creation was to reduce energy consumption by adaptively increasing the clustering hierarchy. In order to create a predetermined number of clusters, the base station helps to determine the clustering hierarchy and imposes a TDMA schedule for each layer of cluster heads. In the LEACH-HPR algorithm [30], the heterogeneous network of sensors divides the nodes into three groups, based on the energy capacity of the nodes. One can also distinguish the LEACH-V version of the algorithm [31]. Here, clusters consist of a cluster head, an alternate cluster head, and other cluster nodes. The alternate cluster head takes over the function of the primary cluster head when it is discharged. This approach increases the stability and reliability of the sensor network. Next, [32], an energy-efficient and trust-aware framework for secure routing EETA-LEACH is proposed. Here, improvements over LEACH consist of introducing trust to provide secure routing, while maintaining the originality of LEACH protocol.

As can be seen, none of the above-mentioned approaches takes into account the potential multifunctional nature of sensor networks. However, this is reasonable as such requirements are relatively new and directly related to the instantiation of large IoT ecosystems. Therefore, this contribution proposes a solution that meets the heterogeneous structures in terms of the functionality of the sensor network that fits into the concept of edge-cloud continuum processing.

3. LEACH ALGORITHM

3.1. Phases in the basic LEACH algorithm

The LEACH algorithm is based on a cyclical change of roles, performed by nodes in a given cluster. Each cluster has nodes that measure and possibly pre-process the collected data, as well as head nodes that act as an aggregator. In order to be able to change the roles cyclically, the operation of the algorithm is based on the configuration phase and the steady state phase. This basic concept is represented in Fig. 1 in the form of a spiral. The spiral directly illustrates the idea that after the configuration phase, during which a given cluster head is selected, there is a steady state phase, the time of which is defined by the assumed number of measurement cycles. After conducting the measurements and sending the processed data to the base station, the configuration (reconfiguration) phase takes place (again), in which a new head in the cluster is (most likely) selected and new clusters may be created, if necessary.

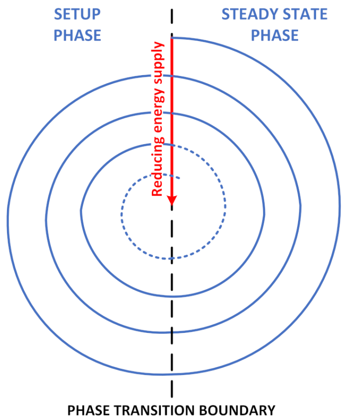


Fig. 1. Phase cyclicity in the LEACH algorithm

In this way, the algorithm contributes to balancing the energy consumption of individual nodes. Obviously, the LEACH cycle can be repeated until all nodes are discharged. This condition is illustrated in Fig. 1 as a red arrow.

3.2. Fundamentals of the algorithm

Let N denote a set of nodes n. Then, in the configuration phase, the cluster heads selection process is started. For this purpose, a random value between 0 and 1 is selected for each node. Next, it is compared with the threshold value $T(n)$ [33]:

$$T(n) = \begin{cases} \dfrac{p}{1 - p * \left(r \mod \dfrac{1}{p}\right)} & \text{if } n \in G, \\ 0 & \text{otherwise,} \end{cases} \tag{1}$$

where: p – the desired percentage of the number of cluster heads in the network (e.g., 0.05), r – current round number (starting with round 0), G – set of nodes that did not act as heads in the last $1/p$ number of rounds, thanks to this, each node will become the head of the cluster within $1/p$ rounds.

If the value drawn for a given node n is lower than the threshold value, then it becomes the head of the cluster, otherwise, node n is connected to the nearest cluster in its vicinity. After sending a certain number of packets related to measurement cycles, the configuration phase is repeated.

4. THE PROPOSED APPROACH

4.1. Multitasking model

The basic LEACH algorithm itself has some limitations that were eliminated by introducing further improvements and changes in the solutions proposed later (see Section 2). However, it was mostly assumed that the system is homogeneous from the point of view of task execution. On the other hand, modern practical applications force openness and universality of systems in order to perform various functions and tasks. Therefore, let us assume that within a given wireless network a set of a finite number of functions is defined as $F = \{f_1, f_2, \ldots, f_i;\ i \in \mathbb{N}\}$. Each of the nodes can perform a set of specific functions, i.e. $F(n) = \left\{f_1^n, f_2^n, \ldots, f_j^n;\ j \leq |F|\right\}$. They can also perform different tasks assigned to different functions. Typically, these tasks are associated with specific sensors and their corresponding measurements. The classic LEACH algorithm does not take this situation into account. It treats the network as a homogeneous structure, where data is sent 'equally' by all individual nodes. Then, only at the level of heads, it aggregates the traffic directed to the base station. Such an approach is justified in the case of single measurements. This variant is shown in Fig. 2a, where member nodes of a given cluster send network traffic (measurements) from different sensors to the one head, selected for this cluster. However, when many measurements are made, the head load can be extremely high. At the same time, it should be remembered that individual nodes, aggregating the measurement traffic at a given moment, may be characterized by limited power resources, for instance, while they may be built of reconfigurable resources whose computational efficiency is adapted to the specific calculations. Looking at it from the perspective of the edge-cloud continuum concept, it should be assumed that the temporary, or permanent, specialization of individual nodes may play a key role in the heterogeneous environments. Therefore, it is proposed that an independent head should (or, at least, might be) be selected for each functionality at each stage of the election process (Fig. 2b). This approach is part of the attempt to even the distribution of the load on individual elements of the network infrastructure. It should be relatively obvious that this approach is preferable for a larger group of nodes.

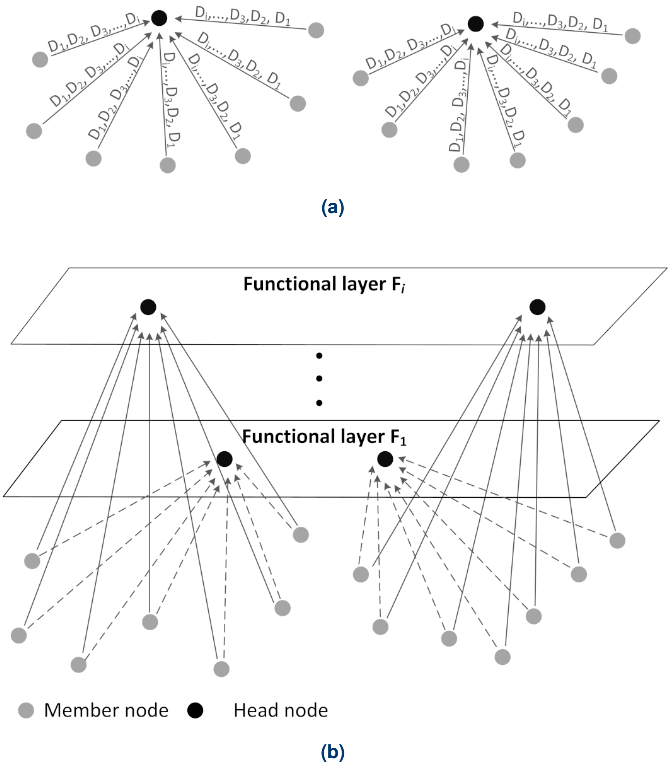


Fig. 2. Communication structure: (a) single head approach for all functionalities; (b) multi-task approach considering many different heads within a given cluster

Multitasking can also affect the rate of energy consumption of individual sensor nodes. At this point, it should be noted that the performance of the LEACH algorithm refers to a situation in which all nodes have an initial equal energy E_0. The energy consumption of the network is dependent on data transmission, reception, and connection processes, and can vary depending on the distance between the nodes. Here, it should be recalled that sensor nodes die when fully discharged. An additional factor affecting the problem of ensuring efficient energy use is the inability to charge the batteries of individual nodes. Such a situation occurs, for instance, in all cases when nodes have to be 'sealed' to protect them from adverse effects of the environment. Thus, the level of available energy at individual nodes is one of the key aspects determining the ability of a given network infrastructure to perform individual functionalities for an extended period of time.

4.2. Algorithm

The adoption of the assumption regarding the dispersion of tasks for heads results in a change in the structure of the LEACH algorithm. The general diagram of the modified algorithm is shown in Fig. 3. The input data for the algorithm consist of area dimensions, number of nodes, number of performed functions, initial energy of nodes, energies, energy lost during cluster head selection, etc. Initially, the $G = N$ set includes all available nodes. Next, the heads are selected, in accordance with the adopted proportions, expressed in the x parameter. The value of $x$is directly affected by, for example, the characteristics of a given network. It is also influenced by the modified rule (1), which has the form:

$$T(n) = \begin{cases} \dfrac{p * |F|}{1 - p * |F| * \left(r \mod \dfrac{1}{p * |F|}\right)} & \text{if } n \in G, \\ [8pt]0 & \text{otherwise.} \end{cases} \tag{2}$$

As a result, the set of heads consists of subsets of heads for each of the functionalities belonging to a c-th cluster $G^r = \left\{G_c^r\right\}$. Thus, the size of the set of heads in the r-th round is $|G^r| = |F| * x * |N|$. The determination of the membership of individual subsets of G_c^r can be conducted in numerous ways, e.g. by the criterion of the minimum distance between individual heads. In the next step, the heads used in a given round are removed from the set G.

Multidimensional clusters are then created to allow data from member nodes to be sent to a dedicated set of heads, established for the cluster. The versatility of the presented algorithm means that any mechanism can be used to assign a given member node to a given set of heads. Here, the explored approach is based on taking into account the criterion of energy savings. Hence, the shortest path to the center, between all heads in a given cluster, is chosen. These activities are followed by a period of measuring

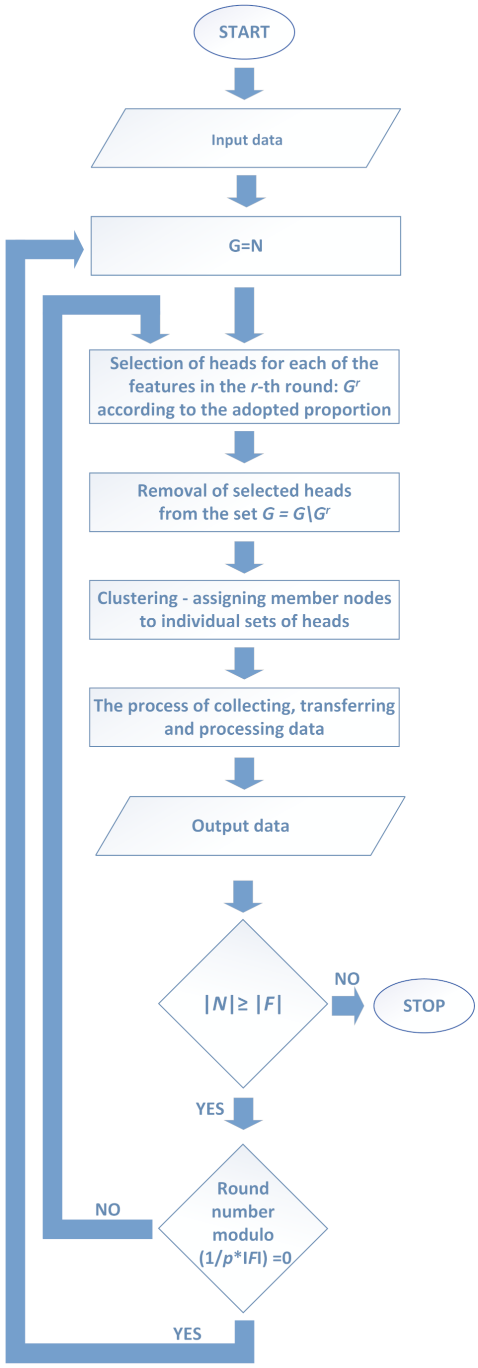


Fig. 3. An algorithm that takes into account multifunctionality

and transmitting data from the member nodes to the heads. After its completion, a condition is checked, which determines whether the available nodes could become the cluster head. Of course, the presented proposal takes into account the conditions related to the number of heads being adequate to the set of supported functionalities. If this condition is met, another series of rounds of the algorithm begins, in which nodes can become heads again. Otherwise, the next round within the same series is implemented. This pattern is repeated until $|N| \geq |F|+1$. The main outputs of the algorithm are multidimensional clusters with a varying number of nodes, variable cluster heads for each supported function, and remaining node energy. The algorithm stops its operation when the number of remaining nodes is less than necessary to perform the function (i.e. to support multi-functionality).

5. EXPERIMENTAL EVALUATION

5.1. Network structure

In order to conduct the research, a Python-based simulation environment, consisting of 100 nodes was developed. The base station is placed in the center of the area and its energy is assumed to be inexhaustible. In the case of the LEACH mechanism, the probability p of selecting a node for a cluster head is 5%, according to research reported in [15]. This work also states that the most energy-efficient system has 3 to 5 clusters for 100 nodes in the network. This means that the optimal percentage of heads is between 3% and 5%. If a node is not within range of any cluster head, it sends data straight to the base station. It was assumed that the location of the nodes cannot change after they are deployed (i.e. sensor mobility was excluded). Sensor nodes die when their energy is exhausted. The communication mode between the nodes adopts the single-hop transmission. For the initial transmission, when hello packets are sent, the packet size is 100 bytes, while when data packets are sent, the data packet size is 4000 bytes.

5.2. Energy consumption

Network energy consumption is calculated based on data transmission, reception, and connection processes, and may vary depending on the node distance. Each node sending data is charged with the energy cost of data transmission marked as E_T and is calculated according to the following formula (based on [34]):

$$E_T = \begin{cases} E_{TX} * L + E_{mp} * L * d^4, & \text{if } d \leq d_0, \\ E_{TX} * L + E_{fs} * L * d^2, & \text{if } d > d_0, \end{cases} \tag{3}$$

where: E_{TX} – energy consumption per bit of data sent, L – the amount of data transferred, E_{mp}/E_{fs} – depends on the transmitter amplifier model, E_{fs} is used for the free space model, while E_{mp} for the multipath model, d – Euclidean distance between sender and receiver, d_0 – distance threshold calculated based on formula (5). The assumed physical quantities for energy are joules and meters for distance.

On the other hand, the node receiving data is charged with the energy cost of receiving data marked as E_R and is calculated according to the formula (see [35] for details):

$$E_R = (E_{RX} + E_{DA}) * L, \tag{4}$$

where: E_{RX} – power consumption per bit for receiving data, E_{DA} – energy consumption in the data fusion process.

$$d_0 = \sqrt{\frac{E_{fs}}{E_{mp}}}. \tag{5}$$

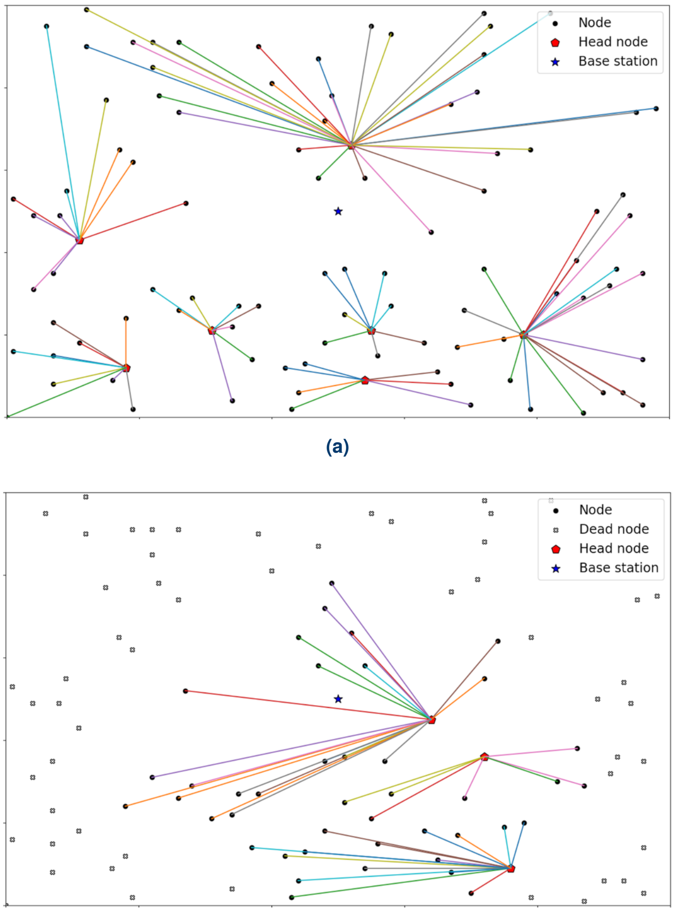
The values of the individual parameters used in the simulation are as follows:

$$E_o = 1\text{ J},$$
$$E_{\text{elec}} = E_{TX} = E_{RX} = E_{DA} = 50\text{ nJ/bit},$$
$$E_{fs} = 10\text{ pJ/bit/m},$$
$$E_{mp} = 0.0013\text{ pJ/bit/m}.$$

Here, E_o is the initial energy of the nodes, while E_{elec} is the energy lost during cluster head selection.

5.3. Experiments

In order to verify the adopted assumptions, and to compare the new MF-LEACH algorithm to the basic one, multiple experiments were conducted. Sample results for a simulation based on the initial number of 100 nodes are presented in Table 1. These results refer to a situation where the maximum number of available functionalities is equal to 4. Assuming the classic approach based on the basic LEACH algorithm, all member nodes implement all available functionalities. Therefore, in each round of the algorithm, each member node sends data packets alternately for all 4 functionalities to one head selected for a given cluster. Such a situation is presented in Fig. 4 for the 1st and 100th rounds, respectively. Indeed, in round 100 it is easy to see nodes that stopped functioning due to the complete energy discharging.



(a)

(b)

Fig. 4. Example operation of the LEACH algorithm (a) round 1, number of dead nodes is 0; (b) round 100, number of dead nodes is 56

In contrast to the classic algorithm, the presented MF-LEACH takes into account the correlation between the number of available functionalities and the number of heads. Figure 5 presents selected visualizations representing an example of the state of operation of the algorithm for functionalities 2 and 3, respectively, in rounds 1 and 90. As can be seen in this figure, in each of the rounds independent heads for individual functions were selected. For the clarity of the perception, they are presented in separate figures. In each round alternately, the member nodes implement all 4 functionalities available in the simulation.

For a better comparison of the operation of the MF-LEACH algorithm, the results for the MF-LEACH-R (random) version are also presented. The modification introduced in this case assumes a random allocation of functionalities performed by individual member nodes in each round. This simulation functionalizes a situation in which one node performs only one functionality at a given moment and another, e.g. all available ones. Still, in each of the rounds, heads for individual functionalities are extracted. In Fig. 6, it can be seen that in the network structure adopted for the simulation, there are member nodes (in a given round) that implement a different number of available functionalities.

The performed simulations confirm that the MF-LEACH algorithm, which is heterogeneous in terms of functions performed by individual nodes, achieves similar results to the basic version of the functionally homogeneous LEACH algorithm. The results presented in Table 1, for the example simulation, show that in individual rounds the results obtained in terms of total energy consumption by nodes are very similar for both algorithms. Moreover, the round number in which the first node discharged is identical.

Table 1
Results of simulation for 100 nodes

	LEACH	MF-LEACH	MF-LEACH-R
TRE – round 50	49.61J	48.78J	70.11J
TRE – round 75	24.42J	23,10J	55.12J
TRE – round 100	1.49J	1.00J	41.42J
FNI	85	85	84
INI – round 100	56	63	19

TRE – total remaining energy, FNI – round number in which the first node was inactive, INI – the number of inactive nodes

It should be noted that in the case of introducing randomness of allocation for the individual member nodes of the individual functionalities, it significantly contributes to the reduction of energy consumption and a much smaller number of nodes that become inactive. These results are also confirmed by Fig. 7a and 7b. In the case of average energy consumption, during the simulation, the results for the basic algorithm, and the proposed

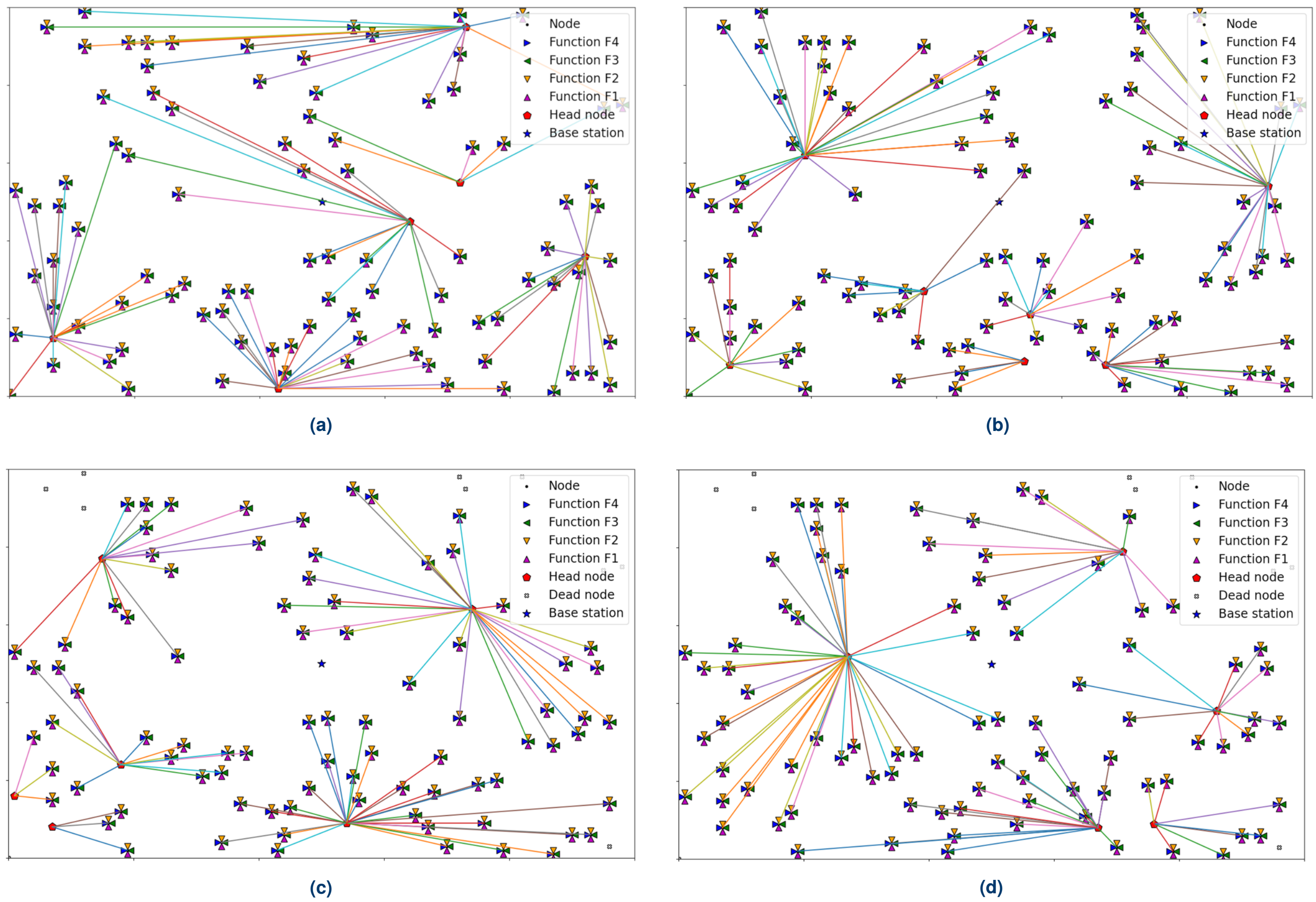


Fig. 5. Example operation of the MF-LEACH algorithm (a) round 1 for functionality 2, number of dead nodes is 0; (b) round 1 for functionality 3, number of dead nodes is 0; (c) round 90 for functionality 2, number of dead nodes is 10; (d) round 90 for functionality 3, number of dead nodes is 10

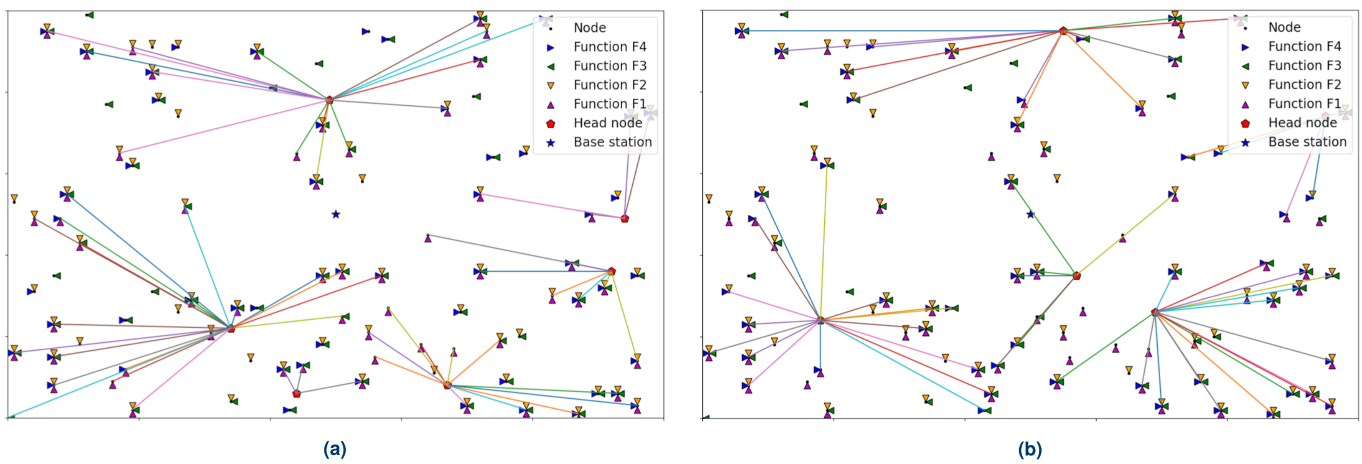


Fig. 6. Example operation of the MF-LEACH-R algorithm (a) for functionality 1 – round 1; (b) for functionality 4 – round 1

modification are similar. On the other hand, as it has already been noted, the random allocation of functionality significantly slows nodes discharging. Figure 7b also confirms the slower process of full inactivity of the nodes in this case, caused by their complete discharge.

The results obtained in this case mean that for an infrastructure, in which it is not required that all nodes perform the same functionalities, random or controlled allocation of functions for member nodes should be considered. However, this research direction is out of the scope of this contribution.

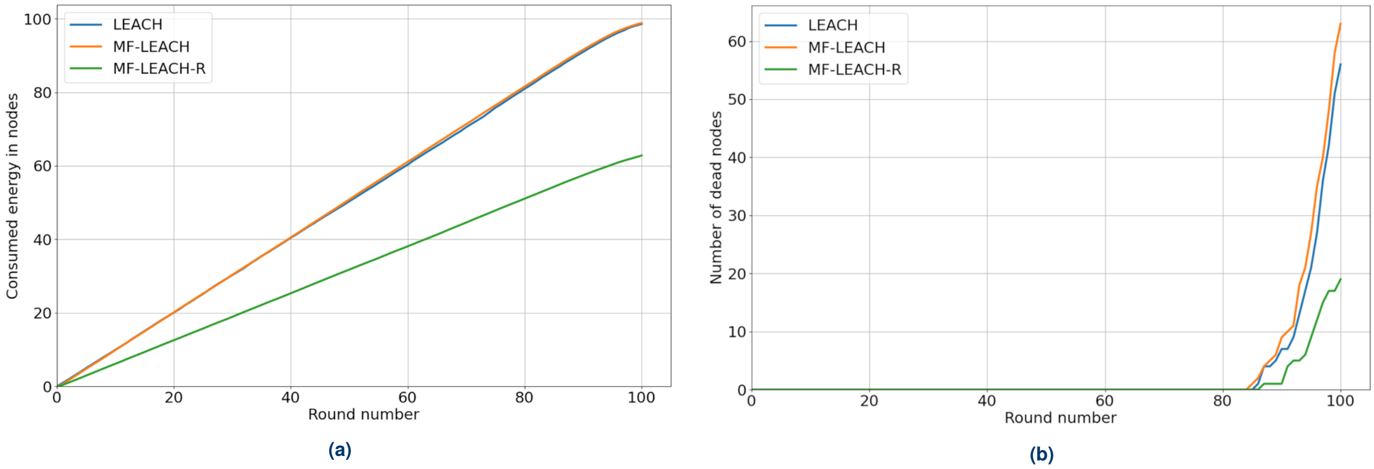


Fig. 7. (a) Consumed energy in nodes; (b) The number of nodes discharged

In order to show a broader perspective on the multifunctionality of the developed solution, further investigations were conducted, considering different packet lengths, depending on the applied function. As part of the simulation, the following packet lengths were experimented with for function $f_1 = 200$ B, for $f_2 = 600$ B, for $f_3 = 1$ KB, and for $f_4 = 1.5$ KB. The goal of the experiments was to verify the performance of the algorithms in a situation where the amount of data transmitted varies, in the case of execution of individual functions. Table 2 presents the results of the completed simulations.

Table 2
Results of simulation for 100 nodes – various packet lengths

	LEACH	MF-LEACH	MF-LEACH-R
TRE – round 50	18.17J	17.49J	45.43J
TRE – round 75	0J	0J	24.29J
TRE – round 100	0J	0J	13.34J
FNI	54	52	51
INI – round 100	100	100	48

In order to compare the results in Table 1 and Table 2, it should be noted that the results in Table 1 refer to the situation where the same packet length was assumed for all functions; and the length of these packets was 64 bytes. Therefore, the energy consumption in subsequent rounds was lower under these conditions. It is important to note, however, that the introduction of variance to the packet length distinctions for the individual functions did not change the core properties of their operation. In fact, the different variants of the algorithms behave similarly. It can be seen that the MF-LEACH algorithm, which is heterogeneous in terms of the functions performed by the individual nodes, still achieves results similar to the basic version of the functionally homogeneous LEACH algorithm. For both algorithms, in this particular simulation, all nodes were already inactive at round 65. In contrast, for MF-LEACH-R, there were still 52 active nodes in the 100th round of the simulation. Thus, when randomization of function allocation is introduced, there is a much slower node discharging in the proposed approach.

Further work included verification of the behavior of the algorithms in the case of variable data transfer intensity for various functions but with the same packet size as for the simulations presented in Table 2. Table 3 summarizes the results of simu-

Table 3
Results of simulation for 100 nodes – variable data transfer intensity

	LEACH	MF-LEACH	MF-LEACH-R
Increasing variant			
TRE – round 50	0.67J	0.26J	33.02J
TRE – round 75	0J	0J	18.41J
TRE – round 100	0J	0J	10.72J
FNI	42	41	40
INI – round 100	100	100	73
Decreasing variant			
TRE – round 50	38.73J	38.29J	58.93J
TRE – round 75	8.55J	7.97J	38.96J
TRE – round 100	0J	0J	25.67J
FNI	71	71	66
INI – round 100	100	100	34
Random variant			
TRE – round 50	17.83J	16.28J	45.95J
TRE – round 75	0J	0J	25.64J
TRE – round 100	0J	0J	15.77J
FNI	53	52	51
INI – round 100	100	100	50

lations conducted in this case. Here, first, it was assumed that for each successive function from the first to the fourth, proportionally more data is transmitted, i.e. if for f_1 one packet was transmitted, then for f_2 two packets, for f_3 three packets and for f_4 four packets (the increasing variant). In the next simulation, this relationship was reversed (the decreasing variant). The third simulation, on the other hand, took into account the randomness of the allocation of the number of packets for each function from 1–4 (the random variant).

The obtained results confirm that comparable operation characteristics of the classic LEACH and the MF-LEACH algorithms remain intact in the case of high-intensity variability. Obviously, the dependence of the load on nodes related to the size of packets and their number causes faster nodes discharging in the event of more data. Also here the version of the algorithm that assumes the randomness of the functionality allocation and the randomness of generating the number of packages for individual functions sustains the network functioning for the longest time.

6. CONCLUSIONS

The contribution of this work is to propose a modification of the LEACH algorithm that allows it to deal with multitasking and multifunctional operation of individual nodes. The developed improvement of the algorithm in this area was based on the concept of selecting a set of separate cluster heads for individual functionalities. Experiments conducted confirm that the energy efficiency of both the basic and the new algorithm is almost the same. Thus, the new solution can be used in systems that take into account the temporary, or permanent, specialization of nodes (such as these described in [13]). The proposed approach may be of immense importance in edge-cloud continuum ecosystems, in which one will strive to optimize processing in a distributed environment. It should be remembered that a distributed IoT environments are often characterized by limited computing and power resources. From this perspective, random or, in the future, controlled management of the dispersion of performed measurement functions may be particularly important. The obtained results confirmed that the randomness of the assignment of performed functions affects the energy life of not only individual nodes but also the entire dispersed infrastructure.

Future work will focus on examining the influence of the process of putting nodes to sleep and waking up on the effectiveness of their operation. Interesting aspects may also be the implementation of load-balancing mechanisms in the proposed variant of the LEACH algorithm. The current research was intended to take into account assumptions that are used by authors of publications on other variants of the LEACH algorithm. However, in future work, we also intend to extend the model to include varying packet lengths for different functions. Furthermore, an important aspect of further research may be to consider different classes of services and QoS. Finally, explorations involving the use of GWEN-type modular nodes will be undertaken.

ACKNOWLEDGEMENTS

This research paper was developed under the project financed by the Minister of Education and Science of the Republic of Poland within the "Regional Initiative of Excellence" program for the years 2019–2023. Project number 027/RID/2018/19, amount granted 11 999 900 PLN. The work of Maria Ganzha and Marcin Paprzycki was funded in part by the European Commission, under the Horizon Europe project aerOS, grant number 101069732.

REFERENCES

[1] A. Hazra, P. Rana, M. Adhikari, and T. Amgoth, "Fog computing for next-generation Internet of Things: Fundamental, state-of-the-art and research challenges," *Comput. Sci. Rev.*, vol. 48, p. 100549, 2023, doi: 10.1016/j.cosrev.2023.100549.

[2] A. Dimou, C. Iliopoulos, E. Polytidou, S.K. Dhurandher, G. Papadimitriou, and P. Nicopolitidis, "A Comprehensive Review on Edge Computing: Focusing on Mobile Users," in *Advances in Computing, Informatics, Networking and Cybersecurity*, vol. 289, P. Nicopolitidis, S. Misra, L.T. Yang, B. Zeigler, and Z. Ning, Eds. Cham: Springer International Publishing, 2022, pp. 121–152. doi: 10.1007/978-3-030-87049-2_30.

[3] V.K. Prasad, M.D. Bhavsar, and S. Tanwar, "Influence of Montoring: Fog and Edge Computing," *Scalable Comput.-Pract. Exp.*, vol. 20, no. 2, pp. 365–376, 2019, doi: 10.12694/scpe.v20i2.1533.

[4] R. Basir, N.A. Chughtai, M. Ali, S. Qaisar, and A. Hashimi, "Mode selection, caching and physical layer security for fog networks," *Bull. Pol. Acad. Sci. Tech. Sci.*, vol. 70, no. 5, p. e142652, 2022, doi: 10.24425/bpasts.2022.142652.

[5] S. Chen and L. Tang, "Flexible English Learning Platform using Collaborative Cloud-Fog-Edge Networking," *Scalable Comput.-Pract. Exp*, vol. 24, no. 3, pp. 339–354, Sep. 2023, doi: 10.12694/scpe.v24i3.2224.

[6] D. Rosendo, A. Costan, P. Valduriez, and G. Antoniu, "Distributed intelligence on the Edge-to-Cloud Continuum: A systematic literature review," *J. Parallel Distrib. Comput.,* vol. 166, pp. 71–94. 2022, doi: 10.1016/j.jpdc.2022.04.004.

[7] A. Paszkiewicz *et al.*, "Network Load Balancing for Edge-Cloud Continuum Ecosystems" in *Proc. Innovations in Electrical and Electronic Engineering ICEEE*, 2022, doi: 10.1007/978-981-19-1677-9_56.

[8] Y. Bar-Yam. *Dynamics of complex systems.* CRC Press, 2019. doi: 10.1201/9780429034961.

[9] G. Cimini, T. Squartini, F. Saracco, D. Garlaschelli, A. Gabrielli, and G. Caldarelli, "The statistical physics of real-world networks", *Nat. Rev. Phy.s*, vol. 1, no. 1, pp. 58–71, 2019, doi: 10.1038/s42254-018-0002-6.

[10] D. Kandris, C. Nakas, D. Vomvas, and G. Koulouras, "Applications of Wireless Sensor Networks: An Up-to-Date Survey,“ *Appl. Syst. Innov.*, vol. 3, no. 1, p. 14, 2020, doi: 10.3390/asi3010014.

[11] K.U. Jaseena and B.C. Kovoor, "Deterministic weather forecasting models based on intelligent predictors: A survey," *J. King Saud Univ.-Comput. Inf. Sci.*, vol. 34, no. 6, pp. 3393–3412, 2022, doi: 10.1016/j.jksuci.2020.09.009.

[12] J. Miranda, P. Ponce, A. Molina, and P. Wright, "Sensing, smart and sustainable technologies for Agri-Food 4.0," *Comput. Ind.*, vol. 108, pp. 21–36, 2019, doi: 10.1016/j.compind.2019.02.002.

[13] G. Elhayatmy, N. Dey, and A.S. Ashour, "Internet of Things Based Wireless Body Area Network in Healthcare," in *Internet of Things and Big Data Analytics Toward Next-Generation Intelligence*, vol. 30, Springer, Cham, 2018, pp. 3–20, doi: 10.1007/978-3-319-60435-0_1.

[14] W.R. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-efficient communication protocol for wireless microsensor networks," in *Proc. 33rd Annual Hawaii International Conference on System Sciences*, 2000, vol. 2, pp. 10, doi: 10.1109/HICSS.2000.926982.

[15] P. Szmeja *et al.*, "ASSIST-IoT: A Modular Implementation of a Reference Architecture for the Next Generation Internet of Things," *Electronics*, vol. 12, p. 854, 2023, doi: 10.3390/electronics12040854.

[16] H. Balakrishnan, A.P. Chandrakasan, and W.B. Heinzelman, "An application-specific protocol architecture for wireless microsensor networks," *IEEE Trans. Wirel. Commun.*, vol. 1, no 4, pp. 660–670, 2002, doi: 10.26636/jtit.2021.147420.

[17] M. Tong and M. Tang, "LEACH-B: An Improved LEACH Protocol for Wireless Sensor Network," *2010 6th International Conference on Wireless Communications Networking and Mobile Computing (WiCOM)*, China, 2010, pp. 1–4, doi: 10.1109/WICOM.2010.5601113.

[18] F. Xiangning and S. Yulin, "Improvement on LEACH Protocol of Wireless Sensor Network," in *Proc. International Conference on Sensor Technologies and Applications (SENSORCOMM)*, 2007, pp. 260–264, doi: 10.1109/SENSORCOMM.2007.4394931.

[19] V. Loscri, G. Morabito, and S. Marano, "A two-levels hierarchy for low-energy adaptive clustering hierarchy (TL-LEACH)," in *Proc. IEEE 62 nd Vehicular Technology Conference*, 2005, pp. 1809–1813, doi: 10.1109/VETECF.2005.1558418.

[20] E.S. Fard, and M.H. Nadimi, "Routing Protocol of Wireless Sensor Network (ED-LEACH)," *Int. J. Sens. Sens. Netw.*, vol. 2, no. 3, pp. 26–30. doi: 10.11648/j.ijssn.20140203.11.

[21] L.Q. Guo, Y. Xie, C.H. Yang, and Z.W. Jing, "Improvement on LEACH by combining Adaptive Cluster Head Election and Two-hop transmission," in *Proc. International Conference on Machine Learning and Cybernetics*, 2010, pp. 1678–1683, doi: 10.1109/ICMLC.2010.5580988.

[22] J. Chen and H. Shen, "MELEACH-L: More Energy-Efficient LEACH for Large-Scale WSNs," in *Proc. 4th International Conference on Wireless Communications, Networking and Mobile Computing*, 2008, pp. 14, doi: 10.1109/WiCom.2008.915.

[23] W. Wang, F. Du, and Q. Xu, "An Improvement of LEACH Routing Protocol Based on Trust for Wireless Sensor Networks," in *Proc. 5th International Conference on Wireless Communications, Networking and Mobile Computing*, 2009, pp. 1–4, doi: 10.1109/WICOM.2009.5303346.

[24] M.S. Ali, T. Dey, and R. Biswas, "ALEACH: Advanced LEACH routing protocol for wireless microsensor networks," *International Conference on Electrical and Computer Engineering*, 2008, pp. 909914, doi: 10.1109/ICECE.2008.4769341.

[25] L.S. Yan, W. Pan, B. Luo, J.T. Liu, and M.F. Xu, "Communication Protocol Based on Optical Low-Energy-Adaptive-Clustering-Hierarchy (O-LEACH) for Hybrid Optical Wireless Sensor Networks," in *Proc. Asia Communications and Photonics Conference and Exhibition, Technical Digest (CD) (Optica Publishing Group, 2009)*, p. ThCC3.

[26] M.A. Abuhelaleh, T.M. Mismar and A.A. Abuzneid, "Armor-LEACH – Energy Efficient, Secure Wireless Networks Communication," in *Proc. 17th International Conference on Computer Communications and Networks*, 2008, pp. 17, doi: 10.1109/ICCCN.2008.ECP.142.

[27] G.S. Kumar, P.M. Vinu, and K.P. Jacob, "Mobility metric based leach-mobile protocol," in *Proc. 16th International Conference on Advanced Computing and Communications*, 2008, pp. 248–253, doi: 10.1109/ADCOM.2008.4760456.

[28] G. Yi, S. Guiling, L. Weixiang, and P. Yong, "Recluster-LEACH: A recluster control algorithm based on density for wireless sensor network," in *Proc. 2nd International Conference on Power Electronics and Intelligent Transportation System (PEITS)*, 2009, pp. 198–202, doi: 10.1109/PEITS.2009.5406834.

[29] M.O. Farooq, A.B. Dogar, and G.A. Shah, "MR-LEACH: Multi-hop Routing with Low Energy Adaptive Clustering Hierarchy," in *Proc. International Conference on Sensor Technologies and Applications*, 2010, pp. 262–268, doi: 10.1109/SENSORCOMM.2010.48.

[30] H. Li, "An energy efficient routing algorithm for heterogeneous wireless sensor networks," in *Proc. International Conference on Computer Application and System Modeling (ICCASM 2010)*, 2010, pp. V3-612–V3-616, doi: 10.1109/ICCASM.2010.5620564.

[31] M.B. Yassein, A. Al-zou'bi, Y. Khamayseh, and W. Mardini, "Improvement on LEACH protocol of wireless sensor network (VLEACH)," *Int. J. Digit. Content Technol. Appl.*, vol. 3, no. 2, pp. 132–136, 2009, doi: 10.4156/jdcta.vol3.issue2.yassein.

[32] A. Miglani, T. Bhatia, G. Sharma, and G. Shrivastava, "An Energy Efficient and Trust Aware Framework for Secure Routing in LEACH for Wireless Sensor Networks," *Scalable Comput.-Pract. Exp.*, vol. 18, no. 3, pp. 207–218, 2017, doi: 10.12694/scpe.v18i3.1301.

[33] S. Varshney and R. Kuma, "Variants of LEACH Routing Protocol in WSN: A Comparative Analysis," *8th International Conference on Cloud Computing, Data Science & Engineering (Confluence)*, 2018, pp. 199–204, doi: 10.1109/CONFLUENCE.2018.8442643.

[34] W. Jin, G. Xiujian, K. Arun, and K. ye-Jin, "An empower hamilton loop based data collection algorithm with mobile agent for WSNs," *Human-centric Comput. Inf. Sci.*, vol. 9, p. 18, 2019, doi: 10.1186/s13673-019-0179-4.

[35] S. Chen, J. Zhou, X. Zheng, and X. Ruan, "Energy-Efficient Data Collection Scheme for Environmental Quality Management in Buildings," *IEEE Access*, vol. 6, pp. 57324–57333, 2018, doi: 10.1109/ACCESS.2018.2873789.