كلية الهندسة بشبرا
FACULTY OF ENGINEERING AT SHOUBRA
ERJ
PRINT ISSN 3009-6049
ONLINE ISSN 3009-6022

ENGINEERING RESEARCH JOURNAL (ERJ)
Volume (52),Issue (4)
October 2023, pp:63-71
https://erjsh.journals.ekb.eg

Study of Clustering Technique Algorithms in IoT Networks

Ahmed Gamal Soliman Soliman Deabes [1,*] , Michael Nasief Mikheal [2], Hala Abd El-kader Mansour [2], Esraa Mosleh [2]

[1] *Modern University for Technology and Information,*
[2] *Department of Electrical Engineering , Faculty of Engineering at Shoubra, Benha University, Shoubra, Cairo, Egypt.*
**Corresponding author*

E-mail address: ahmed_deabes2009@yahoo.com, michaelnasief@yahoo.com, hala.mansour@gmail.com, esraa.soliman@feng.bu.edu.eg

Abstract: The Internet of Things (IoT) refers to a network of interconnected devices that operate on the internet facilitating seamless and efficient data exchange to improve human life. Energy consumption in the IoT network nodes is a major challenge. To overcome this challenge, clustering became a powerful data gathering in IoT applications that saves energy by organizing IoT nodes into clusters. The Cluster Head (CH) oversees all Cluster Member (CM) nodes in each group allowing for the creation of both intra-cluster and inter-cluster connections. There are many algorithms to improve the lifespan of the network, increase the number of active nodes, and extend the remaining energy time in IoT. These algorithms employ techniques such as clustering and optimization to enhance both the energy efficiency and overall performance of the network. In this paper, Low Energy Adaptive Clustering Hierarchy (LEACH), Genetic Algorithm (GA), Artificial Fish Swarm Algorithm (AFSA), Energy-Efficient Routing using Reinforcement Learning (EER-RL), and Modified Low Energy Adaptive Clustering Hierarchy (MODLEACH) algorithms will be studied and MATLAB code will be implemented, tested, and the results will be validated.

Keywords: The Internet of Things - Cluster Head – Energy - Cluster Member.

1. Introduction

The concept of the Internet of Things concerns a system of material objects, such as household appliances, vehicles, and other items, that are equipped with sensors, software, and network connectivity allowing for their integration, communication, and interaction with each other. These devices are connected to the internet, so they communicate with each other and exchange data. The proliferation of mobile devices and internet services has enabled the growth of IoT as these devices provide the necessary connectivity and processing power to support the network of connected devices [1].

The use of IoT has enabled new applications and services, such as smart homes, smart cities, and industrial automation. This enables users to access real-time information about any device connected to the internet increasing productivity and efficiency. For example, IoT-enabled devices in a manufacturing plant can allow operators to monitor equipment remotely and make proactive maintenance decisions [2]. IoT enables devices to share data, send and receive commands, and interact with each other without human intervention [3].

IoT allows people and devices to communicate more efficiently and interact with the environment [4], so humans are free to interact with their surroundings. They will control the items around them rather than just adapting environmental data [5]. However, because of the nature of the network, it confronts several obstacles including dependability, redundancy, and a heterogeneous network with multiple nodes. These difficulties have an influence on the performance of routing protocols at the network layer [6].

Hierarchical routing is one of the effective solutions for extending the life of IoT devices. Nodes in the IoT are organized into clusters, and each cluster is headed by a leader node known as a CH [7,8]. The CH act as intermediaries between the nodes in their cluster and the network's main Base Station (BS) reducing the number of hops required for communication and conserving energy [9,10] . This clustering technique enables network scalability, reduces network traffic, and improves network performance. As a result, battery life is extended, and the overall network lifetime is prolonged [11].

In a network of clusters, there exist several CM nodes that are linked to a singular cluster head node [12]. The CM nodes act as regular nodes in the network and perform various tasks according to the protocol defined for the network. The CH node is responsible for the network management and the coordination of the activities executed by the CM nodes. The CH node generally possesses greater capabilities than those of the CM nodes and has additional responsibilities such as maintaining the cluster topology and routing data between the CM nodes [13]. This routing ensures energy exhaustion savings and a large reduction in communications between IoT nodes. The data transfer and connection into a cluster are the responsibility of the CH node depending on dynamic network effects.

Several algorithms aim to increase the lifespan and energy efficiency of IoT networks. This paper aims to study and test five of these algorithms, namely:

- **LEACH Algorithm:** The Low Energy Adaptive Clustering Hierarchy (LEACH) algorithm is intended to reduce the amount of energy consumed in IoT networks by using a probabilistic approach to cluster formation [14].
- **Genetic Algorithm (GA):** An optimization algorithm that emulates the mechanism of natural selection and can be used for parameter optimization in IoT systems [15].
- **AFSA Algorithm:** The Artificial Fish Swarm Algorithm (AFSA) is an optimization algorithm that belongs to the metaheuristic category and takes inspiration from the collective behavior of fish schools. AFSA can be used to optimize system parameters, such as sensor placement or task scheduling [16].
- **EER-RL Algorithm:** The Energy-Efficient Routing using Reinforcement Learning (EER-RL) algorithm is a routing algorithm that uses reinforcement learning to optimize energy consumption in IoT networks [17].
- **MODLEACH Algorithm:** The Modified Low Energy Adaptive Clustering Hierarchy (MODLEACH) algorithm is a type of algorithm that is derived from the LEACH algorithm, which has been modified to enhance its overall performance or to cater to specific use cases [18].

The rest of this paper is organized as follows: Section II includes a literature review of the selected cluster head technique algorithms for IoT. The details of the algorithms under study are in Section III. Section IV presents the performance analysis, MATLAB implementation, and the simulation results of the algorithms. Section V illustrates the conclusion and future work.

2. Literature review

In [19], the LEACH algorithm uses processes to cluster formation where nodes elect themselves as cluster heads with a probability proportional to their residual energy. This probabilistic algorithm ensures that the distribution of energy load is uniform throughout the network, and nodes with greater remaining energy are more inclined to function as cluster heads. The LEACH algorithm is a simple and effective way to minimize the energy usage of the nodes.

[20] utilize a hybrid that combines the advantages of both centralized and distributed nodes to enhance the energy efficiency and extend the operational lifespan of IoT devices. The IoT-LEACH algorithm also employs a distribution in which the IoT devices within a cluster communicate with each other directly to save energy. The algorithm uses a probabilistic model to choose the cluster head by considering the amount of remaining energy of the IoT devices. This model ensures that the selection of the cluster head occurs in a fair and energy-efficient manner. The IoT-LEACH algorithm is designed for small-scale IoT networks and may not be suitable for large-scale networks. Managing the cluster heads may become excessively burdensome if the number of IoT devices in the network continues to increase reducing the efficiency of the algorithm.

Also, [21] used genetic algorithms that emerged as powerful optimization algorithms that can be applied to various parameters in IoT networks. By encoding potential solutions as strings of genetic information and using selection, crossover, and mutation operators, genetic algorithms can efficiently search the solution space and find optimal solutions. The genetic algorithm is used to optimize the positioning of sensors and minimize the total sensor quantity required for full coverage thus reducing energy consumption and prolonging the network lifetime.

In [22] describes a clustering for IoT nodes localization based on a genetic algorithm. The aim is to improve the accuracy of IoT nodes localization by clustering the sensor nodes and optimizing the CH selection process. The genetic algorithm is used to optimize the process of cluster formation and selection of cluster heads by factoring the remaining energy levels and distance between nodes. The genetic algorithm also helps minimize the computation required for localization, making it suitable for use in large-scale networks. However, there are some limitations. For instance, the accuracy of localization is affected by the number and distribution of anchor nodes. Additionally, the genetic algorithm requires a significant processing power which may be a constraint in energy-constrained IoT nodes.

Furthermore, [23] stated that clustering is important in IoT network as it helps in grouping the nodes based on their attributes and helps in the efficient management of resources. The AFSA algorithm is suitable for tackling optimization issues because of its elasticity, error tolerance, and fast convergence. Similarly, each node adjusts its position to enhance cluster formation. The algorithm optimizes the cluster formation process and prolongs the network's lifespan to ensure that energy usage is distributed evenly across all sensor nodes.

In [24] uses an Artificial Fish Swarm Algorithm to group similar data together in a way that is optimized for efficiency. The algorithm works by simulating the behavior of fish in groups where each fish represents a data point or a centroid in the dataset. The algorithm then performs a series of optimization steps to determine the optimal clustering configuration. The optimization process involves the movement of the fish based on the rules of attraction, repulsion, and alignment which closely resemble the behavior of fish in a swarm.

In addition, [25] optimized the amount of energy consumed by the nodes present in the network while ensuring reliable data transmission. To achieve this objective, EER-RL uses a reinforcement learning algorithm to learn the optimal routing paths based on the current energy level of the nodes and the amount of traffic flowing through the network. Each node in the network is assigned an initial value, which represents the expected reward for choosing a particular action. Each node periodically observes the energy levels of its neighbors and the traffic load in the network. Based on this information, the node is updated. The EER-RL is designed to optimize the energy consumption of the nodes while ensuring reliable data transmission.

In [26], the MODLEACH as a selection criterion for the concept of cluster heads has been amended to take into account the remaining energy levels of the sensor nodes involved. This will guarantee that only sensor nodes possessing adequate energy levels are selected to serve as cluster heads thereby reducing the likelihood of premature node failure due to energy depletion. To further minimize energy consumption, the optimal cluster head selection process combined with data aggregation. The nodes select the cluster heads based on their proximity to data sources and their ability to efficiently aggregate data received from other nodes in their cluster.

Finally, [27] used energy efficiency improvement in the MODLEACH implementation of sleep-wake scheduling for cluster heads. By dynamically alternating between sleep and wake states, cluster heads can conserve energy while still efficiently managing their clusters and transmitting data. These energy efficiency improvements in the Mod-Leach can result in a longer network lifetime, improved reliability, and increased cost-effectiveness.

In this paper, the algorithms (LEACH, GA, AFSA, EER-RL, and MODLEACH) were studied and the MATLAB code will be implemented, tested, and the results will be validated.

3. Algorithms under Study

In this section, a brief explanation of each algorithm will be provided.

3.1 LEACH Algorithm

The LEACH algorithm is routing based on clustering which is frequently employed in IoT networks to minimize energy usage and extend the longevity of the network. The algorithm works by partitioning the network into clusters where each cluster is assigned a specific CH. The CH is accountable for the collection of data from the sensor nodes and delivering it to either the base station or the sink node. The LEACH algorithm operates in rounds in which every cycle comprises two phases: the initial setup phase and the subsequent steady state phase. The selection of CHs during the setup phase is based on the remaining energy levels of sensor nodes and is carried out randomly, and every sensor node determines whether to operate as a CH or a member of a cluster by utilizing a probabilistic function. This function achieves an equitable distribution of CHs throughout the network and maintains energy consumption balance among the sensor nodes. Figure 1 depicts an overview of the LEACH Algorithm.

In the steady state phase, the CHs gather data from their respective cluster members and proceed to transmit it to the BS. Furthermore, the CHs alternate among the sensor nodes to ensure the equal distribution of energy consumption across the network. This process is repeated in each round, with new CHs being selected in each round to prolong network lifetime.

To determine CH use in equation (1) [28] for each round, the algorithm selects a node and generates a random number between 0 and 1. If the generated number falls below the threshold level $T(n)$, then the selected node assumes the role of the CH. It is worth noting that the threshold level is determined by:

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

Here," p" represents the target percentage of CHs that are required." r" denotes the current round, while " G "represents the subset of nodes that have not acted as CHs in the preceding rounds.

The LEACH algorithm adopts a mechanism based on thresholds that reduces energy consumption by deactivating the sensor nodes during periods of inactivity. To achieve an energy consumption balance across the sensor nodes, the threshold value is computed based on the distance between the sensor node and the CH.

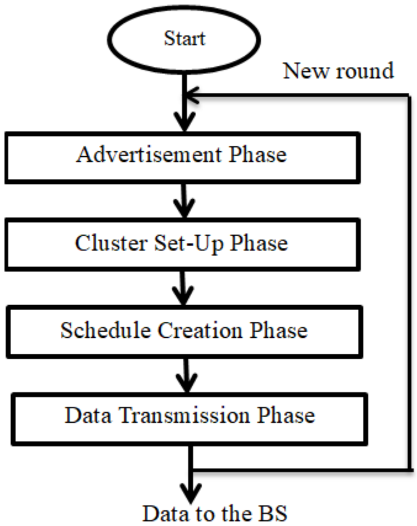

Fig 1. LEACH Algorithm flow chart [14].

3.2 Genetic Algorithm

The Genetic Algorithm is a type of optimization algorithm. The GA algorithm involves creating a population of solutions where each solution represents a possible configuration of the network. These solutions then go through a process of selection, crossover, and mutation which helps to evolve the solutions over much iteration. In the context of IoT networks, the GA algorithm can be used to optimize various aspects of the network such as routing, energy consumption, and node placement.

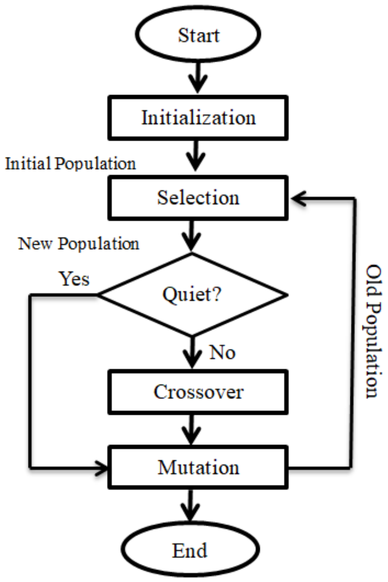

Fig 2. Flow chart of Genetic Algorithm [15].

The GA algorithm is a powerful optimization technique that can be used in IoT networks to solve complex problems such as energy-efficient routing, resource allocation, and task scheduling. Figure 2 shows a general overview of the Genetic Algorithm. Here is a detailed explanation of how the GA algorithm works in IoT networks:

- **Initialization:** The GA algorithm starts by generating an initial population of candidate solutions. In case of energy efficient routing, the population of solutions can be different routing paths between the source and destination nodes.
- **Evaluation:** Each candidate solution is evaluated based on an objective function, such as minimizing energy consumption or maximizing network lifetime. In case of energy-efficient routing, the objective functions to minimize energy consumption while ensuring reliable data transmission.
- **Selection:** A subset of candidate solutions is selected based on their fitness using a selection operator.
- **Crossover:** This process involves selecting pairs of CHs and generating new offspring solutions by utilizing a crossover operator, in the IoT sensor nodes.
- **Mutation:** The offspring solutions that exhibit the highest fitness values are chosen to form the new population of CHs for the following iteration. This process is repeated until a stopping criterion is met such as a maximum number of iterations.
- **Evaluation:** The fitness of the new candidate solutions is evaluated based on the objective function.
- **Replacement:** The least fit candidate solutions are replaced by the new candidate solutions to form the CH.
- **Termination:** The process continues until a termination condition is met.

3.3 AFSA algorithm

The core concept of AFSA is based on functions obtained from fish swarm social behavior. Fish can find regions with more prey in the undersea realm. The individual or group's quest for food by fish results in this circumstance. After that, the artificial fish arrive in a region with a higher intensity and focus on the victim. Figure 3 shows a general overview of the AFSA Algorithm.

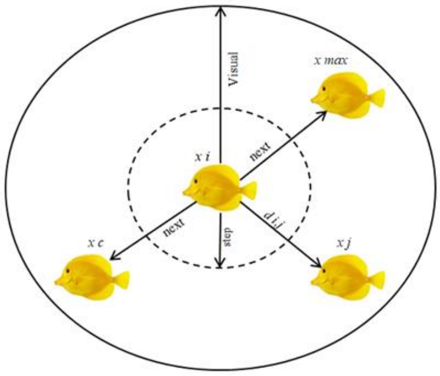

Fig 3. The AFSA algorithm vision concept [16].

This algorithm is adaptable in the tracking area and does not require knowledge of the goal function's value while optimizing in IoT. It also provides excellent practical effects and is not overly sensitive. Figure 4 depicts an overview of AFSA with a detailed description of each phase following.

The first component is searching behavior; the fish are free to wander about in the water looking for food. When the fish find a large amount of food, they rush to it. To compute this behavior, use equation (2) [29]. Where X_c denotes central position of AF.

$$\mathrm{X_j} = \mathrm{X_i} + \mathrm{AF\_visual.\,Rand} \tag{2}$$

Where $\mathrm{X_i}$ denotes the current AF state and $\mathrm{X_j}$ at its observable distance is chosen randomly. The greater AF_visual value, the easier it is to detect the AF extreme global worth. In equation (3) [29], If $\mathrm{Y_j} < \mathrm{Y_i}$ is the problem's last boundary, AF moves forward. Where X_i^{t+1} is the state of AF at time t.

$$X_i^{t+1} = X_i^t + \left(\frac{X_j - X_i^t}{\|X_j - X_i^t\|}\right).AF_{step}.Rand \tag{3}$$

The second component is swarming behavior; fish can obtain food while moving to survive and escape potential threats. To prevent crowding with other companions, they also follow three principles: cohesion, unification, and moving in the identical track like other sets by equation (4) [30].

$$X_i^{t+1} = X_i^t + \left(\frac{X_c - X_i^t}{\|X_c - X_i^t\|}\right).AF\_step.Rand \tag{4}$$

The third component is following behavior. When a fish's movement pattern detects more food, the food is immediately discovered by the other fish. Xi specifies the present situation of AF, and AF locates its surroundings for $\mathrm{dij} < \mathrm{AF\_visual}$ the investigation of AF X_j utilizingY_j. If $Y_j > Y_i$ and $Y_j < AF_{delta}.\frac{Y_i}{n_f}$, AF X_j has a higher nutritional concentration (the cost function has even more enormous value) with minimal congestion.

Equation (5) [16] gives the length of the connecting line between nodes p and q. The median for the typical count of nodes assigned to each cluster is provided by equation (6) [16]. All network nodes should be covered by CH nodes, ensuring complete network coverage. Where $\mathrm{ED(q,p)}$ denotes the length between nodes p and q.

$$ED(q,p) = \sqrt{\sum_{m=1}^{d}(q_m - p_m)^2} \tag{5}$$

$$\mu = \frac{N-K}{K} \tag{6}$$

Where K denotes the number of clusters, N is the total number of IoT nodes and μ is the average number of IoT nodes per cluster.

3.4 EER-RL Algorithm

EER-RL facilitates the enhancement of routing decisions among devices by exchanging local information with nearby nodes leading to the optimized selection of subsequent hops and minimized energy consumption. The transmission of a packet includes information that is specific to the local network and is included in the packet header which is extracted by any nearby device that can overhear it. Utilizing the shared local information that is specific to the local network such as the device's ID, position coordinates, and residual energy level enables the routing table of a device to be updated. This facilitates enhanced routing within the network by allowing the nodes to adapt to changes in the network environment and make more informed routing decisions. EER-RL involves three main stages: network setup and election of cluster heads, formation of clusters, and transmission of data.

During the first stage of the Network Setup and Election of the Cluster Head process, the computation of the initial Q-value for devices occurs, utilizing their local information during the network setup stage. After receiving a message from the base station containing its position coordinates, each device saves the base station's position and computes its initial Q-value by applying equations (7) and (8) [17] and factoring in hop count and the initial energy level. Since all devices have different energy levels, a threshold distance is defined between the BS and CHs to minimize network overhead and assist IoT sensors located far away from the BS in discovering a suitable CH. To prevent energy waste caused by an increase in connection distance, and to ensure that links converge towards the BS instead of diverging, it is necessary to position CHs away from the network's edge.

$$Q = \begin{cases} \frac{1}{N_h}, & if\ E_{min} = E_{max} \\ p \times \left(\frac{E_r - E_{min}}{E_{max} - E_{min}}\right) + (1-p) \times \frac{1}{N_h}, & if\ E_{min} \neq E_{max} \end{cases} \tag{7}$$

$$N_h \cong \frac{D_{link}}{TX_{range}} \tag{8}$$

Where N_h denotes the distances between the nodes, E_r is the receiver energy, E_{min} is the minimum energy, E_{max} is the maximum energy, p denotes number of companions, D_{link} denotes the link distance between the nodes.

During the second stage of cluster formation, after the election of cluster heads, to inform the devices within its transmission range, each Cluster Head (CH) sends an invitation message. The message serves to inform the devices that the CH has been selected to lead. The invitation message sent by each CH includes its ID, location coordinates, and initial Q-value. Devices that are not CHs and overhear these invitation messages use the information to determine which cluster to join, based on factors such as distance. They then send a request to the designated CH, which includes their local information. If a device receives several invitations, particularly when it's positioned at the intersection of various clusters, it can choose to join the cluster that has the closest channel.

The model for energy consumption in the third stage involves both the sender and receiver utilizing energy after a packet transmission. The energy consumption of the sender is higher due to it having to amplify the signal over a distance and send packets through the network. The energy consumption model is used to calculate the energy dissipated during the transmission or reception of packets and to update the residual energy. Equation (9) [17] presents the energy consumption model.

$$\begin{cases} E_{TX}(k,d) = E_{elec} \times K + E_{amp} \times k \times d^m, \\ E_{RX}(k) = E_{elec} \times K, \end{cases} \tag{9}$$

$\mathrm{E_{TX}(k,d)}$ and $\mathrm{E_{RX}(k)}$ denote the energy consumed by the receiver and sender respectively. $\mathrm{d^m}$ is the distance between the nodes, k is the energy coefficient, $\mathrm{E_{elec}}$ is the initial energy, $\mathrm{E_{amp}}$ is the current energy of each node.

3.5 MODLEACH Algorithm

The MODLEACH algorithm is a modified version of the popular LEACH algorithm. In the upcoming round, there is a cut-off level for the formation of CHs. If the current cluster has conserved energy and possesses more energy than the specified level, it will persist as a CH for the succeeding round. The use of this algorithm results in the preservation of energy that would otherwise have been expended in packet routing for new CHs and cluster establishment. Nevertheless, when the energy level of the CH falls below a designated threshold, the algorithm initiates the selection of a replacement. In addition to constraining energy consumption during cluster formation, the algorithm introduces two separate power levels to enhance signals depending on the type of transmission. In a clustered network, there are commonly three types of transmission: Intra-cluster transmission, Inter-cluster transmission, and transmission from the CH to the BS.

Intra-cluster transmission refers to all forms of communication occurring among members of a cluster. Such communication involves member nodes sensing data and conveying it to the CH. The exchange of data between two CHs is an instance of inter cluster transmission, while direct data transmission from a CH to the BS falls under the category of CH to BS transmission.

The energy needed for inter-cluster or CH to BS communication cannot be equal to that needed for intra-cluster communication. Even though the amplification energy is currently uniform for all transmissions, adopting a lower energy level for intra-cluster transmissions instead of CH to BS transmissions can lead to noteworthy energy conservation. Additionally, incorporating multiple power levels can reduce the rate of packet drop.

MODLEACH utilizes a routing algorithm that instructs a node, acting as a CH, to employ high-power amplification. Once the node transforms into a cluster member in the next round, the routing algorithm switches it to low-level power amplification.

4. Algorithms MATLAB Implementation and Simulation Results

This section presents the results of MATLAB 2015 implementation for algorithms mentioned above, tests for alive nodes in each round and the average energy left in the nodes as time progresses, and the results were validated. The hardware specifications include the operating System - Windows 10, Processor - Intel 64, and RAM - 4GB. MATLAB is one of the most widely used software packages in the field of IoT and has the advantage of using realistic methods in radio propagation channel modeling which helps to reduce simulation uncertainty.

4.1 LEACH Algorithm

LEACH Algorithm implemented using MATLAB and the results were validated as mentioned in [14]. In 1400 rounds, simulation was conducted. The network size 100 m ×100 m and a BS located in the middle of the grid. The network consists of 300 IoT nodes each having initial node energy of 2 J. Due to the faster transmission rate of IoT nodes, the data size of packet is 2000 bits. Figure 5 depicts the performance of the LEACH algorithm to describe the count of alive node in each round and the time until the first node dies. It also shows the first dead node after 210 rounds; this refers to the duration from the beginning of the simulation until the first node's death. Figure 6 depicts the energy left in the nodes on average as time progresses. The results show that the LEACH algorithm has the least amount of remaining energy after 600 rounds.

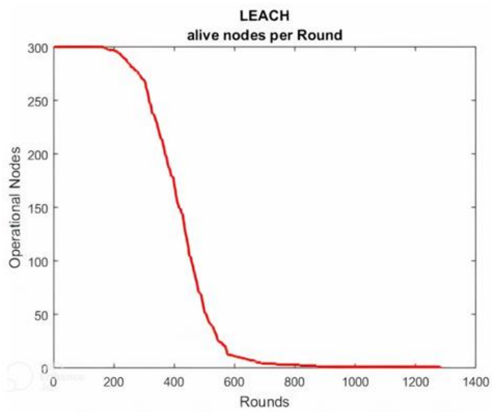

Fig 5. Number of alive nodes per round in LEACH Algorithm.

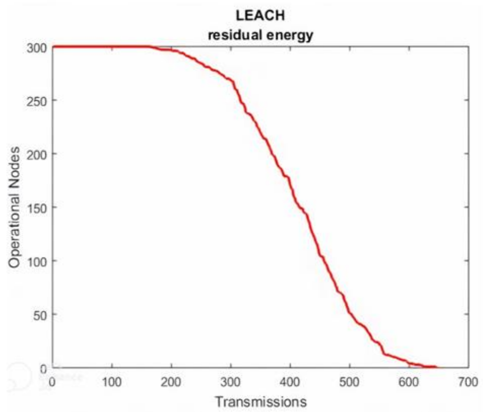

Fig 6. Remaining energy for each round in LEACH Algorithm.

4.2 Genetic Algorithm

Genetic Algorithm was implemented using MATLAB and the results were validated as mentioned in [15]. In 2500 rounds, simulations are conducted. The Network size 400 meters and a BS located in the middle of the grid. The network consists of 400 IoT nodes each having initial node energy of 0.5 J. Due to the faster transmission rate of IoT nodes, the packet size is 4 KB. Figure 7 represents the Genetic Algorithm in which the network operational has a maximum capacity of 2000 rounds. The genetic algorithm illustrated in Figure 8 displays the mean remaining energy. The genetic algorithm has the lowest amount of remaining energy after 1800 rounds.

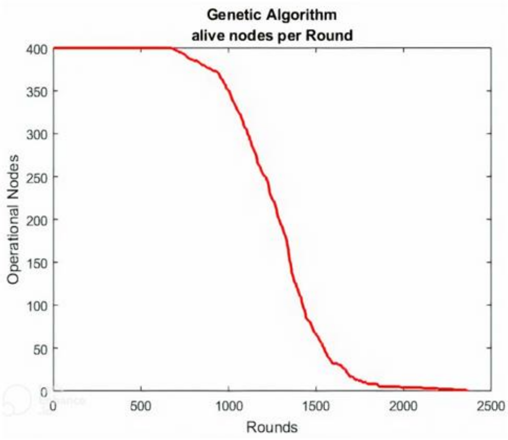

Fig 7. Number of alive nodes per round in Genetic Algorithm.

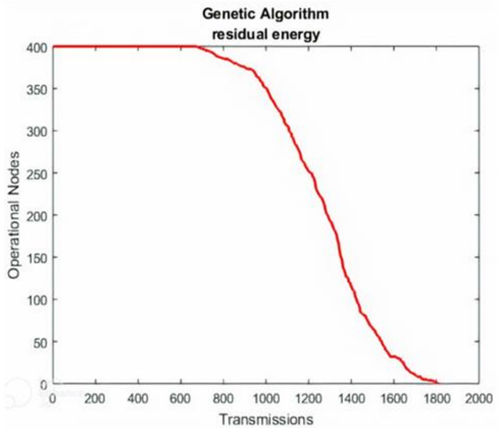

Fig 8. Remaining energy for each round in Genetic Algorithm.

4.3 AFSA Algorithm

AFSA Algorithm was implemented using MATLAB and the results were validated as mentioned in [16]. In 900 rounds, simulations are conducted. The network has a circle with a radius of 500 meters and a BS located in the middle of the grid. The network consists of 300 IoT nodes, each having initial node energy of 0.5 J. Due to the faster transmission rate of IoT nodes, the packet size is 1024 bytes. The performance of the AFSA algorithm is demonstrated in figure 9 about the count of live nodes per round. The AFSA algorithm allows for the completion of up to 400 rounds while maintaining 270 active nodes. Figure 10 depicts the mean remaining energy of the network. The AFSA algorithm has the lowest amount of remaining energy after 710 rounds.

4.4 EER-RL Algorithm

EER-RL Algorithm implemented using MATLAB and the results were validated as mentioned in [17]. In 5000 rounds, simulations are conducted. The network sensing field size 100 m ×100 m and a BS located in the middle of the grid. There are 100 IoT nodes in the network, and they all have an initial energy of 1 joule per node. Due to the faster transmission rate of IoT nodes, the data size is 4000 bits. In terms of the network's lifespan, figure 11 displays the performance assessment of the EER-RL algorithm. Figure 12 exhibits the energy consumption per round.

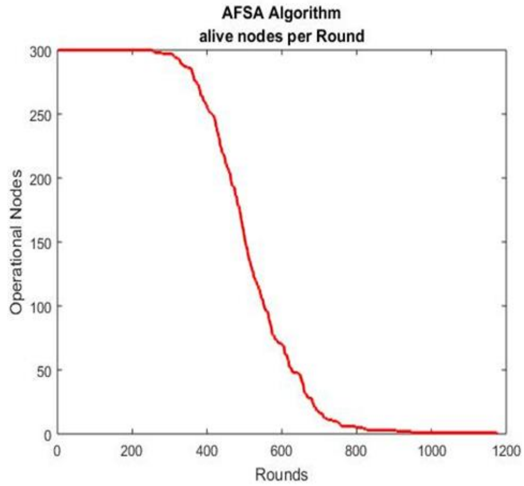

Fig 9. Number of alive nodes per round in AFSA Algorithm.

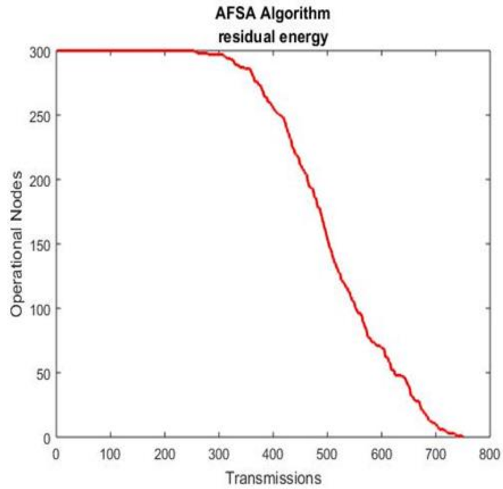

Fig 10. Remaining energy for each round in AFSA Algorithm.

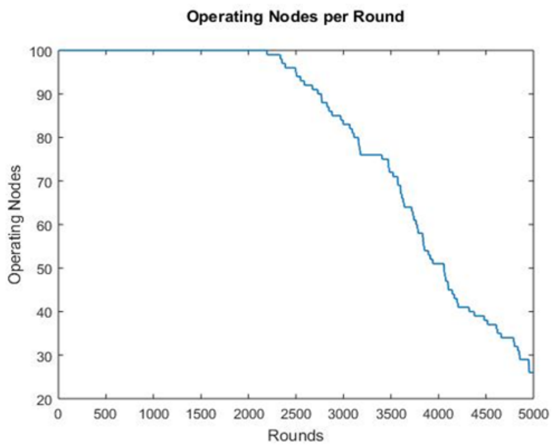

Fig 11. Performance evaluation in EER-RL Algorithm.

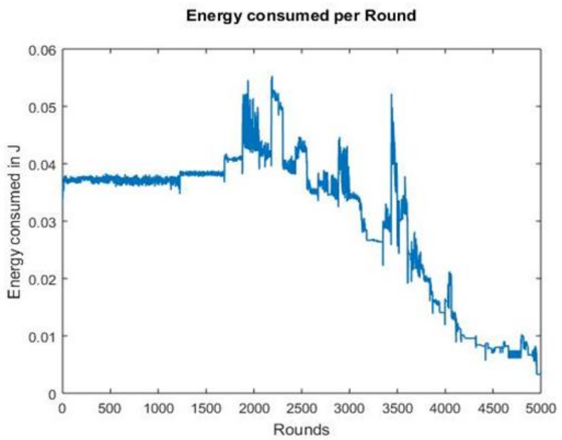

Fig 12. Energy consumption evaluation in EER-RL Algorithm.

4.5 MODLEACH Algorithm

MODLEACH Algorithm was implemented using MATLAB and the results were validated as mentioned in [18]. In 2500 rounds, simulations are conducted. The network sensing field size is 100 m ×100 m and a BS is located in the middle of the grid. The network consists of 100 IoT nodes each having initial node energy of 0.5 J. Due to the faster transmission rate of IoT nodes, the data size is 4000 bits. The count of inactive nodes in the network after 1000 rounds is demonstrated in Figure 13. The count of packets transmitted to the BS is depicted in Figure 14. Figure 15 shows the quantity of data packets transmitted to CH. The graph in Figure 16 displays the number of active nodes within the network. It also shows the first dead node after 100 rounds; this refers to the duration from the beginning of the simulation until the first node's death.

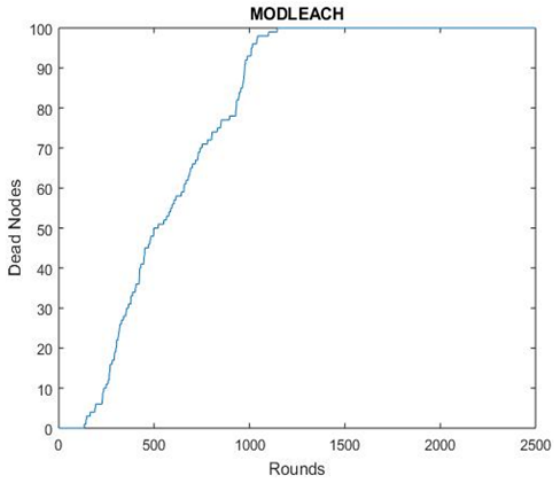

Fig 13. Dead Nodes per round in MODLEACH Algorithm.

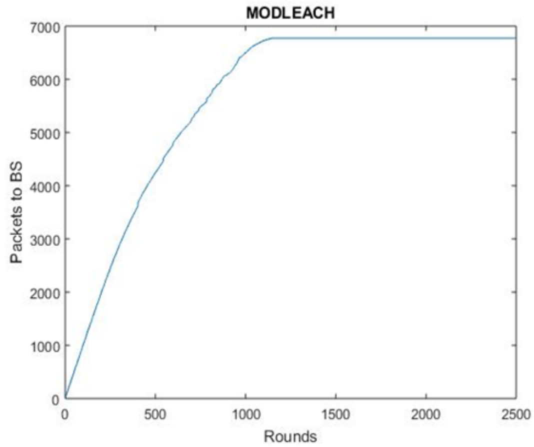

Fig 14. Packets to BS per round in MODLEACH Algorithm.

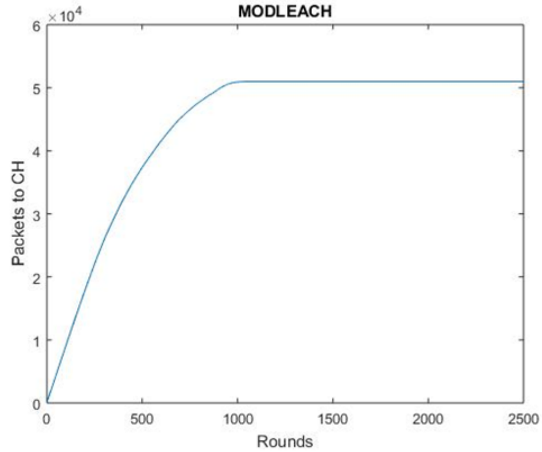

Fig 15. Packets to CH per round in MODLEACH Algorithm.

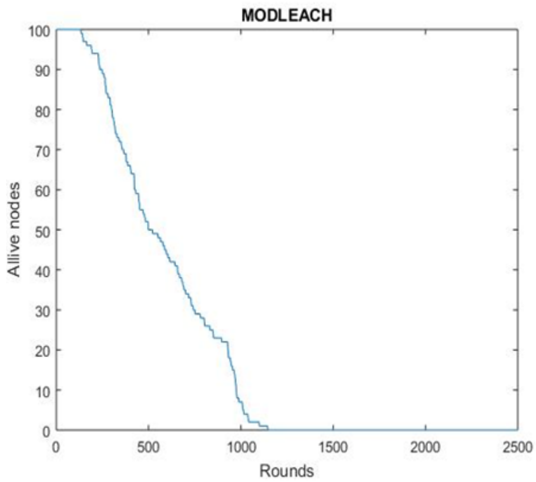

Fig 16. Alive nodes per round in MODLEACH Algorithm.

Conclusions

IoT refers to a network of physical objects connected to the internet enabling them to communicate and share information. This paper concerns the most significant challenge facing the Internet of Things in terms of energy consumption. Clustering is recognized as a solution to reduce energy consumption and increase the network's lifespan. In this paper, LEACH, GA, AFSA, EER-RL, and MODLEACH Algorithms were studied and Matlab code was implemented and tested for alive nodes in each round. The algorithms achieve energy saving, network lifetime, and throughput. The number of alive nodes per round, remaining energy for each round and the results were validated to be used for further research on the performance evaluation comparison for clustering techniques.

REFERENCES

[1] J. Chase, "Introduction The Evolution of the Internet of Things From connected things to living in the data, preparing for challenges and IoT readiness," Texas Instruments, vol. 1, pp. 1–7, 2020.

[2] D. Bandyopadhyay and J. Sen, "Internet of things: Applications and challenges in technology and standardization," Wirel. Pers. Commun., vol. 58, no. 1, pp. 49–69, 2011.

[3] S. Hajiheidari, K. Wakil, M. Badri, and N. J. Navimipour, "Intrusion detection systems in the Internet of things: A comprehensive investigation," Comput. Networks, vol. 160, pp. 165–191, 2019.

[4] A. Sarkeshikian, M. Shafia, A. Zakery, and A. Aliahmadi, "Simulation of stakeholders' consensus on organizational technology acceptance (case study: Internet of Things)," Kybernetes, vol. 50, no. 8, pp. 2367–2396, 2020.

[5] M. Hamzei and N. Jafari Navimipour, "Toward Efficient Service Composition Techniques in the Internet of Things," IEEE Internet Things J., vol. 5, no. 5, pp. 3774–3787, 2018.

[6] M. Sadrishojaei, N. J. Navimipour, M. Reshadi, and M. Hosseinzadeh, "A new clustering-based routing method in the mobile internet of things using a krill herd algorithm," Cluster Comput., vol. 25, no. 1, pp. 351–361, 2022.

[7] D. Valencia, "A Non-threshold-based Cluster-head Rotation," 2018 IEEE Glob. Commun. Conf., pp. 1–6, 2018.

[8] M. Amirinasab, S. Shamshirband, A. T. Chronopoulos, A. Mosavi, and N. Nabipour, "Energy-efficient method for wireless sensor networks low-power radio operation in internet of things," Electron., vol. 9, no. 2, 2020.

[9] N. M. A. Latiff, C. C. Tsimenidis, and B. S. Sharif, "Energy-aware clustering for wireless sensor networks using particle swarm optimization," IEEE Int. Symp. Pers. Indoor Mob. Radio Commun. PIMRC, pp. 5–9, 2007.

[10] R. Aslani and M. Rasti, "A Distributed Power Control Algorithm for Energy Efficiency Maximization in Wireless Cellular Networks," IEEE Wirel. Commun. Lett., vol. 9, no. 11, pp. 1975–1979, 2020.

[11] A. Dhumane, S. Chiwhane, K. Mangore Anirudh, and S. Ambala, Cluster-Based Energy-Efficient Routing in Internet of Things, vol. 311. Springer Nature Singapore, 2023.

[12] V. Rajaram and N. Kumaratharan, "An optimized clustering using hybrid meta-heuristic approach for wireless sensor networks," Int. J. Commun. Syst., vol. 33, no. 18, pp. 1–12, 2020.

[13] T. M. Behera, S. K. Mohapatra, U. C. Samal, M. S. Khan, M. Daneshmand, and A. H. Gandomi, "Residual energy-based cluster-head selection in WSNs for IoT application," IEEE Internet Things J., vol. 6, no. 3, pp. 5132–5139, 2019.

[14] E. F. Ahmed Elsmany, M. A. Omar, T. C. Wan, and A. A. Altahir, "EESRA: Energy efficient scalable routing algorithm for wireless sensor networks," IEEE Access, vol. 7, pp. 96974–96983, 2019.

[15] R. Muthukkumar et al., "A genetic algorithm-based energy-aware multi-hop clustering scheme for heterogeneous wireless sensor networks," PeerJ Comput. Sci., vol. 8, pp. 1–30, 2022.

[16] M. Sadrishojaei, N. Jafari Navimipour, M. Reshadi, M. Hosseinzadeh, and M. Unal, "An energy-aware clustering method in the IoT using a swarm-based algorithm," Wirel. Networks, vol. 28, no. 1, pp. 125–136, 2022.

[17] V. K. Mutombo, S. Lee, J. Lee, and J. Hong, "EER-RL: Energy-Efficient Routing Based on Reinforcement Learning," Mob. Inf. Syst., vol. 2021, 2021.

[18] D. Mahmood, N. Javaid, S. Mahmood, S. Qureshi, A. M. Memon, and T. Zaman, "MODLEACH: A variant of LEACH for WSNs," Proc. - 2013 8th Int. Conf. Broadband, Wirel. Comput. Commun. Appl. BWCCA 2013, pp. 158–163, 2013.

[19] and H. B. H.B Wendi R. Heinzelman, A. Chandrakasan, "Proceedings of the 33rd Hawaii International Conference on System Sciences," Energy-Efficient Commun. Protoc. Wirel. Microsens. Networks, vol. 00, no. c, pp. 1–10, 2000.

[20] S. Bhatia, A. Kumar, A. Kumar, and A. S. Alshuhail, "Performance analysis of energy efficient improved LEACH protocol in IoT networks," IET Commun., no. May, pp. 1–17, 2022.

[21] W. Abdallah, S. Mnasri, and T. Val, "Genetic-Voronoi algorithm for coverage of IoT data collection networks To cite this version : Genetic-Voronoi algorithm for coverage of IoT data collection networks," 2020.

[22] Y. Pu et al., "Method : Fractional Steepest Descent Approach," IEEE Trans. Neural Networks Learn. Syst., vol. 26, no. 4, pp. 653–662, 2015.

[23] X. Li, B. Keegan, and F. Mtenzi, "Energy efficient hybrid routing protocol based on the artificial fish swarm algorithm and ant colony optimisation for WSNs," Sensors (Switzerland), vol. 18, no. 10, 2018.

[24] [24] M. Ouyang, J. Xi, W. Bai, and K. Li, "Band-Area Application Container and Artificial Fish Swarm Algorithm for Multi-Objective Optimization in Internet-of-Things Cloud," IEEE Access, vol. 10, pp. 16408–16423, 2022.

[25] S. E. Bouzid, Y. Serrestou, K. Raoof, and M. N. Omri, "Efficient Routing Protocol for Wireless Sensor Network based on Reinforcement Learning," 2020 Int. Conf. Adv. Technol. Signal Image Process. ATSIP 2020, pp. 6–10, 2020.

[26] Z. Pang, W. Zhang, S. Wang, J. Luo, and H. Chen, "Energy-Efficient Improvements in Mod-LEACH Protocol on Optimal Cluster Heads Selection," Proc. - IEEE 2018 Int. Congr. Cybermatics 2018 IEEE Conf. Internet Things, Green Comput. Commun. Cyber, Phys. Soc. Comput. Smart Data, Blockchain, Comput. Inf. Technol. iThings/Gree, pp. 1765–1769, 2018.

[27] S. Ananda Kumar, P. Ilango, and G. H. Dinesh, "A modified LEACH protocol for increasing lifetime of the wireless sensor network," Cybern. Inf. Technol., vol. 16, no. 3, pp. 154–164, 2016.

[28] B. Parmar, J. Munjani, J. Meisuria, and A. Singh, "A Survey of routing protocol LEACH for WSN," Int. J. Sci. Res. Publ., vol. 4, no. 1, pp. 2–5, 2014.

[29] T. K. Saini and S. C. Sharma, "Self-managed access scheme for demand request in TDM/TDMA star topology network," Def. Sci. J., vol. 69, no. 1, pp. 80–86, 2019.

[30] S. Branch et al., "a Review of Artificial Fish Swarm Optimization," vol. 5, no. 1, pp. 107–148, 2012.