Proposing a Modern Clustering Method for Increasing the Lifetime of Wireless Ad-Hoc Networks

Goldoust Hamed and Khani Ziba

Department of Information Technology and Computer Engineering, Azarbaijan Shahid Madani University, Tabriz, Iran

Abstract
This paper proposes a clustering method using Cuckoo algorithm to increase the lifetime of wireless ad-hoc networks. The proposed algorithm selects cluster heads based on path length energy. We also propose another algorithm for the proper distribution of selected cluster heads. The second algorithm randomly selects the best cluster head and prevents the frequent selection of one node as the best cluster head which would result in the waste of energy and early death of cluster heads. We compared the first method with LEACH, FSCA, GSAGA and CORDF in terms of the first dead node, energy consumption, the number of live and dead nodes, computation volume, and clustering indices. The results confirmed the significant advantage of the proposed algorithm over the aforementioned methods. Finally, we compared the second algorithm with the first one in terms of distribution, generation rate, data delivery delay, and energy harvesting. The results confirmed that the second algorithm was more reliable and cost-efficient than the first algorithm in the aforementioned criteria.

Keywords:
Wireless ad-hoc network, clustering algorithm, pilot cluster head, optimal cluster head

1. Introduction

An efficient clustering protocol may significantly improve the network performance. In cluster-based networks, cluster heads are responsible for energy-hungry tasks. The recent studies have focused on the selection of nodes with energy harvesting capability. Researchers have proposed many contracts for energy consumption optimization, which can be divided into hierarchical and non-hierarchical groups [1-4]. These contracts are based on clustering algorithms and the relationship between ad-hoc nodes. Non-hierarchical contracts represent energy problem by clustering methods. Some examples of non-hierarchical contracts are Data Centric [5], SMECN [6], Flooding [7], Gossiping [8] and Diffusion Directed [9].

Non-hierarchical methods consume a lot of energy due to direct relationship between node and base station, because the nodes have to send the data to far distances. Hierarchical and cluster-based contracts collect the data with minimum energy consumption. Some examples of hierarchical contracts are LEACH, PEGASIS [10], HIT [11], LPT [12] and DBASIC_STATION [13]. LEACH and PEGASIS are the standard contracts in this group.

Manuscript received September 5, 2026
Manuscript revised September 20, 2026
https://doi.org/10.22937/IJCSNS.2026.26.9.3

So far, many clustering algorithms have been proposed. These algorithms often differ from each other in terms of cluster head selection method. In LEACH algorithm (Low Energy Adaptive Clustering Hierarchy) [14], cluster heads are randomly selected and ad-hoc nodes have equal chance to become cluster head. FSCA [15] is another clustering algorithm which develops cluster heads based on the remaining energy and intra-cluster communication cost.

In CORDF [16], nodes are divided into clusters with unequal sizes. Smaller clusters are closer to the station and the cluster heads of the smaller clusters can preserve more energy. PEGASIS [17] is another clustering algorithm which forms a chain of nodes until the central station, so that each node combines the received data with its own data and sends the combined data to its neighbor and the data is finally transmitted to the station.

Cuckoo algorithm was first designed for continuous optimization problems. Later, it was discretized for binary problems [18]. This algorithm has been used for solving problems such as data collection and node location in wireless ad-hoc networks [19-20]. Wireless ad-hoc networks perform clustering using certain criteria such as remaining energy of ad-hoc nodes, the distance between ad-hoc nodes, and path length energy. The algorithms have been compared in Table 1. So far, many clustering algorithms have been proposed, but none has proved efficient in all criteria. This paper aims to propose an algorithm with a good efficiency in the majority of these criteria.

Table 1: Comparison of clustering algorithms

	LEACH	LEAH-C	FSCA	GSAGA	PEGASIS	CORDF
Cluster process	Distributed	Centralized	Distributed	Distributed	Distributed	Distributed
Energy waste	Average	How	Low	Average	Low	Low
Cluster head rotation	Yes	Yes	No	No	Yes	Yes
Network lifetime	Average	Average	High	Average	High	High

Number of cluster heads	Variable	Variable	Variable	Variable	Variable	Variable
Clustering topology	Single-step	Single-step	Multistep	Multistep	Multistep	Multistep
Cluster head probability criterion	Probability function	Probability function	Random	Random	Probability function	Probability function
Multilevel	No	No	Yes	Yes	Yes	Yes

2. Proposed Method

Cuckoo algorithm [22] is a heuristic algorithm inspired by a bird named cuckoo which places its eggs in the nest of other birds whereby entrusting them with the maintenance of its eggs. Each cuckoo randomly places its eggs in the nest of host birds which are located in its egg laying radius (ELR). When all cuckoos laid their eggs, some eggs which are less like the host bird's eggs are identified and thrown away from the nest. The rest of chicks are fed by the host bird. After becoming adult birds, cuckoos live in their habitats for a while. When egg laying season comes, they form some clusters using k-means method and then migrates to an optimal point (target point). The concept of immigration in Cuckoo algorithm is that each cuckoo tries to migrate to the best point. In other words, it tries to shorten its distance from the target point [23].

The proposed algorithm mentioned in [21] uses Heinzelmann radio model which is a classic radio model for wireless ad-hoc networks [4]. The radio model includes transmitter and receiver nodes separated by 'd' intervals. E_{tx} and E_{rx} are the energy consumed in electronic circuits of the sender and receiver. E_{amp} is the energy consumed in transmitter amplifier which depends on the transmission model, i.e. multipath free space with acceptable bit error rate. We consider ε_{fs} and E_{amp} as the energy consumed in the amplifier circuit for free space model and multipath fading model respectively. The electronic circuits of the sender and receiver are dependent on digital coding, modulation, filtering and data dissemination. Therefore, if a node is a cluster head, E_{agg} energy consumption, which refers to the energy consumed for data collection, will also exist.

Package sending method has been shown in Table 2 (row 22). L_p is the length of package in bits. The value of n is 2 for free space and 4 for multipath fading. Let's assume that a node sends a package. Each bit of the package consumes E_{tx} value of transmitter electronic circuits and E_{amp} value of amplifier energy. In total, E_t energy is consumed for sending a package with the length of L_p.

Package receiving method has been shown in row 23. Here, likewise, L_p is the length of package in bits. Let's assume that ad-hoc node receives a package. Each bit of the package consumes E_{rx} value of the energy of electronic circuits. In total, E_r energy is consumed for receiving a package with the length of L_p.

Table 2: Symbols

No.	Symbol	Description
1	d	Distance
2	E_{tx}	Energy consumed in transmitter electronic circuits
3	E_{rx}	Energy consumed in receiver electronic circuits
4	E_{amp}	Energy consumed in transmitter amplifier
5	ε_{fs}	Free space model of the energy consumed in transmitter circuit
6	E_{amp}	Multipath fading model of the energy consumed in amplifier circuit
7	E_{agg}	Energy consumed for data collection
8	L_p	Pack length in bit
9	n=2	Free space
10	n=4	Multipath fading
11	E_t	Total energy
12	E_r	Total energy
13	BASIC STATION	*Base station*
14	C HEAD	*Cluster head*
15	p	*Percentage of needed cluster heads*
16	G	*Total nodes for becoming cluster head*
17	r	*Number of repetitions*
18	E_{CH}	*Remaining energy of cluster head*
19	E_{Node}	*Remaining energy of origin node (ordinary node)*
20	$D_{CH, Node}$	*Distance between cluster head and ordinary node*
21	$FU=(E_{CH}+E_{Node})/D_{CH, Node}$	*Utility function*
22	$E_t = (L_p + E_{tx}) + (L_p * E_{amp} * d_n)$	Package sending method

23	$E_r = (L_p * E_{rx})$	Package receiving method
24	$T(n) = \begin{cases} \dfrac{P}{1 - P\left(r \bmod \frac{1}{P}\right)} & if\ n \in G \\ 0 & otherwise \end{cases}$	*Random number*

2.1 Hypotheses of the Proposed Algorithm

1. In both proposed algorithms, the network has one sink node which is in a fixed position.
2. Basic station (sink) is unlimited in terms of energy and processing.
3. All ad-hoc nodes can directly communicate with each other or with basic station (BASIC_STATION).
4. In both proposed algorithms, it is assumed that only one step exists from ordinary node to cluster head (C_HEAD) and from cluster head (C_HEAD) to basic station (BASIC_STATION).
5. All ad-hoc nodes are static and aware of location.
6. In both proposed algorithms, it is assumed that network nodes have been randomly distributed in the network space. Network nodes have equal processing and communication power and energy reserves (at the beginning of algorithm).

2.2 The first algorithm (pilot cluster head)

The first algorithm has three steps:

• Pilot clustering
• Formation of clusters
• Data transmission

In pilot clustering step, an initial clustering is done based on LEACH algorithm. Then, in the step of cluster formation, ad-hoc nodes select the best cluster head based on a function proposed in this research and are attracted by it. In this step, data is transmitted between ad-hoc nodes and cluster head and between cluster head nodes to basic station.

2.2.1 Pilot Clustering

First, the basic station (BASIC_STATION) distributes a percentage of needed cluster heads (C_HEAD) in the entire network. This percentage is shown by P. Also, BASIC_STATION distributes the data relating to the location of all ad-hoc nodes in the entire network.

After receiving the data, all ad-hoc nodes select a random number between 0 and 1 and compare it with T(n). If the selected number is smaller than T(n), the node is selected as cluster head for the current round. Otherwise, it is considered as ordinary node. In T(n) equation, P is the percentage of needed cluster heads, G is total nodes which compete for becoming cluster head (CH position), and r is the number of rounds (repetition). The aforementioned conditions determine which node is cluster head and which node should be considered as ordinary node.

2.2.2 Formation of Clusters

In this step, the best cluster head is selected from among the pilot cluster heads based on the proposed utility function. In the proposed algorithm, the following criteria are used for selecting the best cluster head. Since the proposed method is implemented on a single-phase basis, path length energy is total remaining energy of the origin node together with the remaining energy of cluster head. The remaining node energy is used for proper distribution of energy consumption. The distance criterion is used to include the nodes which are closer to cluster head. In other words, a node is more likely to be included in cluster if it is closer to the cluster head.

2.2.3 Selection of the Best Cluster Head

After the step of pilot clustering, a number of cluster heads are determined. These cluster heads distribute a control package in the network for attracting ordinary nodes. This package includes the remaining energy of the cluster head. The ordinary nodes in the network consider cluster heads as nest and try to select the best cluster head (C_HEAD) for laying eggs. They select the best cluster head based on the proposed utility function.

In the proposed utility function, E_{CH} is the remaining energy of the cluster head, E_{Node} is the remaining energy of origin node (ordinary node), and $D_{\mathrm{CH,\ Node}}$ is the distance between cluster head and ordinary node. Total remaining energy of the origin node and cluster head energy constitute total path length energy. The distance criterion enables to include the nodes which are closer to cluster head. FU value node compares its own cluster head (FU(own_ID) with F value of other cluster heads (FU(n)). The node with the highest utility function (FU) will be selected as the best cluster head. The ordinary nodes send their data to the best cluster head based on TDMA schedule.

2.2.4 Data Transmission

Once the clusters are formed and the nodes select the best cluster head, the step of data transmission begins. In this step, cluster heads send the data, which have been collected from the nodes based on TDMA schedule, to the final station. FU function selects the best cluster head on a random basis. For example, the best cluster head is selected with the probability of 0.8 and one cluster head is selected with probability of 0.2.

The pilot cluster head steps:

Step 1: Sensor nodes are randomly created in the network.
Step 2: Control Packet is sent in the range of 105 m
Step 3: *If* (The central station has received the control Packet)
then
The Packet is destroyed
exit
Step 4: *T(n)* function is computed.
Step 5: The random number is created.
Step 6: *if* (**Step 5** > **Step 4**) *then*
The Packet is destroyed
exit
Step 7: The number of N nests (pilot cluster head) are created.
Step 8: The created nests (pilot cluster heads) send the control Packet throughout the network.
Step 9: The nodes receive the control Packet and *F(n)* function is computed for selecting the best nest.
Step 10: *F(nown_id)* function is computed
Step 10: *if* (**Step 9** < **Step 10**) *OR* (distance greather than 25)
then
Go to **Step 1**:
Step 11: The node selects the best nest (best cluster head).
Step 12: The nodes send their data to the best nest (best cluster head).
Step 13: The nests (cluster heads) collect the data based on TDMA schedule and send them to the central station.
Step 14: The central station collects the data Packets.

2.3 The second algorithm: optimal cluster head (random)

While the first algorithm offers a good clustering method, the proposed function selects the cluster heads on a pilot basis. In cuckoo algorithm, all cuckoos select the optimal nest and move towards it. For this purpose, after the step of pilot clustering, each nest (pilot cluster head) creates a random number which is placed inside the control package. The created nests (pilot cluster heads) send the control package throughout the network. The nodes receive the control package and each node produces a random number between 0 and 1. Then, the algorithm compares the random number produced by the node with the random number of the package. Finally, if the random number of the node is smaller than the random number of package and the computed FU(n) function has the biggest value, the cluster head is accepted. Therefore, the best cluster heads are selected randomly, which results in the proper distribution of the cluster heads (best cluster heads) and prevents the frequent selection of one node as the best cluster head which would cause the waste of energy and early death of cluster head.

The optimal cluster head steps:

Step 1: Sensor nodes are randomly created in the network.
Step 2: Control Packet is sent in the range of 105 m
Step 3: *If* (The central station has received the control Packet)
then
The Packet is destroyed
exit
Step 4: *T(n)* function is computed.
Step 5: The random number is created.
Step 6: *if* (**Step 5** > **Step 4**) *then*
The Packet is destroyed
exit
Step 7: The number of N nests (pilot cluster head) are created.
Step 8: The nests (pilot cluster heads) generate a random number which is placed in the control package. The created nests (pilot cluster heads) send the control package throughout the network..
Step 9: The nodes receive the control package
Step 10: F(n) function is computed for selecting the best nest.
Step 11: Each node generates a random number.
Step 12: *F(nown_id)* function is computed
Step 10: *if* (**Step 10** < **Step 12**) *OR* (distance greater than 25) *OR (*Package random number >= random number*)*
then
Go to **Step 1**:
Step 11: The node selects the best nest (best cluster head).
Step 12: The nodes send their data to the best nest (best cluster head).
Step 13: The nests (cluster heads) collect the data based on TDMA schedule and send them to the central station.
Step 14: The central station collects the data Packets.

3. Results

In this part, we first compare the proposed method with other popular algorithms and then compare the two proposed algorithms. We compared the first proposed algorithm with LEACH [24], FSCA [25], GSAGA [26] and CORDF [27] using MATLAB software. Table 3 contains the parameters used in the simulation.

Table 3: Simulation Parameters

Percentage of becoming cluster head	0.05
Initial energy of node	2
Electronic energy	50
Data aggregation energy	5
Node heterogeneity percentage	0.2
Maximum number of rounds	2000
Basic station location	At center/left corner

4.1 Comparison of the first dead node

Table 4 represents death duration of the first node (energy loss) in each round of the network for the aforementioned methods in different environments.

Table 4: The first dead node for the networks with varied sizes and node number

Network parameters	Algorithm type				
150 node with network size: 250×250	LEACH	FSCA	GSAGA	CORDF	Pilot cluster head
150 node with network size: 250×250	78	92	118	210	324
175 node with network size: 300×300	53	59	90	195	249
200 node with network size: 400×400	15	17	42	180	198

4.2 Energy Consumption

Figure 1 compares the proposed method with other algorithms in terms of energy reduction. As you can see, pilot cluster head (P1) offers better energy rate than other methods.

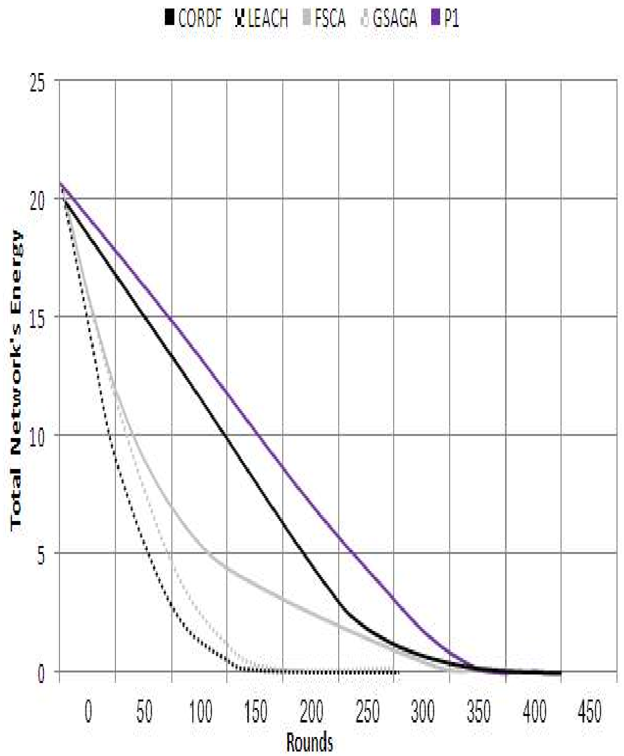


Figure 1: Network energy rate reduction with the size of 400×400 and node number of 200

4.3 The Number of Live Nodes

As you can see in Figure 2, the proposed method has improved the efficiency in terms of network lifetime. Compared with other methods, the pilot cluster head method increases the death time of the first node. Even after the first node of the network dies, if the network allows activity until all nodes consume their energy, the number of passed rounds will still be high.

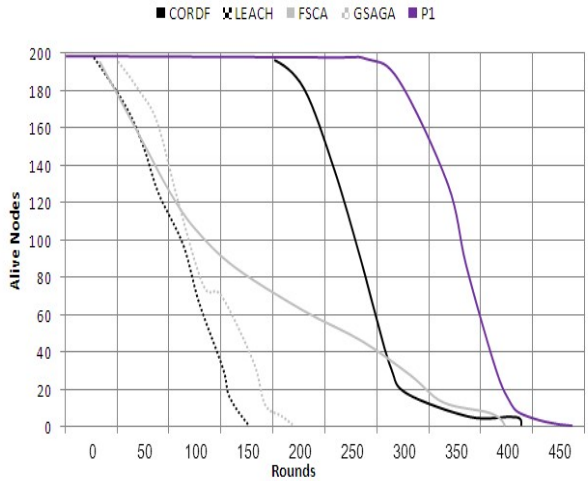
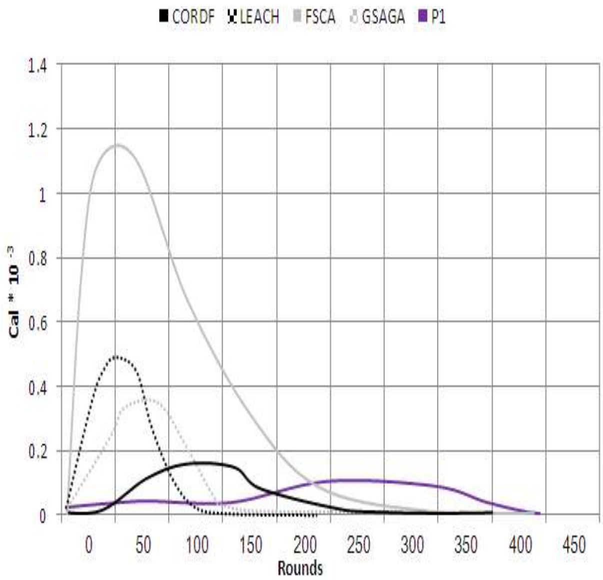


Figure 2: Comparison of the number of live nodes in the network with the size of 400×400 and node number of 200

4.4 Computation Volume

Computation volume is an important parameter in clustering wireless ad-hoc networks. High computation volume results in more energy consumption and reduction of node lifetime. Figure 3 illustrates the comparison of the proposed method with other four methods. As you can see, the proposed method offers shorter computation volume, which confirms its advantage over other methods.

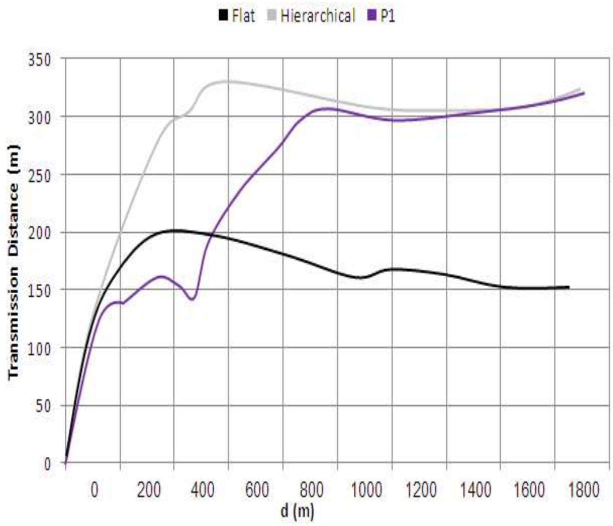


Manuscript received September 5, 2026
Manuscript revised September 20, 2026
https://doi.org/10.22937/IJCSNS.2026.26.9.3

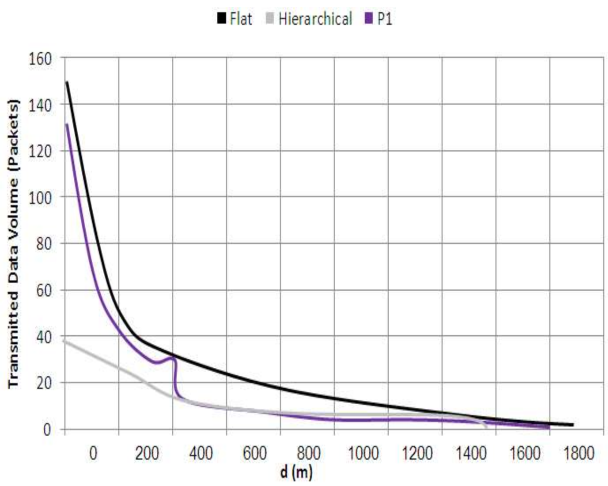
Figure 3: Computation volume in the network with the size of 400×400 and node number of 200

4.5 Distance from Sink

Figure a4 illustrates the average distance of each node from the sink. As you can see, the proposed method bears a bigger transmission distance compared to flat and hierarchical methods. It should also be noted that the proposed method offers varied internal and external communication distances, which is explained by varied algorithms accepted in each area. There is also a fluctuation in transmission distance, which may be explained by the changes in clustering algorithms. Figure b4 illustrates the average volume of the data transmitted by each node with the distance of *d* meters from the sink. As you can see, the proposed method bears the highest data transmission volume compared with hierarchical [28] and flat [29] methods. Moreover, data volume increases as *d* gets closer to the sink node.



(a)



(b)

Figure 4: Comparison of performance of three different clustering algorithms

4.6 Quality of Formed Clusters

We assessed the quality of formed clusters using Dunn and Silhouette standards. As you can see in Figure 5, the optimal method has higher Dunn and Silhouette coefficients than the pilot method, which confirms the better quality of formed clusters in the proposed method.

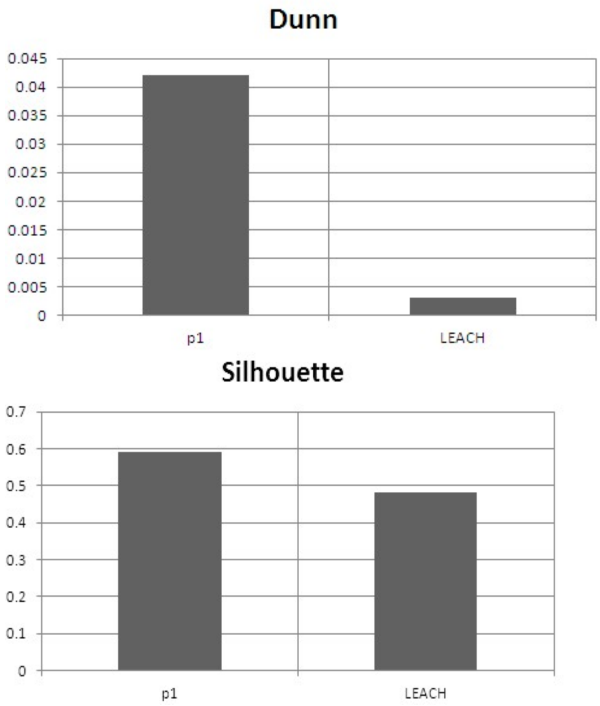


Figure 5: Clustering standards

4.7 Different Distributions

We examined the pilot cluster head method (p1) with optimal cluster head method (p2) using normal, uniform, exponential and grid distributions. As you can see in Figures 6-9, the first method offers better clustering protocol, which may be explained by the selection of centralized cluster heads which provides better load balance in the network nodes. In both protocols, the increased number of nodes results in the diminished performance, which may be explained by the conflict of data and the needed time for retransmission of messages. When the station is located at the corner, the performance of normal, uniform and grid distributions significantly declines, while the performance of exponential topology increases due to location of the basic station at the corner in which the nodes have gathered. According to simulation results contained in Table 2, in military applications in which the death of the first node is an important parameter and the network becomes unreliable after the first node dies, the performance of the second method is better when the basic station is located at the center compared to when it is located at the corner. In both algorithms, the first node dies later than in normal, exponential and grid distributions. In certain applications in which network survival is important even with only one node, the basic station is better to be located at the corner. It should be noted that LND>2000 means that only one node has survived until the end of the last round (2000). With the reduction of initial energy from 0.2 to 0.1, the network remains alive up to 1000 rounds.

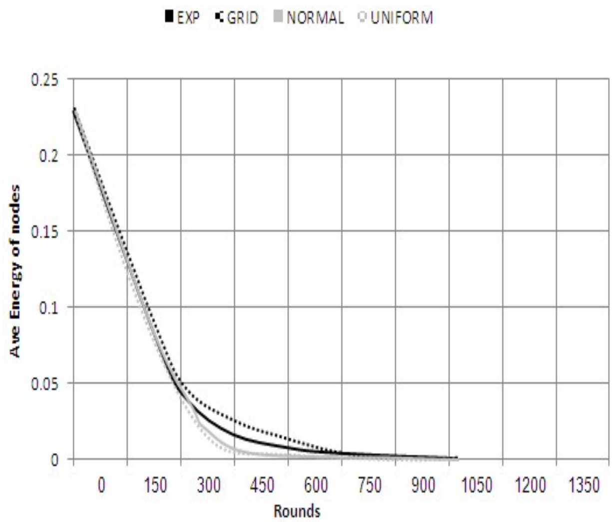


Figure 6: Average energy consumption of P1 nodes with four different distributions

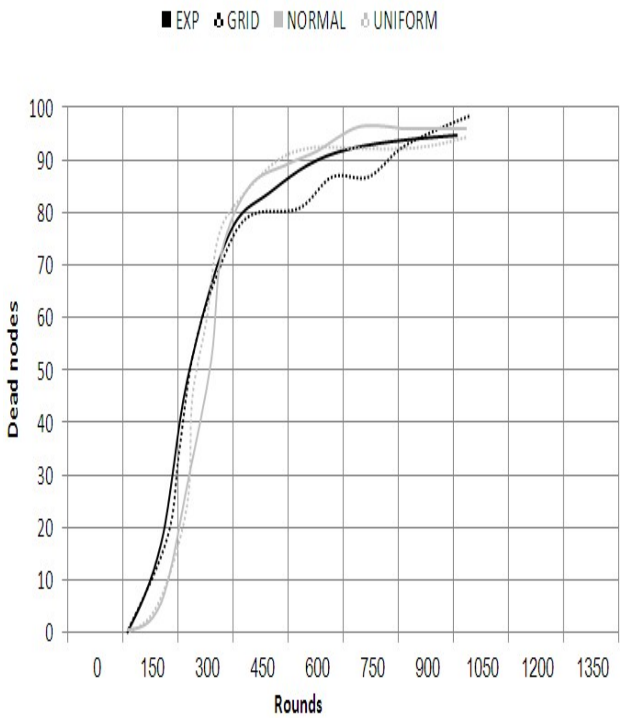


Figure 7: The number of dead nodes of P1 with four different distributions

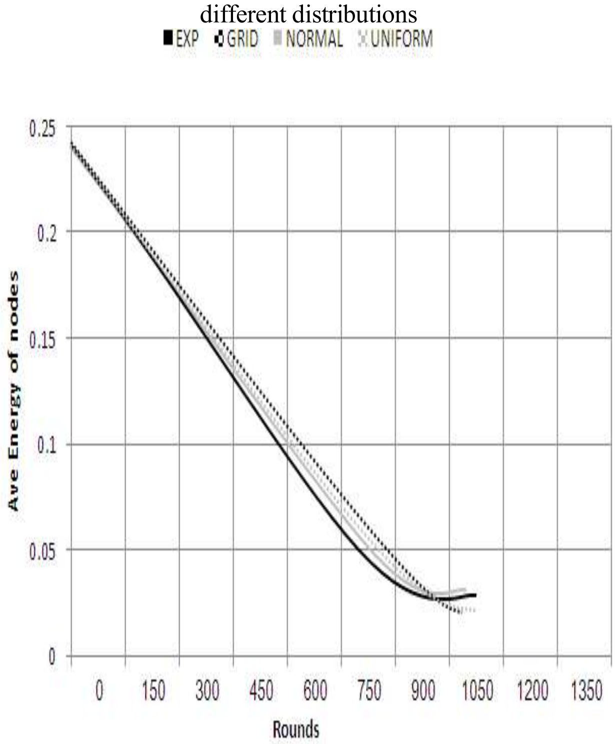


Figure 8: Average energy consumption of P2 nodes with four different distributions

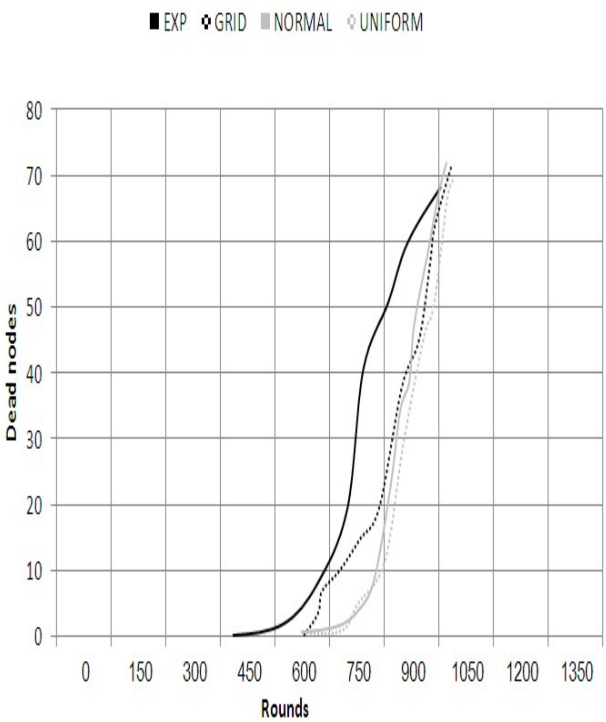


Figure 9: The number of dead nodes of P2 with four different distributions

Table 5: Different distributions

P2	Basic station at center			Basic station at corner	P1	Basic station at center			Basic station at corner
Normal	FND	516	575	239	Normal	FND	203	197	108
	HND	817	804	695		HDN	342	336	257
	LND	1255	>2000	>2000		LND	1744	>2000	>2000
Uniform	FND	666	728	266	Uniform	FND	221	208	98
	HDN	928	979	598		HDN	341	340	248
	LND	>2000	>2000	>2000		LND	1613	>2000	>2000
Exponential	FND	541	528	482	Exponential	FND	180	147	160
	HDN	876	734	980		HDN	315	311	324
	LND	>2000	1922	>2000		LND	>2000	>2000	>2000
Grid	FND	559	837	178	Grid	FND	180	151	58
	HDN	916	802	655		HDN	336	315	226
	LND	1997	>2000	>2000		LND	>2000	>2000	>2000

4.8 Data Generation Rate

Figure 10 illustrates the impact of data generation rate on data delivery rate in both methods. As you can see, the increased data generation rate results in the diminished performance in both protocols. This may be explained by the conflict of MAC layer, fast consumption of the limited network resources, and the increased data generation rate. The second method consumes less energy by reducing the number of message copies. Moreover, due to the use of optimal cluster head in the network, the load balance between station nodes is distributed and the amount of traffic and conflict declines. In the first method, due to the use of pilot cluster head and abundant copies of a message, a lot of traffic and conflict exists.

As you can see in Figure 11, the increased data generation rate results in the increased data delivery delay rate in both protocols. The increase of data delivery delay rate in the second algorithm is less than in the first algorithm, because the optimal cluster head reduces the network traffic. Moreover, this algorithm selects the nodes which not only are more likely to be delivered but also move in line with the selected station.

4.9 Density of ad-hoc nodes

In this part, the number of nodes varies from 25 to 200. Figure 12 illustrates the impact of the number of nodes on data delivery rate in both methods. The more the number of nodes in the network, the better results the second method produces compared to the first method. In pilot cluster head method, the increased number of nodes results in the increased number of data messages and the increased conflict. But optimal cluster head method reduces the amount of conflict. As you can see in Figure 13, the increased number of ad-hoc nodes results in the increased delivery delay rate in both methods. In the second method, each node reduces the data delivery rate by selecting the optimal cluster head and checking the probability of delivery. Therefore, the proposed method offers less delivery delay rate compared to the first method.

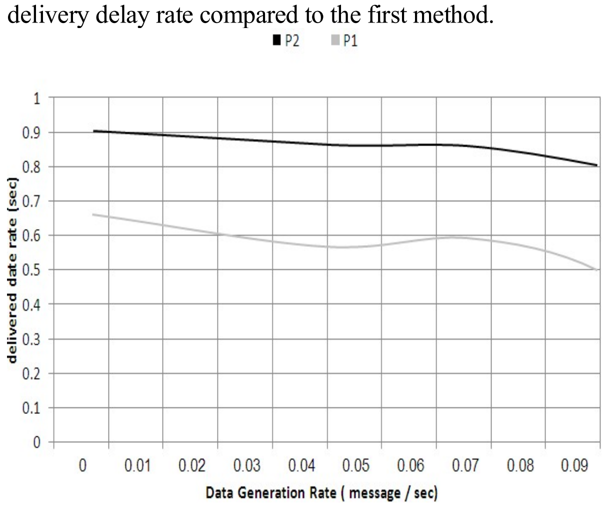


Figure 10: The impact of data generation rate on the average data delivery rate

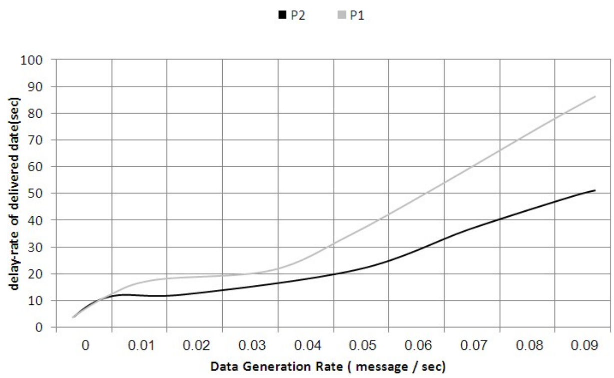


Figure 11: The impact of data generation rate on the average data delivery delay rate

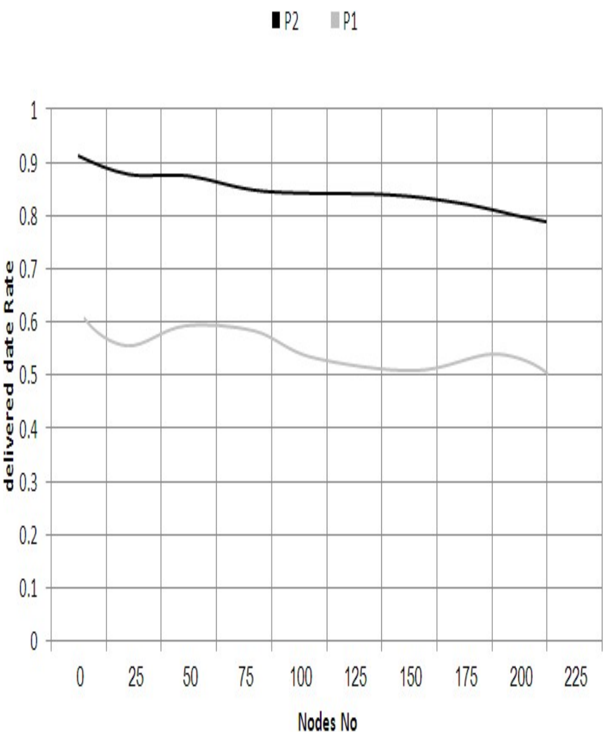


Figure 12: The impact of note density on the average data delivery rate

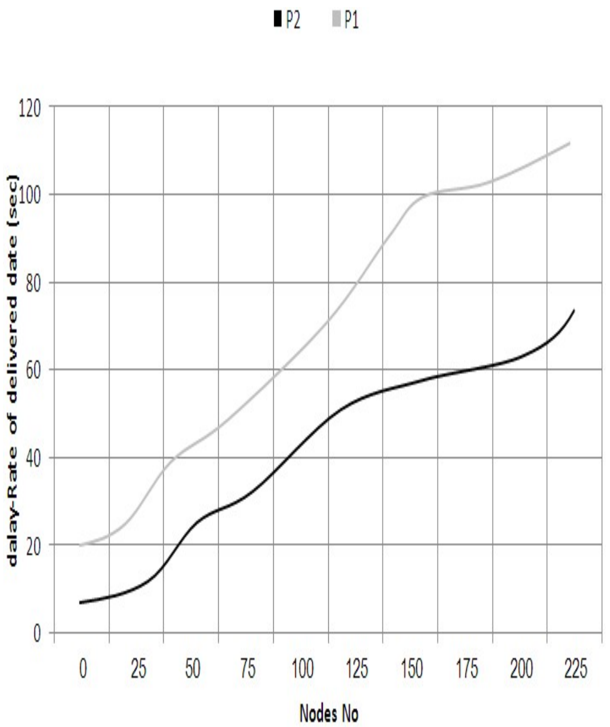


Figure 13: The impact of node density on the average data delivery rate

4.10 Energy Harvesting

In the first scenario, we changed the number of energy harvesting nodes from 5% to 30% of total nodes in order to determine the number of energy harvesting nodes. We selected energy harvesting rate of $60\mu J/_S$ and assumed that all nodes with energy harvesting capability had equal energy harvesting rate. Figures 14-15 illustrate the changes of lifetime in different numbers of energy harvesting nodes.

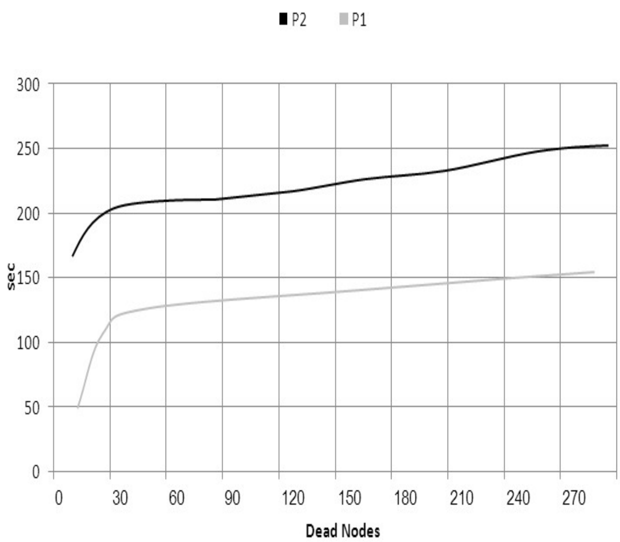


Figure 14: The changes of network lifetime with equal energy harvesting rate (5% of nodes have energy harvesting capability)

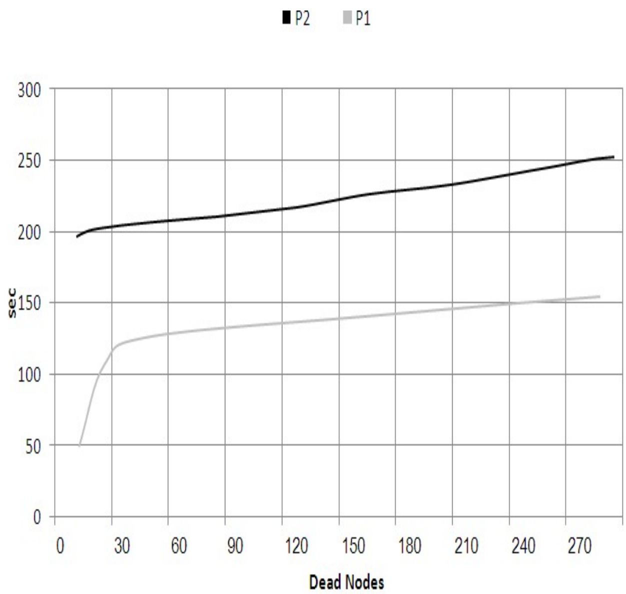


Figure 15: The changes of network lifetime with equal energy harvesting rate (33% of nodes have energy harvesting capability)

Energy harvesting rate of nodes may vary due to the impact of environment on wireless ad-hoc networks. Further tests indicated that energy harvesting rate varied from one node to the other. In the second scenario, we changed the energy harvesting rate of the nodes from $56\mu^{J}/_{S}$ to $60\mu^{J}/_{S}$ of total nodes in order to determine the impact of energy harvesting rate on the network lifetime. As with the first scenario, we increased the number of energy harvesting nodes from 5% to 30%. Figures 16-17 illustrate the changes of lifetime in different energy harvesting rates with varied number of energy harvesting nodes.

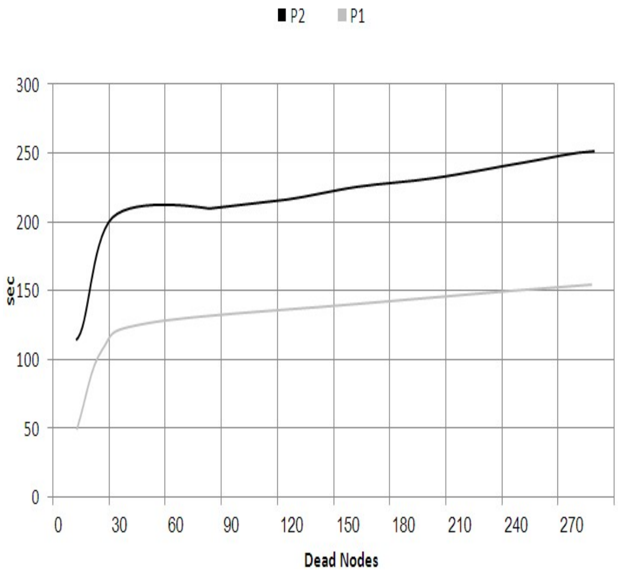


Figure 16: The changes of network lifetime in different energy harvesting rates (5% of nodes have energy harvesting capability)

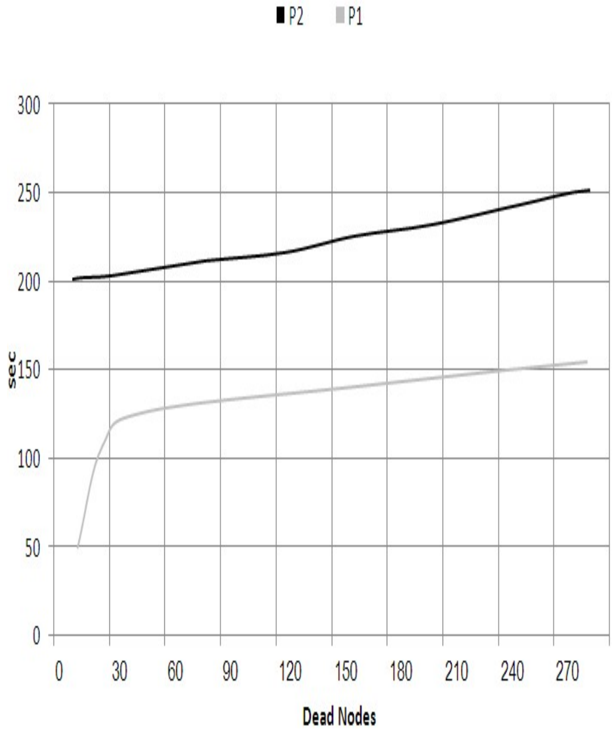


Figure 17: The changes of network lifetime in different energy harvesting rates (30% of nodes have energy harvesting capability)

The simulation results represented in figures 14-17 indicate that optimal clustering method offers a better lifetime than the pilot method in terms of the death of the first and last nodes. The pilot clustering technique does not consider the energy harvesting capability of the nodes. The simulation results indicated that a part of the nodes had energy harvesting capability. The pilot method considers the energy harvested by a node during the network operation time as a remaining accumulated energy in the process of re-clustering. But it does not consider the momentary energy harvesting rate in the process of initial clustering. In addition, the pilot method only considers a specific type of cost and overlooks their combination. On the other hand, the optimal method makes a balance between intra-cluster communication cost, net energy harvesting rate, vicinity, remaining energy, and maximum energy saving capacity. Thanks to its advantages such as selection of healthy nodes and the most effective energy harvesting node in re-clustering processes, this method increases the lifetime of cluster head node at the beginning of each process. Consequently, it reduces the number of re-clustering processes which is a major challenge in clustering methods. Energy protection has always been a major challenge in ad-hoc networks. To solve this problem, modern wireless nodes were developed with the capability to harvest energy from environmental resources. Since cluster heads need to process and transmit data for all cluster members, they may have a short lifetime.

4. Conclusion

Optimization of energy consumption in wireless ad-hoc networks may increase the network lifetime. Clustering is an efficient method for increasing the network lifetime and reducing energy consumption. So far, many clustering algorithms have been proposed, but none has proved efficient in all criteria. This paper proposed an algorithm with a good efficiency in the majority of these criteria. The proposed algorithm selects cluster heads based on path length energy. We also proposed another algorithm for the proper distribution of selected cluster heads. The second algorithm randomly selects the best cluster head and prevents the frequent selection of one node as the best cluster head which would result in the waste of energy and early death of cluster heads. We compared the first method with LEACH, FSCA, GSAGA and CORDF in terms of the first dead node, energy consumption, the number of live and dead nodes, computation volume, and clustering indices. Finally, we compared the second algorithm with the first one in terms of distribution, generation rate, data delivery delay, and energy harvesting. The results confirmed that:

- The pilot cluster head method reduces energy consumption more uniformly compared to other methods.
- Compared with other methods, the pilot cluster head method increases the death time of the first node. Even after the first node of the network dies, if the network allows activity until all nodes consume their energy, the number of passed rounds will still be high.
- The proposed method offers less computation volume compared to other methods, which confirms the advantage of this method over other methods.
- The proposed method bears a bigger transmission distance compared with flat and hierarchical cluster head methods.
- The optimal method offers higher Dunn and Silhouette coefficients compared to the pilot method, which confirms the better quality of formed clusters in the proposed method.
- The first method offers better clustering protocol, which may be explained by the selection of centralized cluster heads which provides better load balance in the network nodes. In both protocols, the increased number of nodes results in the diminished performance, which may be explained by the conflict of data and the needed time for retransmission of messages. When the station is located at the corner, the performance of normal, uniform and grid distributions significantly declines, while the performance of exponential topology increases due to location of the basic station at the corner in which the nodes have gathered.
- In both protocols, the increased data generation rate results in the diminished performance. This may be explained by the conflict of MAC layer, fast consumption of the limited network resources, and the increased data generation rate. The second method consumes less energy by reducing the number of message copies. Moreover, due to the use of optimal cluster head in the network, the load balance between station nodes is distributed and the amount of traffic and conflict declines. In the first method, due to the use of pilot cluster head and abundant copies of a message, a lot of traffic and conflict will exist and there will be a low data delivery rate.
- The increased data generation rate results in the increased data delivery delay rate in both protocols. The increase of data delivery delay rate in the second algorithm is less than in the first algorithm, because the optimal cluster head reduces the network traffic. Moreover, this algorithm selects the nodes which not only are more likely to be delivered but also move in line with the selected station.
- In pilot cluster head method, the increased number of nodes results in the increased number of the delivered data messages and the increased data density and conflict. But optimal cluster head method reduces the amount of conflict.
- In both modes, the optimal clustering method offers a better lifetime compared to the pilot method in terms of the death of the first and last nodes.

References

[1] Wei, L. (2016). EACC: an Improved Clustered Wireless Sensor Network Scheme

[2] Balakrishnan, A., & Rino, P. C. (2015, September). A Novel Anomaly Detection Algorithm for WSN. In 2015 Fifth International Conference on Advances in Computing and Communications (ICACC) (pp. 118-121). IEEE

[3] Ashitha, C., John, S. P., & Simpson, S. V. (2016). Intrusion Detection System for a Cluster Based WSN. Wireless Communication, 8(7), 274-277

[4] Karaoglu, B., & Heinzelman, W. (2015). Cooperative load balancing and dynamic channel allocation for cluster-based mobile ad hoc networks. IEEE transactions on mobile computing, 14(5), 951-963

[5] Farivar, R., Fazeli, M., & Miremadi, S. G. (2005, August). Directed flooding: a fault-tolerant routing protocol for wireless sensor networks. In 2005 Systems Communications (ICW'05, ICHSN'05, ICMCS'05, SENET'05) (pp. 395-399). IEEE

[6] Li, L., & Halpern, J. Y. (2001, June). Minimum-energy mobile wireless networks revisited. In Communications, 2001. ICC 2001. IEEE International Conference on (Vol. 1, pp. 278-283). IEEE

[7] Vardhan, S., Wilczynski, M., Portie, G. J., & Kaiser, W. J. (2000). Wireless integrated network sensors (WINS): distributed in situ sensing for mission and flight systems. In Aerospace Conference Proceedings, 2000 IEEE (Vol. 7, pp. 459-463). IEEE

[8] Hedetniemi, S. M., Hedetniemi, S. T., & Liestman, A. L. (1988). A survey of gossiping and broadcasting in communication networks. Networks, 18(4), 319-349

[9] Braginsky, D., & Estrin, D. (2002, September). Rumor routing algorthim for sensor networks. In Proceedings of

the 1st ACM international workshop on Wireless sensor networks and applications (pp. 22-31). ACM

[10] Lindsey, S., & Raghavendra, C. S. (2002). PEGASIS: Power-efficient gathering in sensor information systems. In Aerospace conference proceedings, 2002. IEEE (Vol. 3, pp. 3-1125). IEEE

[11] Culpepper, J., Dung, L., & Moh, M. (2003, October). Hybrid indirect transmissions (HIT) for data gathering in wireless micro sensor networks with biomedical applications. In Computer Communications, 2003. CCW 2003. Proceedings. 2003 IEEE 18th Annual Workshop on (pp. 124-133). IEEE

[12] Lee, M., & Wong, V. W. (2005, December). LPT for data aggregation in wireless sensor networks. In GLOBECOM'05. IEEE Global Telecommunications Conference, 2005. (Vol. 5, pp. 6-pp). IEEE

[13] Amini, N., Fazeli, M., Miremadi, S. G., & Manzuri, M. T. (2007, May). Distance-based segmentation: an energy-efficient clustering hierarchy for wireless microsensor networks. In Fifth Annual Conference on Communication Networks and Services Research (CNSR'07) (pp. 18-25). IEEE

[14] Heinzelman, W. R., Chandrakasan, A., & Balakrishnan, H. (2000, January). Energy-efficient communication protocol for wireless microsensor networks. In System sciences, 2000. Proceedings of the 33rd annual Hawaii international conference on (pp. 10-pp). IEEE

[15] Prabhu, S. B., & Sophia, S. (2011). A survey of adaptive distributed clustering algorithms for wireless sensor networks. International Journal of Computer Science and Engineering Survey, 2(4), 165

[16] Li, C., Ye, M., Chen, G., & Wu, J. (2005, November). An energy-efficient unequal clustering mechanism for wireless sensor networks. In IEEE International Conference on Mobile Adhoc and Sensor Systems Conference, 2005. (pp. 8-pp). IEEE

[17] Lindsey, S., & Raghavendra, C. S. (2002). PEGASIS: Power-efficient gathering in sensor information systems. In Aerospace conference proceedings, 2002. IEEE (Vol. 3, pp. 3-1125). IEEE

[18] Mahmoudi, S., Rajabioun, R., & Lotfi, S. (2013). Binary Cuckoo Optimization Algorithm. nature

[19] Dhivya, M., & Sundarambal, M. (2011). Cuckoo search for data gathering in wireless sensor networks. International Journal of Mobile Communications, 9(6), 642-656

[20] Zhang, S., Li, J., He, B., & Chen, J. (2016). LSDV-Hop: Least Squares Based DV-Hop Localization Algorithm for Wireless Sensor Networks. Journal of Communications, 11(3)

[21] Dhivya, M., Sundarambal, M., & Anand, L. N. (2011). Energy efficient computation of data fusion in wireless sensor networks using cuckoo based particle approach (CBPA). Int'l J. of Communications, Network and System Sciences, 4(04), 249

[22] Yang, X. S., & Deb, S. (2009, December). Cuckoo search via Lévy flights. In Nature & Biologically Inspired Computing, 2009. NaBIC 2009. World Congress on (pp. 210-214). IEEE

[23] Rajabioun, R. (2011). Cuckoo optimization algorithm. Applied soft computing, 11(8), 5508-5518

[24] Tashtoush, Y. M., & Okour, M. A. (2008, December). Fuzzy self-clustering for wireless sensor networks. In Embedded and Ubiquitous Computing, 2008. EUC'08. IEEE/IFIP International Conference on (Vol. 1, pp. 223-229). IEEE

[25] Tashtoush, Y. M., & Okour, M. A. (2008, December). Fuzzy self-clustering for wireless sensor networks. In Embedded and Ubiquitous Computing, 2008. EUC'08. IEEE/IFIP International Conference on (Vol. 1, pp. 223-229). IEEE

[26] Zhang, J., Lin, Y., Zhou, C., & Ouyang, J. (2008, December). Optimal model for energy-efficient clustering in wireless sensor networks using global simulated annealing genetic algorithm. In Intelligent Information Technology Application Workshops, 2008. IITAW'08. International Symposium on (pp. 656-660). IEEE

[27] R. Taimourei Yenesary and E. Saeedian, "CORDF: cluster heads optimum choice and route discovery in multi-hop connection by using fuzzy logic in wireless sensor networks," American J. ofScientific Research, vol. 64, no. 1, pp. 84-90, Jun. 2012.

[28] Chaubey, Nirbhay K., and Dharti H. Patel. "Energy Efficient Clustering Algorithm for Decreasing Energy Consumption and Delay in Wireless Sensor Networks (WSN)." Energy 4, no. 5 (2016)

[29] Rajeshwari, P., Shanthini, B., & Prince, M. (2015). Hierarchical Energy Efficient Clustering Algorithm for WSN. Middle-East Journal of Scientific Research, 23, 108-117