Engineering and Applied Science Research 2021;48(5):529-536 Research Article

EASR

Engineering and Applied Science Research

https://www.tci-thaijo.org/index.php/easr/index

Published by the Faculty of Engineering, Khon Kaen University, Thailand

LBCM: Energy-efficient clustering method in wireless sensor networks

Hifzan Ahmad*[1)] and Narendra Kohli[2)]

[1)]Dr. A. P. J. Abdul Kalam Technical University, Lucknow, India
[2)]Department of Computer Science, Harcourt Butler Technical University, Kanpur, India

Received 23 September 2020
Revised 11 January 2021
Accepted 15 February 2021

Abstract

Wireless sensor networks (WSNs) are a collection of battery-powered sensor nodes deployed in an environment that is most of the time isolated in nature. Usually, the power consumes in WSNs is due to sensing or sensed data forwarding. The data forwarding operation requires communication among nodes and its forwarding node (usually cluster head). Therefore, the design of an algorithm which can make a cluster and chooses a cluster head play a vital role in WSNs. Our objective is to perform energy-efficient communication between sensor nodes and broaden the network's lifetime by balancing the load of less energy-constrained gateways. Clustering is an effective technique to lessen the sensor nodes' energy dissipation in a wide-range wireless sensor network to increase the network's lifetime and obtain scalability and robustness. Though, if any of the gateways remain overloaded by a massive quantity of sensor nodes, it may fall soon, and the network's lifetime can end in a short duration. Therefore, it is necessary to adjust the gateways' load to prolong the network's lifetime. The paper includes a newly introduced algorithm named LBCM (Load Balanced Clustering Method) that adjusts the gateways' load and performs energy-efficient communication among the sensor nodes in WSNs. The simulation outcome of the introduced algorithm shows that our proposal is more energy-efficient than the existing algorithm.

Keywords: Load balancing, Energy efficiency, Clustering, Communication cost, Wireless sensor network

1. Introduction

Data gathering is a quickly developing and fast-growing discipline in today's age of computing. The sensor nodes produce a modest plus straightforward resolution for those applications, particularly within the ungracious and low-maintenance regions where standard methodologies end up being expensive. The Wireless Sensor Networks (WSNs) in the ongoing years has attracted a lot of consideration because of its latent capacity and a wide assortment of uses, for example, physical, ecological observing, battlefield surveillance, security, and disaster management [1]. A WSN includes countless low-power sensor nodes typically deployed in a non-uniform way within the region of interest to monitor the events and assemble the sensed data over the sensing field. These sensor nodes are small gadgets that are fit to quantify the occasion of a couple of events, for instance, changes in any physical measure like temperature, weight, pressure, etc., or movement of an item or environment over the sensing field [2, 3]. The sensor nodes in such systems are normally expendable and supposed to go on till their power diminishes. In this way, power is a scarce asset for such systems and must be overseen to broaden the life of the sensor nodes for the span of a specific strategy.

The clustering of sensor nodes is one of the compelling ways to moderate energy and has numerous benefits over ordinary strategies [4]. Numerous scientists have addressed the clustering as it can limit the redundant messages exchange among the sensor nodes. It is especially beneficial for those applications that require scalability and is quite useful in data gathering because it can decrease the interferences and adjust the load amongst sensor nodes [5]. The vast majority of the existing techniques of clustering depend on choosing a cluster head from the applicable standard sensor nodes that incorporate cluster-ID [6], residual energy [7], degree of connectivity [8, 9], and so forth. Nonetheless, vast numbers concerning those methods do not acknowledge load balancing amongst cluster heads. A few techniques alternate the job of cluster heads amongst the traditional sensor nodes to adjust the power utilization with re-clustering [10].

Many of these methods utilize multi-hop transmission to transmit the monitored data to the base station since they cannot maintain long-distance communication. Therefore, the hot spot problem persists, i.e., the closer to the sink's sensor nodes bring more inter-cluster traffic and, consequently, exhaust their power quicker than distant sensor nodes. Meanwhile, the path discovery and updating the path put an additional load on the network. The researchers introduced a new design in [11, 12] where less energy-constrained sensor nodes termed gateways have been set up to overwhelm such burdens and unbalanced dispersion of power in the system. The sensor nodes are gathered nearby such gateways to make clusters, as presented in Figure 1. The gateways are also accountable for maintaining sensor nodes beneath its cluster. The sensor nodes can communicate to only those gateways which belong to the same cluster as them. On using the method, as mentioned earlier, the overall communication load of the sensor nodes can be better administered. Though, the inappropriate distribution of the sensor nodes to the gateways in the clustering process can make some gateways overloaded by an increasing number of sensor nodes. The performance of the WSN can be demoted by such overload since

*Corresponding author. Tel.: +9188 4000 3817
Email address: hifzan@gmail.com
doi: 10.14456/easr.2021.55

the overloaded gateways utilize their power quickly, resulting in performing re-clustering, which is inconvenient for low-power sensor nodes. Moreover, if the sensor nodes are not balanced among cluster heads, it increases channel congestion, ultimately creating packet loss. The subsequent packet loss can cause repetition in communication, which will lead to less channel utilization. Further, packet loss and repetition will cause inefficient energy consumption and increases the overall delay.

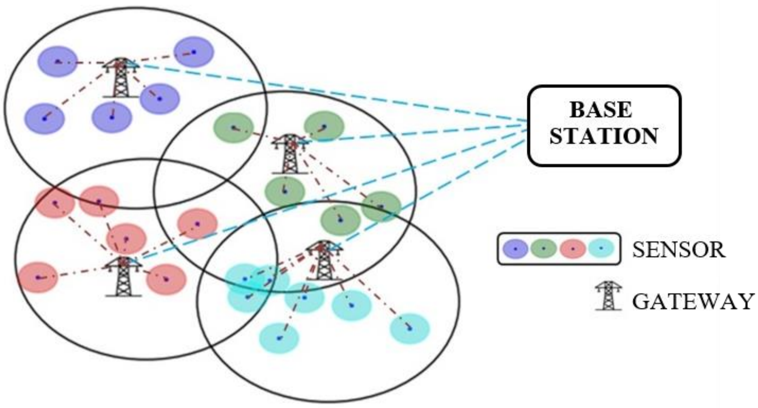

Figure 1 Gateway clustered WSN

The paper mainly focuses on the clustering of the sensor nodes nearby some gateways, including the next two goals; i) to balance the load of sensor nodes per gateway and ii) to minimize the communication energy dissipation of the sensor nodes within the clusters. We introduced a novel algorithm, namely LBCM (Load Balanced Clustering Method), which takes care of both the goals. The simulation results of LBCM reveal that it is more energy-efficient than the similar algorithm shortest distance clustering (SDC) [12], distributed energy-efficient load balancing algorithm (DELBA) [13], centralised energy-efficient load balancing algorithm (CELBA) [14], load-balanced clustering algorithm (LBCA) [12], and energy-efficient load-balanced clustering algorithm (EELBCA) [15].

The remainder of the paper is structured as follows. Section 2 reviews the related work, and in section 3, the system model has been discussed. Section 4 described and explained the proposed load-balanced clustering method (LBCM) algorithm, while section 5 illustrated the experimental results. Finally, the paper is concluded in section 6.

2. Related works

One of the most contemplated fields in the WSNs is clustering [4]. The principal objective of cluster-based routing protocol is to proficiently keep up the power dissipation of sensor nodes by affecting them in multi-hop communication inside a cluster and implementing data collection and fusion to minimize the number of forwarded information toward the base station and transmission range of sensor nodes.

The author described the cluster-based protocol, namely LEACH [10] and LEACH-C [16], where the selection of cluster heads is accomplished by the sink randomly. The cluster head determination is arbitrary in the LEACH algorithm, and the task of the CH is regularly switched amongst the sensor nodes present in the cluster. Consequently, choosing a sensor as a CH with deficient power may fall quickly and diminishes the network's lifetime. While in LEACH-C, the sink operates a simulated annealing algorithm to obtain the optimal solution with better places to decrease cluster heads' energy dissipation.

APTEEN [17] is an improved version of TEEN [18], with all the characteristics of a TEEN. The sensor nodes in the TEEN protocol continuously detect the signal and convey the information shortly to the sink at whatever points the information constraint outperforms the threshold value. APTEEN was exhibited concerning hybrid networks and captured both periodic data gathering and response to time-critical circumstances. TEEN's fundamental disadvantage is that if the detected value is not surpassing the threshold, the sensor nodes do not communicate, while in APTEEN, the sensor nodes forward information intermittently at nearly long delays while simultaneously sending information when the detected value surpasses its edge.

In shortest distance clustering (SDC) [12], a gateway adds sensor nodes in its cluster if the Euclidean distance between them is shortest. In the clusters formed by the SDC, the gateway's average energy consumption is minimum; however, the load of sensor nodes on the gateways is not balanced. The sensor nodes clustered by the SDC algorithm consume less communication energy initially but later on consumed more energy due to the burden of re-clustering. The experimental outcome reveals that the performance of SDC diminishes with the increase in the number of clusters.

Though the above-discussed algorithms have not examined the load balancing of sensor nodes in the clusters, the authors introduced a load-balanced clustering algorithm, LBCA [12], concerning load balancing and communication energy issues. To keep the system's load-balanced, they prefer an objective function termed as the variance of load in the system. Nevertheless, the energy consumption of the sensor nodes during communication inside their cluster is still high.

In [14], the authors offered a load balancing algorithm CELBA that performs better than LBCA in terms of load balancing and energy dissipation. The algorithm is shown to be more efficient concerning the issues of load balancing and communication energy. They used an objective function designated as base mean cardinality during cluster development. The authors in [13] proposed a distributed version of CELBA named DELBA. It is proved to be more energy-efficient than CELBA concerning energy utilization.

The authors stated EELBCA [15] to discuss the energy efficiency and load balancing of the gateways in terms of their cardinality in WSNs. EELBCA is a clustering algorithm based on min-heap, which runs in O (n log m) time for n sensor nodes and m gateways. In the clustering phase, a min-heap containing the gateways formed in the order of smallest to the most extensive number of sensor nodes assigned to the gateways. Among the not allotted sensor nodes, a sensor node nearest to the root node and within its communication range is allotted to the root node of the min-heap. At that point, to maintain the min-heap property, the min-heap rearranges to obtain the minimum loaded gateway at the root. The algorithm proceeds with the same process until all the sensor nodes are assigned to the CHs.

3. System model

The multi-gateway model of the clustered sensor network is displayed in Figure 1. The network comprises just two sorts of nodes; sensor nodes and gateways. The gateway's primary function is to carry out data accumulation and transfer the accumulated data toward the base station. The overall communication is performed over the symmetric wireless links; that is, a wireless link between two sensor nodes is most effective when they are within the communication reach. The gateways can perform long-distance communication when contrasted with the sensor nodes, and the gateways need to be within the communication reach of one another. This paper expects all the sensor nodes and the gateways to be stable once they are deployed. All the sensor nodes and the gateway nodes are assigned unique IDs and the TDMA plan at the bootstrapping process. The positions of the gateway nodes are obvious and immoveable all through the network's lifetime. All the sensor nodes are thought to know about their position through certain GPS or utilizing some localization methods [11].

Table 1 Factors used in communication

Factor	Value (taken)	Meaning
$\boldsymbol{\beta_t}$	50 nJ/bit	Energy diminished in transmitter
$\boldsymbol{\beta_r}$	50 nJ/bit	Energy diminished in receiver
$\boldsymbol{\beta_{amp}}$	$100\ \mathrm{pJ/bit/m^2}$	Energy diminished in transmitter amplifier
P	10 Kbits	Data packet size
D	−	Distance traverse by the message

3.1 Energy model of sensor nodes

The model of energy dissipation for each sensor has been adopted from [12]. The critical energy factors for communication in the discussed model are the energy/bit diminishes via the transmitter, the energy depleted in the transmitter amplifier, and the energy/bit consumed by the receiver. Let the energy to transmit P bits is denoted by EN_{TP} and the energy to receive P bits is denoted by EN_{RP}. For passing through a distance D, a path loss of $1/D^2$ has been assumed. The parametric equations for energy consumption of the transmitter as well as receiver are expressed as Equation (1) and Equation (2). The typical value of each factor taken and its meaning is summarized in Table 1.

$$EN_{TP} = \left(\beta_t + \beta_{amp} * D^2\right) * P \quad (1)$$

$$EN_{RP} = \beta_r * P \quad (2)$$

The communication cost of the gateway (GW) is equivalent to the summation of the energy consumed in communication by all the sensor nodes that belong to the gateway GW and expressed as Com_{Cost}. Let us assume that $Cost(S_i, GW)$ is the cost to perform communication between the sensor S_i and the gateway GW, $EN_{TP}(P, D)$ is the energy utilized to transfer P bits of data across the distance D by the sensor S, and $EN_{RP}(P)$ is the energy spent to accept P bits of data by the gateway GW. Then the communication cost between the sensor S_i and the gateway GW, and the overall communication cost ($Com_{Cost}(GW)$) of the gateway GW is determined by the equations presented in Equation (3) and Equation (4) as follows:

$$Cost(S_i, GW) = EN_{TP}(P, D) + EN_{RP}(P) \quad (3)$$

$$Com_{Cost}(GW) = \sum_{i=1}^{n} Cost(S_i, GW) \quad (4)$$

where n is the number of sensor nodes that belong to the gateway GW.

4. Proposed algorithm

4.1 K-Means algorithm

It is a basic iterative clustering algorithm that depends on deciding the underlying number of clusters by characterizing the underlying centroid value [19]. It has numerous points of interest, such as simple implementation, quick convergence, and basic numerical ideas [20]. The number of clusters relies upon the *k*-value setting in the *k*-means algorithm [21]. In practice, the value of *k* is usually hard to describe. The determination of the *k* value legitimately decides the cluster that ought to be clustered into numerous clusters. This algorithm demands specific numbers in determining the number of clusters *k*, as the initial cluster center may be modified with the goal that this event may bring about insecure gathering of data [22]. The output of the algorithm relies upon the chosen center values on clustering.

The optimal position for the gateways in the field of interest is determined using the *k*-means clustering algorithm. The gateways deployed on their positions in the sensing field. The determined locations of the gateways stay fixed until the last sensor node of the network fall.

4.2 LBCM (Load-Balanced Clustering Method) algorithm

The primary purpose of LBCM is to cluster the sensor nodes efficiently around the gateways. The clustering facilitates network scalability to a large number of sensor nodes and prolongs the network's life by providing the sensor nodes to maintain energy in communication with neighboring sensor nodes and by balancing the load among the gateways. Clusters are formed based on the cost of communication and the load on the gateways.

The gateways find the sensor nodes that are placed within their transmission range in the bootstrapping process. The gateways broadcast a message indicating the start of the clustering process. In response, sensor nodes forward their locations and IDs straight to the gateways. Each gateway comprises the IDs and locations of sensor nodes in its transmission range set.

Algorithm 1: The pseudo-code of LBCM Algorithm

```
\\ calculate Avg. Euclidean Distance of each Gateway GW from all the deployed sensor nodes
for i = 1 to K do                 \\ No. of Gateways
    Euc_Dist_GWi = 0;
    for j = 1 to N do    \\ No. of sensor nodes
            Euc_Dist_GWi += Euc_Dist(GWi, Sj);
            GW_LISTi = (Sj, Euc_Dist(GWi, Sj)) ;
    end
    Avg_Euc_Dist_GWi = Euc_Dist_GWi / N;
end
\\sorting the gateways based on their Avg. Euclidean distance in descending order
GWk = Sortdesc(Avg_Euc_Dist_GWk);
\\sorting the gateway list based on their Euclidean distance in ascending order
GW_LISTi = Sortasc(GW_LISTi);
\\initialize the load of each Gateway
for i = 1 to K do
    LOAD[GWi] = 0;
end
\\adding node to each gateway one by one
LBF = Ceil(SN/GK);    \\where N is no. of sensor nodes and K is no. of gateways
for i = 1 to LBF do
    for j = 1 to K do
            if (GW_LISTj(Sj) ∉ LOAD[GWK]) then
    \\sensor Sj add in a cluster of GWj when it does not belong to any other GWK
                    LOAD[GWj] += GW_LISTj(Sj);
            end
            else
                    While(GW_LISTj(++Sj) ∉ LOAD[GWK]) do
                            LOAD[GWj] += GW_LISTj(Sj);
                            break;
                    end
            end
    end
end
```

A sensor node S_j refers to the range set TRSet of gateway GW_i if the following conditions hold:

$$S_j \in TRSet_{GWi} \; iff \;\; TR_{GWi} \geq Euc\_Dist(S_j, GW_i) \;\; AND \; TR_{Sj} \geq \; Euc\_Dist(S_j, GW_i) \tag{5}$$

where, TR_{GWi} and TR_{Sj} signify the transmission range of gateway GW_i and sensor node S_j, respectively, and $Euc\_Dist(S_j, GW_i)$ signifies the Euclidean distance between the sensor node S_j and the gateway GW_i. The $TRSet$ of a gateway contains the IDs of its sensor nodes and their Euclidean distance from it.

To balance the load of sensor nodes on each gateway, a load balancing factor (LBF) has expressed in Equation 6 as follows:

$$\text{LBF} = \text{Ceil}(\frac{S_N}{GW_K}) \tag{6}$$

where, S_N and GW_K is the number of sensor nodes and gateways deployed in the sensing field, and Ceil( ) is a function that returns the smallest integer value that is bigger than or equal to a number.

In the clustering process, each gateway calculates the average Euclidean distance from the sensor nodes belongs to the $TRSet_i$ of a gateway GW_i. The order in which gateways start to join the sensor nodes in its cluster will be the decreasing order of their average Euclidean distance. The gateway GW_i adds sensor nodes sequentially from $TRSet_i$ to its cluster in increasing order of Euclidean distance. Before adding sensor nodes to their cluster, all the gateways must ensure that the number of sensor nodes belongs to their cluster should be less than or equal to the LBF. This step continues until the number of non-clustered sensor nodes is equal to zero. Initially, the number of non-clustered sensor nodes is equivalent to the number of sensor nodes deployed. When the clustering process is finished, all the sensor nodes are notified about the cluster's ID, and every sensor node refers to just one cluster. During inter-cluster communication, all the data are forwarded through the gateways. The pseudo-code of the suggested LBCM is outlined in Algorithm 1.

The silhouette coefficient used for cluster validation decides how well a data point is clustered, and it calculates the average distance between clusters. It is a measurement used to evaluate the integrity of a clustering method. Its score varies from -1 to 1. A score of 1 show that the clusters are very dense and well separated from one another. A score near 0 designates that the clusters are overlapped. The worst score is -1, which shows that the clusters are assigned incorrectly. Our algorithm's silhouette score is 0.41258 to 10 clusters, which designates that the clustering is performed in the right way, as depicted in Figure 2. The silhouette score is low because of the energy-efficient load balancing of sensor nodes on the gateways.

5. Experimental results

The potency of our proposed clustering method has been verified through simulation. This segment represents the simulation conditions, performance metrics, and experimental outcomes. The experiments performed on Spyder 3.3.3 IDE with an Intel(R) Core(TM) i3-2350M CPU @ 2.30 GHz and 4 GB RAM is operating on 64-bit Microsoft Window 7. The random dispersion of sensor nodes and the gateways on the positions determined by the k-means algorithm in the sensing field of 1000 * 1000 square meters has been described in Figure 2.

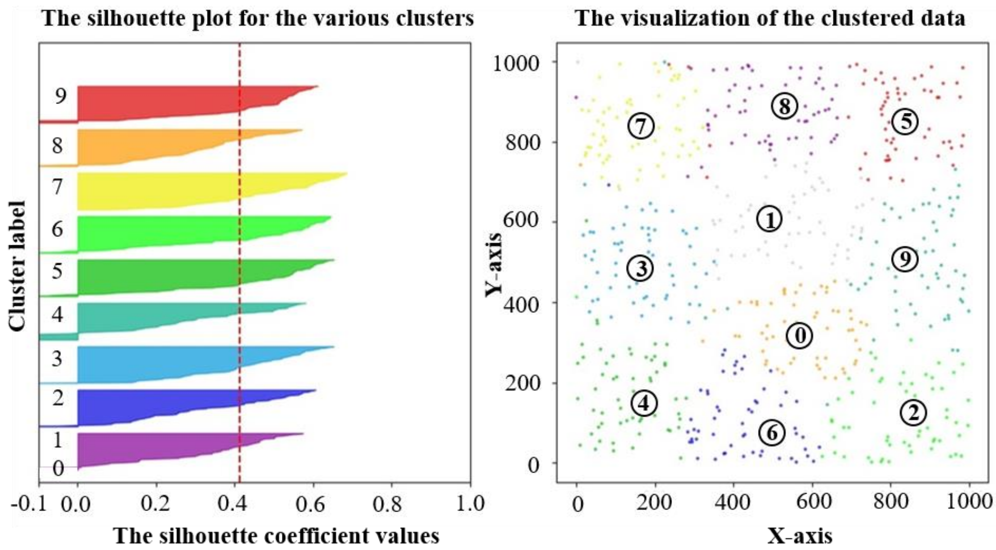

Figure 2 Silhouette analysis for K-Means clustering on 600 sensor nodes with 10 clusters

The number of sensor nodes varies between 100 and 600, while the number of gateways set to 10 in the simulation work. Each sensor is initialized with the initial energy of 5 joules. The sensor is said to be non-operational when its residual energy is less than the energy expected to forward the data packets. The highest span of the sensor nodes is equal to 0.5 times the maximum distance between two sensor nodes. The packet size of data to be transmitted is set to 10K bits. All the sensor nodes are viewed as the equivalent, and no inclination is given to any sensor node or gateway. The gateways are deployed at the position determined by the K-means algorithm. The gateways are fixed on their position throughout the lifetime of the network. After clustering, the sensor nodes belonging to their cluster are displayed in Figure 2.

5.1 Performance of LBCM

This section presents the outcomes attained by our simulation work. Firstly, we briefly discussed the LBCM algorithm. After that, our achievement of the proposed algorithm was presented by contrasting the outcomes, including the SDC, LBCA, CELBA, DELBA, and EELBCA. The three unique properties of the network have been measured, which depend on various metrics.

Table 2 Simulation results to illustrate the effect of sorting the gateways based on their average Euclidean distance in ascending and descending order

# Sensor nodes	Order of the Gateway list for clustering					
	Descending order		Ascending order		Deviation in % by ascending order	
	Avg. sensor life (rounds)	Average C.E. (joules)	Avg. sensor life (rounds)	Average C.E. (joules)	Average C.E. (joules)	Avg. sensor life (rounds)
100	742	0.03929	743	0.03983	+1.37	+0.13
200	638	0.02561	638	0.02755	+7.58	+0.00
300	606	0.02732	606	0.0285	+4.32	+0.00
400	573	0.02443	574	0.02536	+3.81	+0.17
500	551	0.03334	552	0.03516	+5.46	+0.18
600	525	0.03091	526	0.03267	+5.69	+0.19
350	**606**	**0.03015**	**607**	**0.03151**	**+4.70**	**+0.11**

In WSNs, the distance among the cluster head and nodes plays a crucial role in communication cost. The initial assumption is that if the sensor nodes have a closer cluster head, communication costs will be less. Therefore, the sensor nodes need to distribute among the cluster head sensibly. Through the LBCM algorithm, the near and sensible distribution of sensor nodes among the gateway achieved using the average Euclidean distance and sequential addition of sensor nodes. The Euclidean distance-based node assignment solves the purpose of reducing communication costs. Moreover, the sensor nodes assignment can be chosen from the descending order list because the descending order performs better concerning the ascending order list, as shown in Table 2. If the clustering performs according to the gateway list sorted in ascending order, it will consume an average of 4.70% more average communication energy than the descending one, while the deviation in terms of sensor life is negligible.

5.1.1 Standard deviation of load per gateway

The standard deviation of the load of each gateway is determined by varying the number of sensor nodes from 100 and 600. The number of gateways is fixed during the test, which are 10 in this case. The identical test was executed on LBCM, SDC, DELBA, CELBA, LBCA, and EELBCA.

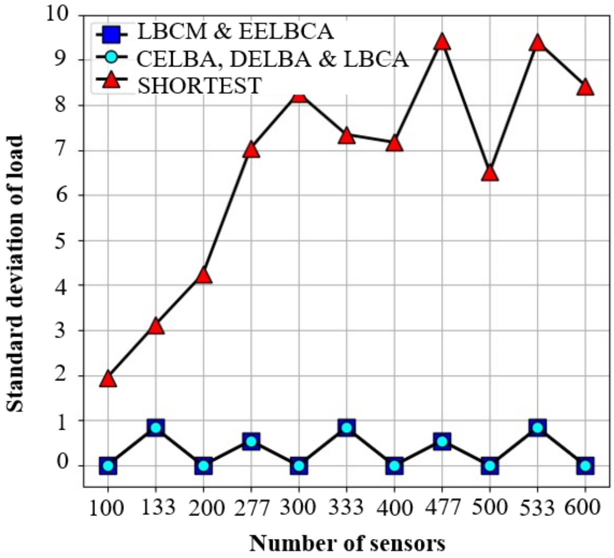

Figure 3 Standard deviation of load with increased number of sensor nodes

The outcomes displayed in Figure 3 demonstrated that the LBCM and other discussed algorithms are equal in performance, while the rising graph of SDC designates that variation in load is growing with an increase in the number of sensor nodes. The linear increase in load in line with each gateway designates that all the presented algorithms can deal with scalability. In contrast, for SDC, an intensely burdened gateway cannot deal with the network, resulting in an unusual packet drop rate.

5.1.2 Average communication energy per gateway

The average communication energy of sensor nodes for transmitting a packet containing sensed data of capacity 10 K bits to its gateway has been estimated by utilizing Equation 4. Likewise, the equivalent is also evaluated by increasing the number of sensor nodes from 100 to 600 in the network. The first gateway in which the clustering process starts plays an essential role in energy-efficient clustering. After calculating the average Euclidean distance of sensor nodes from each gateway, the gateway list is sorted in descending order. The simulation result illustrated in Table 2 shows that if the clustering is performed according to the gateway list sorted in descending order, the average communication energy consumption within the cluster is minimum compared with the gateway list sorted in ascending order.

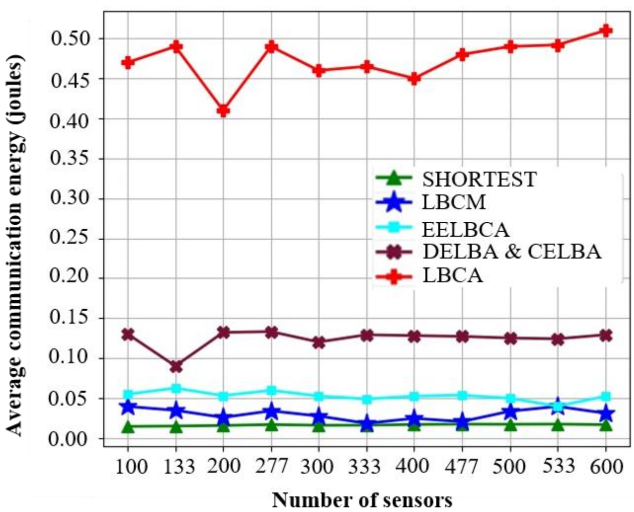

Figure 4 Average communication energy per gateway with increased number of sensor nodes

The experiments played out, and the outcome from Figure 4 reveals the efficiency of the LBCM algorithm. The LBCM is more energy-efficient amongst DELBA, CELBA, LBCA, and EELBCA but not as good as the SDC. The SDC at first expends less transmission energy; however, because of its unbalancing nature of the load, it places substantial weight on the gateway, due to which the overall performance of the gateway may diminish.

5.1.3 Average lifetime of sensor nodes

There is an assumption that there is no delay in data processing or transmission, and the data packets are not dropped during communication. The average lifetime of sensor nodes has been evaluated by increasing the number of sensor nodes from 100 to 600. The number of gateways is set to 10 during experiments. The average communication energy consumption of sensor nodes within the cluster is minimal in LBCM compared with other existing algorithms except SDC. Minimal communication energy consumption positively increases the average lifetime of the sensor nodes.

As it can be analyzed from the simulation result displayed in Figure 5, that the LBCM achieves a better average lifetime of sensor nodes amongst DELBA, CELBA, LBCA, and EELBCA. In the case of SDC, the sensor nodes utilize minimal communication energy. In correlation with SDC, LBCM works approximately equivalent to the shortest distance clustering in the same environment.

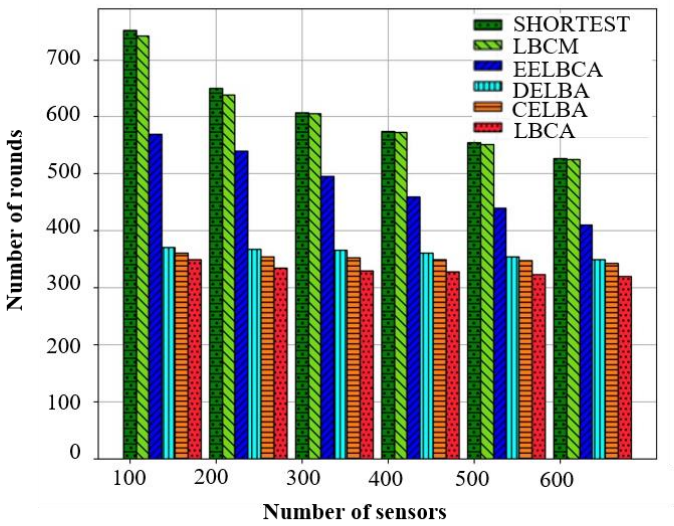

Figure 5 Average lifetime of sensor nodes

6. Conclusions

This paper has presented a new energy-efficient LBCM algorithm for multi-gateway WSNs to cluster the deployed sensor nodes around less energy-constrained gateways and balance the load of sensor nodes among these gateways. The role of the gateway is to act as a relay node to transmit data from sensor nodes to the base station. If the sensor nodes are not dispersed evenly around the gateways, the clusters created will be of unbalanced load, which will affect the lifetime and energy consumption of the network. LBCM balances the load of the gateways in terms of the number of allotted sensor nodes and the communication load of the gateways. The proposed algorithm's experimental results have been presented based on three performance metrics, i.e., gateway loads, energy consumption, and sensor nodes' lifetime. The suggested LBCM algorithm consistently balances the load among different clusters and gives better performance than similar algorithms concerning energy consumption and sensor nodes' lifetime.

7. References

[1] Akilidz IF, Su W, Sankarasubramaniam Y, Cayirci E. Wireless sensor networks: a survey. Comput Network. 2002;38:393-422.
[2] Guo J, Jafarkhani H. Movement-efficient sensor deployment in wireless sensor networks with limited communication range. IEEE Trans Wireless Comm. 2019;18(7):3469-84.
[3] Niculescu D, Nath B. Ad-hoc positioning system (APS). GLOBECOM'01. IEEE Global Telecommunications Conference; 2001 Nov 25-29; San Antonio, USA. New York: IEEE; 2001. p. 2926-31.
[4] Abbasi AA, Younis M. A survey on clustering algorithms for wireless sensor networks. Comput Comm. 2007;30:2826-41.
[5] Zhao M, Yang Y, Wang C. Mobile data gathering with load balanced clustering and dual data uploading in wireless sensor networks. IEEE Trans Mobile Comput. 2015;14(4):770-85.
[6] Ahmed I, Peng M, Wang W. A unified energy efficient cluster id based routing scheme for wireless sensor networks - a more realistic analysis. International Conference on Networking and Services (ICNS '07); 2007 Jun 19-25; Athens, Greece. New York: IEEE; 2007.
[7] Younis O, Fahmy S. HEED: A hybrid, energy-efficient, distributed clustering approach for ad-hoc sensor networks. IEEE Trans Mobile Comput. 2004;3:366-79.
[8] Yick J, Mukherjee B, Ghosal D. Wireless sensor network survey. Comput Network. 2008;52(12):2292-330.
[9] Liu M, Cao J, Chen G, Wang X. An energy-aware routing protocol in wireless sensor networks. Sensor. 2009;9(1):445-62.
[10] Heinzelman WR, Chandrakasan A, Balakrishnan H. Energy-efficient communication protocols for wireless microsensor networks. Proceedings of the 33rd annual Hawaii international conference on system sciences; 2000 Jan 7; Maui, USA. New York: IEEE; 2000. p. 3005-14.

[11] Low CP, Fang C, Jim MN, Ang YH. Efficient load-balanced clustering algorithms for wireless sensor networks. Comput Comm. 2008;31:750-9.

[12] Gupta G, Younis M. Load-balanced clustering of wireless sensor networks. IEEE International Conference on Communications; 2003 May 11-15; Anchorage, USA. New York: IEEE; 2003. p. 1848-52.

[13] Kuila P, Jana PK. Energy efficient and load balanced clustering algorithms for wireless sensor networks. Proc Tech. 2012;6: 771-7.

[14] Tarachand A, Kumar V, Raj A, Kumar A, Jana PK. An energy efficient load balancing algorithm for cluster-based wireless sensor networks. 2012 Annual IEEE India Conference (INDICON); 2012 Dec 7-9; Kochi, India. New York: IEEE; 2012. p. 1250-4.

[15] Kuila P, Jana PK. Heap and parameter-based load balanced clustering algorithms for wireless sensor networks. Int J Comm Network Distr Syst. 2015;14(4):413-32.

[16] Heinzelman WB, Chandrakasan AP, Balakrishnan H. An application-specific protocol architecture for wireless microsensor networks. IEEE Trans Wireless Comm. 2002;4:660-70.

[17] Manjeshwar A, Agrawal D. APTEEN: A hybrid protocol for efficient routing and comprehensive information retrieval in wireless. Proceedings 16th International Parallel and Distributed Processing Symposium; 2002 Apr 15-19; Ft. Lauderdale, USA. New York: IEEE; 2002. p. 195-202.

[18] Manjeshwar A, Agrawal D. TEEN: A routing protocol for enhanced efficiency in wireless sensor networks. Proceedings of 15th international parallel and distributed processing symposium; 2001 Apr 23-27; San Francisco, USA. New York: IEEE; 2001. p. 2009-15.

[19] Park GY, Kim H, Jeong HW, Youn H. A novel cluster head selection method based on k-means algorithm for energy efficient wireless sensor network. 27th International conference on advanced information networking and applications workshops; 2013 Mar 25-28; Barcelona, Spain. New York: IEEE; 2013. p. 910-5.

[20] Li X, Yu L, Hang L, Tang X. The parallel implementation and application of an improved k-means algorithm. J Univ Electron Sci Tech China. 2017;46(1):61-8.

[21] Han L, Wang Q, Jiang Z, Hao Z. Improved K-means initial clustering center selection algorithm. Comput Eng Appl J. 2010;46(17):150-2.

[22] Likas A, Vlassis N, Verbeek J. The global k-means clustering algorithm. Pattern Recogn. 2003;36(2):451-61.