ENGINEERING RESEARCH JOURNAL (ERJ)

Vol. 1, No. 43 Jan. 2020, pp. 87-91

Journal Homepage: http://erj.bu.edu.eg

Energy Conservation using K-Means algorithm for IoT Networks

Ahmed Gamal SolimanSolimanDeabes [1], Dr. Michael Nasief Mikheal [2] , Assoc. Prof. Dr. Mostafa Fouda [3] , Prof. Dr. HalaAbd El-kader Mansour[4]

[1]Teaching Assistant in Modern University for Technology and Information.

[2] Doctor of Communication - faculty of Engineering – Shoubra - Benha University.

[3] Assoc. Professor of Communication - faculty of Engineering – Shoubra - Benha University.

[4]Professor of Communication - faculty of Engineering – Shoubra - Benha University

Abstract. : Internet of things (IoT) is an anywhere anytime networking technique to connect the smart devices irrespective of their location. loT is a new paradigm that is highly gaining ground in today's world. Wireless Sensor Network (WSN) is a prime part of IoT. The main goal of the clustering scheme is to collect and aggregate data packets in an efficient method so as to reduce power consumption, data accuracy and to increase network lifetime. In this paper,a cluster-based energy-efficient router placement scheme for IoT was implemented, where the K-Means algorithm is used to select the cluster header. The performance of the implemented scheme was evaluated in terms of energy consumption, throughput, end-to-end delay, and packet loss. The simulation results using the python simulator and Nodemcu 'ESP8266' show that the energy consumption of the implemented scheme improving the lifetime of the network by 35.7% over that of the low-energy adaptive clustering hierarchy (LEACH). The packet loss of the network is improved by 32% over that of LEACH. The throughput of the network is improved by 27.5% over that of LEACH.

Keywords: Internet of things, wireless sensor networks, cluster head selection, K-means algorithm.

1. INTRODUCTION

IoT comes into the networked linkage of daily devices which increases the pervasiveness of the Internet by mixing any smart device. It allows the linkage of people and devices everywhere, anytime with anybody using any track. In IoT, a massive number of objects can be found through different types of sensors and actuators which are linked to the Internet by heterogeneous access networks such as WSN, wireless mesh network (WMN), etc.[1].In IoT, heterogeneous devices can exchange information about their position, identities, and condition to people as well as other devices that are linked to the internet. These devices would be used in the design of new engineering clarifications to community trouble such as smart greenhouses, power metering, agricultural drones, livestock monitoring, etc.

WSN is a primary fraction of IoT which assists in aggregating information from the surroundings. The sensor nodes (SNs) always employ energy constrained little batteries for their energy supply [2]. Therefore, battery consumption is a critical worry in extending the lifetime of a network operation[2]. WSN consists of several low-cost low-energy SNs that can perform sensing, easy calculation, and send of the sensed data to the cluster head (CH).

Clustering is one of the great processes for extending the network lifetime in IoT. It aggregates SNs into clusters and select CHs for all the clusters. CHs collect the data from nodes of each cluster and send the aggregated data to the gateway .Clustered systems overcome the communication expenses and offer an effective resource allocation, thereby lessening the total power consumption and interference between SNs [2].To implement the suggested plan using the K-Means algorithm in building CHs to reduce network power consumption and increases network lifetime.

The results show that the implemented scheme provides better than performance than low energy adaptive clustering hierarchy (LEACH) in terms of power consumption and network lifetime.

The rest of this paper is organized as follows. Section II mention to the different data aggregation techniques proposed for IoT. The details of the implemented scheme are discussed in Section III. Section IV demonstrates the performance analyses of our implemented scheme and presents comparative results. Section V illustrates the final conclusion.

2. BACKGROUND

Several researches focus on grouping the sensor devices into clusters to extend the lifetime of IoT[3]. Chandrakasan et al. [4]suggested a clustering-based protocol called LEACH (Low Energy Adaptive Clustering Hierarchy). LEACH consists of two phases: the set-up phase and the steady phase. The set-up phase in which the sensor nodes are organized by groups and select cluster heads. This is followed by a steady phase the CHs will send the aggregated data to the gateway .The duration of the steady phase is larger than that of the set-up phase. However, LEACH affords a lot of calculations during the selection of CHs ,which would further shorten the network lifetime.

LEACH-C [5] proposes a scheme with more regularly spread clusters than LEACH [6]. In both LEACH and LEACH-C, the CHs immediately contact with the gateway which attributes to more energy consumption and decreases lifetime battery.

A two-tier cluster-based data aggregation protocol TTCDA [7]uses similar method where CHs transmit the aggregated data to the gateway.

Tree-based architecture is suitable to achieve energy efficiency [8]. Tree based energy efficient protocol for sensor information (TREEPSI) [9]constructs a tree considering sink as the root node. All the leaf nodes forward the data to their parents and then are routed towards the sink. Medlej et al. in [10], [11] and [12]proposed tree-based aggregation protocols that work in two stage. The first stage is at the local level aggregation and the second stage is at the aggregators level. At each period, each node forwards its aggregated data set to their proper aggregator which posteriorly aggregates all data sets coming from different sensor nodes and forwards them to the sink.

3. PROPOSED PROTOCOL

In the following section, the implementation of the cluster is described based on power efficiency in IoT.

Several advantages are provided by the data aggregation using clustering techniques such as removing the data redundancy, decreasing the communication cost, improving the lifetime of networks.

A sensor that is away from the CH consumes a huge amount of power therefor exhausts its energy within a little time. As seen in Figure 1, if the gateway randomly selects sensors as CHs, the power consumed by the sensors to access the CHs will not be optimal.

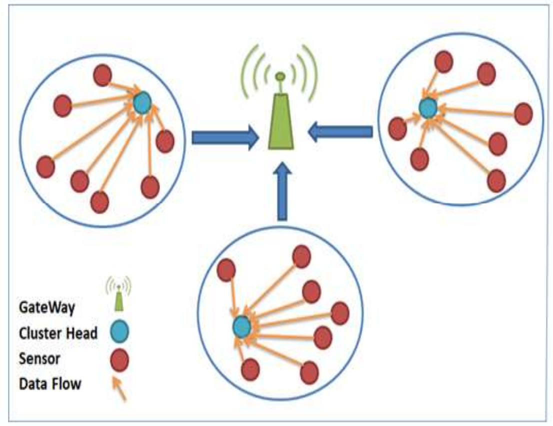


Fig 1 - Location of the CH without using the K-Means algorithm.

However, using the K-Means algorithm, when the CH is located at the centroid of each cluster, the energy consumed by the sensors that are away from the CH will be decreased, which will increase the lifetimes of the sensors. Figure 2 shows the CHs placed at the centroids of the clusters.

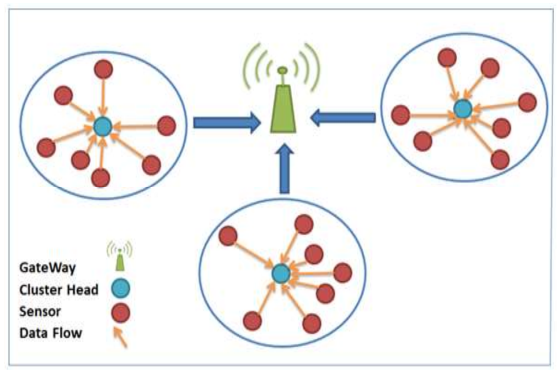


Fig 2 - Location of the CH in the centroid using the K-Means algorithm.

The objective of the proposed clustering technique, the implemented K-means algorithm in the cluster calculates the distance between each sensor and to selects the CH which is the nearest to the centroid for the sensors.

The first step in location limitation is calculating the distance between the sensors whose locations are unknown and the nodes whose locations are known. In order to accurate location position, distance placement should be improved. Distance calculations can be impacted by interference, receiving device, transmitting device and more. So as to determine the better method of distance limitation, tests were run to compare the received signal strength indication (RSSI) between node and sensor and calculate the distance between them using this equation(1)[2]

$$d = 10^{\frac{\frac{t_x}{10} - r_x - 10\log\frac{4\pi}{c/f}}{20\rho}} \quad (1)$$

Here, d is the distance between node and sensor, t_x is the transmitter signal power, r_x is receiver signal power, c is the speed of light, f is the transmitter frequency and ρ is the path loss.

The next step is to calculate the location indoor of the sensor. After calculating the distance between node and sensor, the distance can be applied to limit location indoor. And then selects CH near the centroid.

The four nodes will be installed in the corners, with their fixed x, ylocation in the SQL (database). Then a sensor broadcast signal, the four nodes pick up this signal. Then limitation location these sensors.

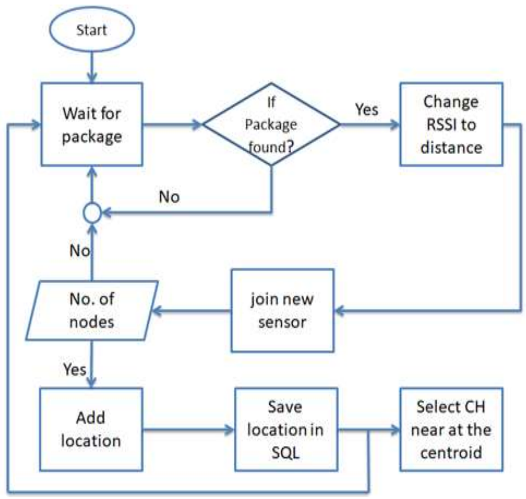
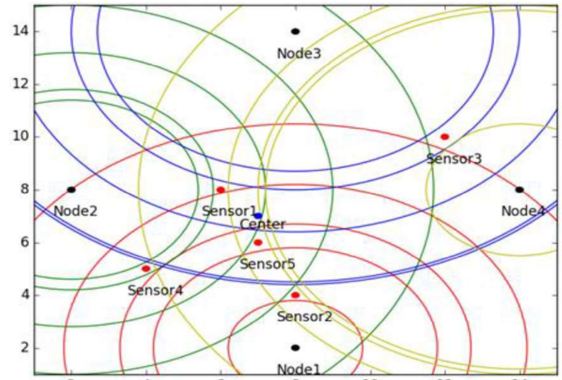


Fig 3 - Server operation Flowchart.

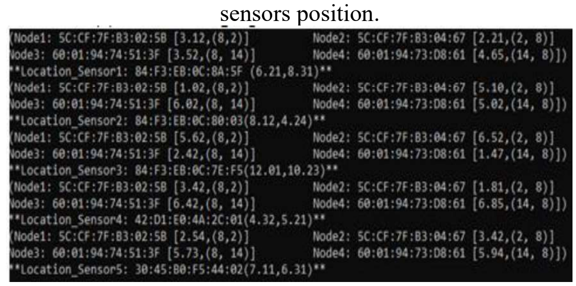
Figure 3 gives a detailed flow of the implementation scheme. Each node from four nodes searching for any broadcast signals from sensors. When a signal is found, the node calculates RSSI between nodes and sensor, and then node sends a UDP packet to a server using python which contains the MAC address of the nodes, sensor, and RSSI.

The server system in python will have a UDP socket that receives incoming packets from the nodes. Once a packet is received, the server looks in SQL for the MAC addresses in the packet.

The database in SQL includes locations fixed x, y for nodes, found based on each node's MAC address in the database. Using this information, a design includes all the nodes locations and RSSI for every sensor and remove the duplicate node information for every sensor. Then add information to the new node.



(a)Test result graphical four nodes with five sensors position.



```
(Node1: 5C:CF:7F:B3:02:5B [3.12,(8,2)]        Node2: 5C:CF:7F:B3:04:67 [2.21,(2, 8)]
Node3: 60:01:94:74:51:3F [3.52,(8, 14)]       Node4: 60:01:94:73:D8:61 [4.65,(14, 8)])
**Location_Sensor1: 84:F3:EB:0C:8A:5F (6.21,8.31)**
(Node1: 5C:CF:7F:B3:02:5B [1.02,(8,2)]        Node2: 5C:CF:7F:B3:04:67 [5.10,(2, 8)]
Node3: 60:01:94:74:51:3F [6.02,(8, 14)]       Node4: 60:01:94:73:D8:61 [5.02,(14, 8)])
**Location_Sensor2: 84:F3:EB:0C:80:03(8.12,4.24)**
(Node1: 5C:CF:7F:B3:02:5B [5.62,(8,2)]        Node2: 5C:CF:7F:B3:04:67 [6.52,(2, 8)]
Node3: 60:01:94:74:51:3F [2.42,(8, 14)]       Node4: 60:01:94:73:D8:61 [1.47,(14, 8)])
**Location_Sensor3: 84:F3:EB:0C:7E:F5(12.01,10.23)**
(Node1: 5C:CF:7F:B3:02:5B [3.42,(8,2)]        Node2: 5C:CF:7F:B3:04:67 [1.81,(2, 8)]
Node3: 60:01:94:74:51:3F [6.42,(8, 14)]       Node4: 60:01:94:73:D8:61 [6.85,(14, 8)])
**Location_Sensor4: 42:D1:E0:4A:2C:01(4.32,5.21)**
(Node1: 5C:CF:7F:B3:02:5B [2.54,(8,2)]        Node2: 5C:CF:7F:B3:04:67 [3.42,(2, 8)]
Node3: 60:01:94:74:51:3F [5.73,(8, 14)]       Node4: 60:01:94:73:D8:61 [5.94,(14, 8)])
**Location_Sensor5: 30:45:B0:F5:44:02(7.11,6.31)**
```

(b)Test result text four nodes with five sensors position.

Fig 4 - Test result output of the system.

Figure 4 shows the results of the system as text and graphical representation. The graphical output shows (a) four nodes as black dots. Surrounding the nodes are different color circles that represent the calculated distance of each node. The system prediction the location of the sensor is the red dot. The text output lists (b) the arrays of the nodes and all sensor objects which are detected. Each node that is detected (i.e. the sensor) has its MAC address, distance, and x, y fixed location finally the server prints the calculated distance.

In figure 5, the locations of the sensors are saved in SQL and then select CH near the centroid.

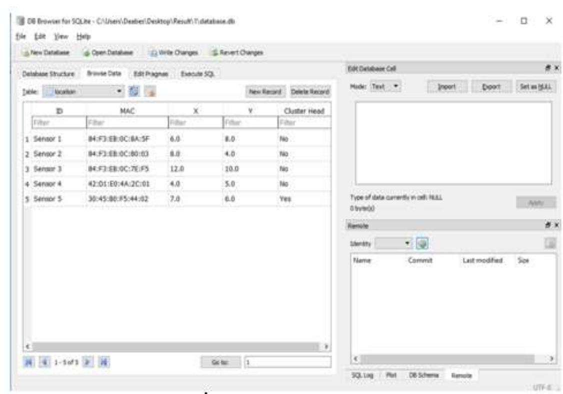


Fig 5 - Database Storage.

Evaluation metrics

The calculation of the total remaining lifetime of the sensors in a cluster using this equation[2]

$$R_{Lcluster} = \frac{1}{4\pi S}\sum_{i=1}^{n}\frac{E_a - E_{th}}{r_i^{\,2}} \quad (2)$$

Here, $R_{Lcluster}$ is the total remaining lifetime of the sensors, E_a is the actual energy of the sensor,

E_{th} is an energy threshold, S is strength, and $14\pi r_i^2 S$ is the energy transmitted in the cluster.
Then calculation of the remaining energy of the sensor using this equation[2]

$$E_d = E_a - \int_0^f P_c dt \quad (3)$$

Here, E_d is the remaining energy if it has been sending for 'f' of time at a distance of 'd', P_c power consumption of the sensor, $P_c = (4\pi S)^2$
In the cluster, the calculation of total energy consumed by all the sensors and that consumed to sending to the gateway by using this equation[2]

$$E_{tcluster} = \sum_{i=1}^{n} E_a - 4\pi Sf \sum_{i=1}^{n} {E_d}^2 + E_{CTW} \quad (4)$$

Here, E_{CTW} is the energy consumed to cluster head send to the gateway in the server.

4. PERFORMANCE EVALUATION

This section discusses the evaluation of the performance of the implemented scheme in terms of energy consumption, as well as the end-to-end delay, packet loss, and throughput. The evaluation was performed using the python tool in a 64-bit Windows 10 operating system. The implementation parameters are listed in Table 1.

Table1. Implementation parameters.

Parameter	Size
Size of network	$50m \times 50m$
Node type	Nodemcu "ESP 8266"
Examined parameter	Real-time and best-effort applications
Simulator	Python and Arduino
Simulation time	90 min
Number of Sensors	16
Number of iterations	35

Energy consumption

As indicated in Figure 6, the energy consumption of the implemented scheme improving the lifetime of the network by 35.7% over that of LEACH. The higher energy consumption of the LEACH may have resulted from the transmitter and receiver between sensors during the select CH. When there is a little number of sensors in the CH, the energy consumption of the sensors is high. As seen in Figure 6, the consumed energy decreases when the number of sensors increases.

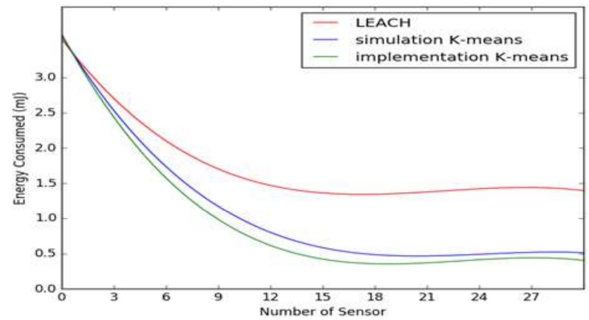


Fig 6 - Energy consumption.

End-to-end delay

As indicated in Figure 7, The End-to-end delay of the network is improved by 30.4% over that of LEACH. When the CH the transmitter and receiver with many sensors, it aggregates data from all the sensors in that cluster and sends the aggregated data to the gateway. At this period, packet losses occur and delay. Initially, when the number of sensors is small, the end-to-end delay is high, and as the number of sensors increases the delay decreases. When the number of sensors increases, energy consumption decreases continuously it reaches the maximum and throughput increases. After reaching the highest, the throughput begins to decrease.

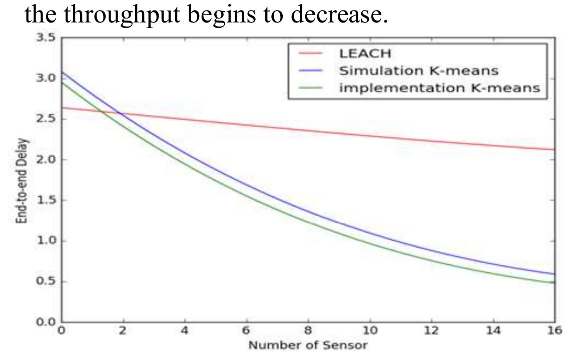


Fig 7 - Total delay.

Packet loss

As indicated in Figure 8, the packet loss of the network is improved by 32% over that of LEACH. LEACH had extra packet is high, and as the number of sensors increases the delay decreases. When the number of sensors increases, energy consumption decreases until it reaches the maximum and throughput increases.

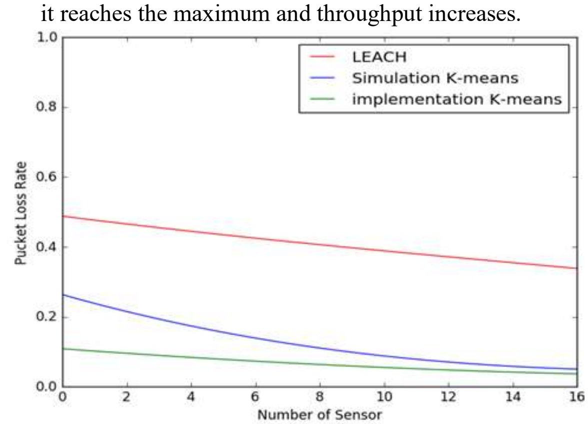


Fig 8 - Traffic transmitter and received.

Throughput

As indicated in figure 9, the throughput of the network is improved by 27.5% over that of LEACH. The results show that increasing the number of sensors increases throughput. After reaching the highest, the throughput begins to decrease. In the case of LEACH, the throughput gradually increases with an increase in the number of sensors.

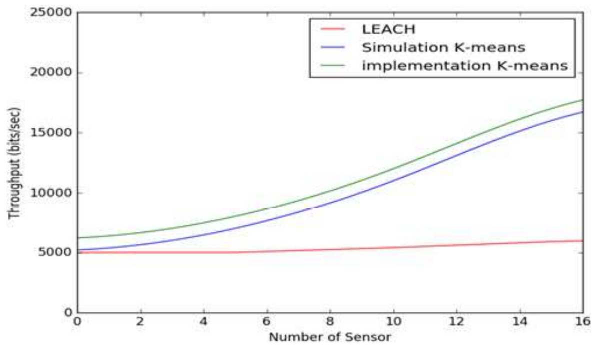


Fig 9 - Throughput versus the number of Sensors.

5. Conclusion

Io T is the future of technology that will decide how we control and react with our daily devices and make them more efficiently. The main problem with IoT is the improper use of energy. In this paper, a cluster-based energy-efficient router placement scheme for IoT was implemented, where the K-Means clustering algorithm to place the routers in the process of selects CHs. The implementation scheme achieves higher energy saving, network lifetime and throughput. The performance of the implemented scheme was evaluated in relation to the packet loss, throughput, end-to-end delay, energy consumption. In the simulation results, observe that the scheme accomplished a better execution than LEACH.The energy consumption of the implemented scheme improving the lifetime of the network by 35.7% over that of the LEACH. The packet loss of the network is improved by 32% over that of LEACH. The throughput of the network is improved by 27.5% over that of LEACH.

REFERENCES

[1] L. Atzori, A. Iera, and G. Morabito, "The Internet of Things: A survey," *Comput. Networks*, vol. 54, no. 15, pp. 2787–2805, 2010.

[2] A. F. Jemal, R. H. Hussen, D. Y. Kim, Z. Li, T. Pei, and Y. J. Choi, "Energy-efficient selection of cluster headers in wireless sensor networks," *Int. J. Distrib. Sens. Networks*, vol. 14, no. 3, 2018.

[3] A. K. Idrees, W. L. Al-Yaseen, M. A. Taam, and O. Zahwe, "Distributed Data Aggregation based Modified K-means technique for energy conservation in periodic wireless sensor networks," *2018 IEEE Middle East North Africa Commun. Conf. MENACOMM 2018*, pp. 1–6, 2018.

[4] 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.

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

[6] V. Geetha, P. V. Kallapur, and S. Tellajeera, "Clustering in Wireless Sensor Networks: Performance Comparison of LEACH & LEACH-C Protocols Using NS2," *Procedia Technol.*, vol. 4, pp. 163–170, 2012.

[7] D. Mantri, N. R. Prasad, R. Prasad, and S. Ohmori, "Two Tier Cluster based Data Aggregation (TTCDA) in Wireless Sensor Network," *2012 IEEE Int. Conf. Adv. Networks Telecommunciations Syst. ANTS 2012*, pp. 117–122, 2012.

[8] S. Kaur and R. C. Gangwar, "A Study of Tree Based Data Aggregation Techniques for WSNs," *Int. J. Database Theory Appl.*, vol. 9, no. 1, pp. 109–118, 2016.

[9] S. S. Satapathy and N. Sarma, "TREEPSI: tree based energy efficient protocol for sensor information," pp. 4 pp. – 4, 2006.

[10] J. Bahi, A. Makhoul, and M. Medlej, "An Optimized In-Network Aggregation Scheme for Data Collection in Periodic An Optimized In-Network Aggregation Scheme for Data Collection in Periodic Sensor Networks HAL Id : hal-00940000," no. July, 2012.

[11] J. M. Bahi, A. Makhoul, and M. Medlej, "A two tiers data aggregation scheme for periodic sensor networks," *Ad-Hoc Sens. Wirel. Networks*, vol. 21, no. 1–2, pp. 77–100, 2014.

[12] J. M. Bahi, A. Makhoul, and M. Medlej, "Data aggregation for periodic sensor networks using sets similarity functions," in *IWCMC 2011 - 7th International Wireless Communications and Mobile Computing Conference*, 2011, pp. 559–564.