

## Determination of the K-optimal number of chains-based routing protocol formed by the K-Means algorithm for the WSN

Samah Alnajdi, Fuad Bajaber

Department of Information Technology, Faculty of Computing and Information Technology,  
King Abdulaziz University, Kingdom of Saudi Arabia

---

### Article Info

#### Article history:

Received Dec 12, 2019

Revised Feb 15, 2020

Accepted Feb 29, 2020

---

#### Keywords:

Chain-based clustering  
Energy consumption  
hierarchical routing protocols  
K-means algorithm  
Network lifetime  
Optimal k  
Overhead  
WSNs

---

### ABSTRACT

Wireless sensor networks comprise of a large number of lightweight and relatively low-cost computational nodes which their main task is to sense the surrounding environment and collect the information to send it wirelessly to a central point to take the appropriate actions. Although these networks had been used in various applications, achieving this task is challenging due to the many constraints of sensor nodes including their limited processing power, communication bandwidth, and power supply. Therefore, an energy efficient routing protocols had to be developed specifically for sensor networks to insure longer lifetime and reasonable performance of the network. In this work, we propose an energy efficient hierarchical routing protocol using chain-based clustering. By simulation on MATLAB, the proposed protocol proved to enhance the performance as it prolongs the lifetime of the network and decreases the energy consumption, the transmission delay, and the overhead compared to other existing protocols as it depends on some advanced methods including dynamic selection of number of chains method, k-means clustering method, advanced greedy chain construction method, and multi-factor based leader selection method.

Copyright © 2020 Institute of Advanced Engineering and Science.  
All rights reserved.

---

### Corresponding Author:

Samah Abdullah Alnajdi,  
Department of Information Technology,  
Faculty of Computing and Information Technology,  
King Abdulaziz University,  
Email: alnajdisamah@gmail.com

---

## 1. INTRODUCTION

A wireless sensor network (WSN) is a specialized network that consists of a number of lightweight nodes known as sensors. The sensors are randomly deployed in the affected area to monitor the environment and gather the required information, such as movement, lights, or temperature [1]. Moreover, they must work cooperatively and communicate to send the gathered information to a more powerful node to be used for useful purposes. Although the WSNs had been widely used in various applications due to the sensing ability of the sensors, the tiny size that allows the sensor to be deployed in hard-to-reach places, the low costs and the self-organizing behavior of the sensor nodes, there are many issues in WSNs due to the limited functions of sensors. The sensors normally have limited processing capacity, limited communication bandwidth, limited memory, and, most importantly, a limited power supply since they have un-rechargeable batteries.

Considering the limitations of sensor nodes that are discussed above, applying an efficient routing mechanism is one of the main critical issues in WSNs. As routing in WSNs differs from conventional networks, many routing algorithms and protocols proposed specifically for WSNs belong to various categories. For instance, these include location-based protocols, mobility-based protocols, heterogeneity based protocols, quality of service (QoS)-based protocols, hierarchical protocols, multipath-based protocols, and data-centric protocols [2]. Hierarchical routing is the most used routing mechanism in WSNs as it mainly

aims to establish efficient energy consumption using the sensor nodes and the overall long lifetime of the network. Clustering is performed in hierarchical routing where the network is broken into layers. The first layer presents all the nodes since they too are partitioned into clusters based on some clustering algorithms. In each cluster, one node is elected to manage the cluster and is known as the cluster head. All the cluster heads together represent the second layer of the network that is responsible for sending over farther distances, such as to the BS. However, hierarchical routing protocols themselves have different types of clustering: cluster-based, chain-based, or tree-based clustering.

In cluster-based routing protocols, the nodes in each cluster are meant to only communicate and forward the data directly to their cluster head. For tree-based routing protocols, the routing path is built in a parent-child manner, and each node always forwards to its parent node. On the other hand, chain-based routing protocols connect the nodes together to form a chain, and the data are forwarded along the chain to the BS. In this paper, we review most of the existing chain-based protocols and propose our version of the improved chain-based routing by including some advanced algorithms during the different phases of the routing path development process, including finding the optimum number of clusters in the case of chain-based clustering. The paper organized as follows. The literature review in Section 2 is followed by our network and the used communication radio model in Section 3. The proposed protocol is given in Section 4, and its performance evaluation in comparison with other protocols in addition to statistical analysis is provided in Section 5. Finally, the conclusions are given in Section 6.

## 2. LITERATURE REVIEW

Connecting the sensor nodes in a chain-like topology would highly benefit the energy consumption compared with other clustering approaches. Thus, more studies have been carried out to improve the routing process using chain-based clustering. We review most of these routing protocols by focusing on their chain construction algorithms and deployment area division methods. Table 1 summarizes all the reviewed protocols.

Table 1. Chain-based hierarchical routing protocols

Routing protocol	Network type	Number of chains	Number of chains	Chain construction method	Leader selection
PEGASIS [3]	homogeneous	single	Static	Greedy algorithm	In turn, in round-robin manner
EECB [4]	homogeneous	single	Static	Greedy algorithm with distance threshold	Considers both the residual energy of the node and the distance to BS
IIEPB [5]	homogeneous	single	Static	Advanced greedy algorithm	Uses weighting method that considers both the residual energy of nodes and the distance from node to BS.
SCBC [6]	heterogeneous	Multiple	Static	Greedy algorithm but SCBC starts with the nearest and the Leader node with two different chains to form one chain.	The residual energy and cost function are considered. Secondary leader nodes are selected to send to the BS
Ghosh et al. [8]	homogeneous	single	Static	Ant Colony Optimization [7]	Considers both the remaining energy of the node and the distance to BS
Gupta and Saraswat [9]	homogeneous	single	Static	Greedy algorithm with the sensor nodes allowed to opt visited nodes again if they are the nearest one from those sensor	Considers the degree of nodes in addition to the residual energy and distance from the BS
EPEGASIS [10]	homogeneous	Multiple	Static	Greedy algorithm	In turn, in round-robin manner
CHIRON[11]	homogeneous	multiple	Static	Greedy algorithm	Considers the residual energy only.
Shekh et al. in [13]	homogeneous	Multiple	Static	Greedy algorithm	Considers the nodes density in addition to the residual energy and distance from the BS
Hadjila et al. [14]	homogeneous	multiple	Static	Nodes in each group forms a single chain and sorted according to their ordinates.	First node in the chain, i.e. closest to the BS is the leader node.
Jawad and Ali [15]	homogeneous	multiple	Static	Greedy algorithm	Considers both the residual energy of the node and the distance to BS
Ruan et al. [16]	homogeneous	multiple	Static	Ant Colony Optimization	Uses the neural network to select the leader nodes.

Routing protocol	Network type	Number of chains	Number of chains	Chain construction method	Leader selection
Patel and Munjani [17]	homogeneous	multiple	Static	Greedy algorithm	In turn In round-robin manner
Bhatti and Raina [18]	Homogeneous	Single	Static	Modified Greedy algorithm with fuzzy system and cuckoo search algorithm	In turn In round-robin manner
PEGASIS-INL [19]	Homogeneous	Single	Static	Greedy algorithm	Based on the measured RSSI value
PEG-BBO [20]	Homogeneous	Single	Static	Biogeography-Based Optimization	x
IECBSN [21]	Homogeneous	multiple	Static	Forms multiple chains P, where P equals N/M as N the initial number of nodes and M the chains length. Starting from the closest node to the BS.	Considers both the residual energy of the node and the distance to BS
Hop PEGASIS [22]	Homogeneous	multiple	Static	Forms five clusters starting from the closest to the BS.	x

**2.1. Chain construction**

The nodes in chain-based clustering are supposed to involve multi-hop communication where the data packets are transmitted over several nodes in order to reach the final destination. However, the distance between each node and their neighbor node in the chain plays a major role in energy consumption (a larger distance means more energy consumption for transmitting). The first and most-used algorithm in chain-based routing protocols for connecting the nodes together is the greedy algorithm. In the greedy approach, local optimization is performed in which each node gets connected to the nearest available node despite the final result. As a result, a chain of nodes is constructed, but as the process progresses, the communication distance gets longer, as shown in Figure 1(a). The first use of this algorithm in hierarchical routing was in the PEGASIS protocol proposed by Lindsey and Raghavendra [3]. Other scholars suggested some modifications to solve the long link problem (LL) caused by the greedy algorithm where the data packet has to be transmitted through halve of the nodes in the chain in best case scenario to reach the leader node, and through all the nodes in worst case scenario if the leader node is located at the end of the chain. In EECB protocol [4], applying a threshold distance was proposed that if the distance between a node and its "ready to be" neighbor node exceeds a threshold value, the two nodes will not be allowed to connect. Instead, the latter node will search the chain of already connected nodes and connect to the nearest one of them.

A more advanced modification of the greedy algorithm was presented in the IEEPB protocol [5]. In the IEEPB, all nodes get the chance to search the already formed chain of nodes to connect to the nearest one of them. Therefore, the communication distance was further reduced in comparison to the PEGASIS and EECB algorithms. Figure 1(b) shows a chain constructed based on the Advanced Greedy Algorithm proposed in the IEEPB. Furthermore, the SCBC protocol [6] expanded the research list for the nodes by starting the construction of two chains at the same time. Therefore, as the process progresses, the nodes will have two already formed chains to search for their nearest node. Another known algorithm used for chain construction is Ant Colony Optimization (ACO) [7]. The algorithm guarantees the optimum communication path between the nodes and was applied in [8] by Ghosh et al. Nevertheless, connecting all the nodes into a single chain is not the most sufficient way of routing. The length of the routing path in terms of the number of hops/nodes affects the transmission delay since each node in the path has to wait to receive the data from its previous neighbor, aggregates the received data with its own sensed data, and then transmits it to the next neighboring node. Moreover, the leader node, which is the one sending to the BS, could be at any place on the chain and occasionally farther from the BS than other nodes. Therefore, redundant data transmission occurs when the closer nodes have to send their data in the direction of the leader node to be sent again by the leader in the direction of the BS. Gupta and Saraswat in [9] suggested that to order the nodes from the farthest to closest from the BS during the chain construction phase, the nodes are only allowed to connect with the nodes that are closer to the BS so the data transmission is more often in the BS direction.



Figure 1. Chain of nodes constructed based on the greedy algorithm; (b) A chain of nodes constructed based on the advanced greedy algorithm

## 2.2. Network partition

In addition to the transmission delay and the redundant data transmission problems, constructing a single chain including all the nodes causes high energy consumption, especially in high-density node situations. The probability of having more connections for the nodes means higher energy consumed for receiving data from the multiple child nodes. Therefore, some protocols are focused on finding the optimum methods to divide the nodes into multiple chains. One of the earliest efforts was the EPEGASIS protocol [10] in which dividing the deployment area into levels using a concentric clustering scheme was suggested, as presented in Figure 2(a). Per each level, a separate chain is constructed based on the greedy algorithm. The division method is further enhanced by the CHIRON protocol [11] to slightly control the lengths of the chains by applying the BeamStar technique [12] and dividing each level in half. Moreover, Shekh et al. in [13] wanted to control the lengths of chains even more in the farthest groups by dividing each level into a fixed sized group. The result of the partitioning method performed by Shekh et al. is shown in Figure 2(b).

In addition, other protocols performed vertical partitioning instead. For instance, the SCBC protocol divided the area into sectors with the BS as the head of division, as shown in Figure 2(c). Conversely, Hadjila et al.'s protocol [14] divided the area in a parallel manner. However, all the previous discussed works performed the partitioning in terms of the area space, regardless of the actual locations of the deployed sensor nodes. For random deployment, this could produce unnecessary energy consumption since the nodes may be forced to be assigned to groups with farther nodes to connect with instead of closer available neighbors. Therefore, partitioning the nodes based on their locations would save significant energy, and one of the known applied methods in other hierarchical protocols is k-means clustering. Jawad and Ali applied the method for chain-based routing in [15].

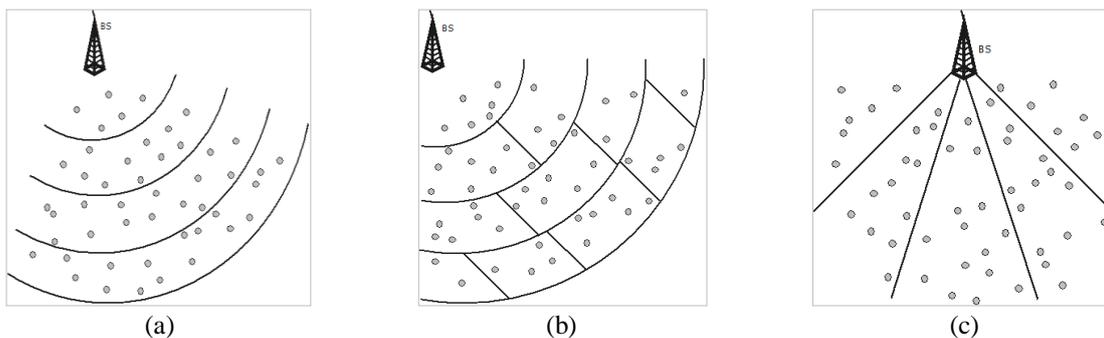


Figure 2. (a) Concentric clustering scheme; (b) Network partitioning by Shekh et al.; (c) Sectors scheme in SCBC

## 2.3. Leader nodes selection

In hierarchical routing, the node that is given the leader role will be responsible for receiving all the collected data from the rest of the nodes and forwarding it to the BS. Although the sizes of the received data packets are usually fixed due to data fusion and aggregation execution on every node, the leader node would still expend more energy than the other nodes due to the often large distance to the BS. Therefore, the selection of the leader node has to be a careful process in order to not over-consume the node's energy. Some frequently used factors for leader selection are the residual energy of the candidate node and its distance from the BS. In addition, some methods consider the number of connections or child nodes.

## 3. NETWORK AND RADIO MODEL

### 3.1. Network model

In our work, we assume the following.

- Sensor nodes are stationary and randomly distributed in a square field.
- The BS is stationary as well and deployed outside of the sensing field on one of the symmetric axes.
- The network is homogeneous since all sensor nodes have the same resources capabilities.
- The BS is not energy constrained.
- Each node has the capability to communicate with BS directly.

Figure 3 shows a representation of our network model.

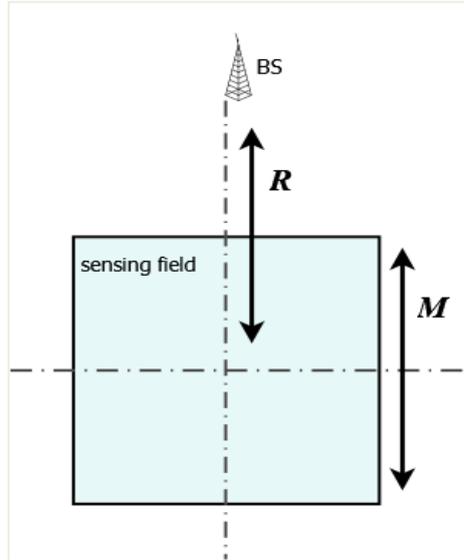
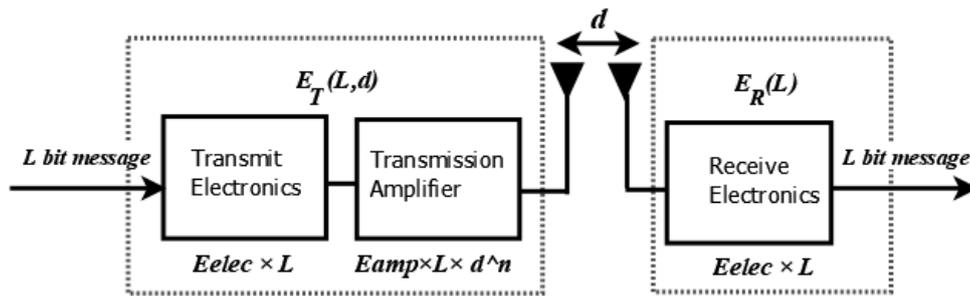


Figure 3. Network model

**3.2. Radio model**

The same radio model described in [23, 24] is adopted here.



In this model, to transmit an L bit message a distance of d, the radio expends the following:

$$E_T(L, d) = E_{elec} \times L + E_{amp} \times L \times d^n \tag{1}$$

To receive the L bit message, in the model, the radio expends the following:

$$E_R(L) = E_{elec} \times L \tag{2}$$

The sensor dissipates  $E_{elec}$  on running the transmitter’s or the receiver’s circuitry. In the transmission, the sensor would also dissipate  $E_{amp}$ , and the value depends on the distance between the sensor and the receiver node. In addition, the sensor will dissipate additional energy in the case of transmission. In our adopted combined model, if the distance between the sensor and the receiving node is small, the free space model is used, and the sensor would dissipate  $E_{amp} \times L \times d^2$ , where  $E_{amp} = E_{fs} = 100 \text{ pJ/bit/m}^2$ . Otherwise, the multipath fading model is used, and the sensor would dissipate  $E_{amp} \times L \times d^4$ , where  $E_{amp} = E_{mp} = 0.0013 \text{ pJ/bit/m}^4$ .

**4. THE PROPOSED SCHEME**

Our Improved Multi-Chain Routing Protocol (IMCRP) consists of several phases that are executed cooperatively by the BS and sensor nodes throughout the network’s lifetime. Figure 4 shows the IMCRP execution model from the start to the end of the network’s lifetime.

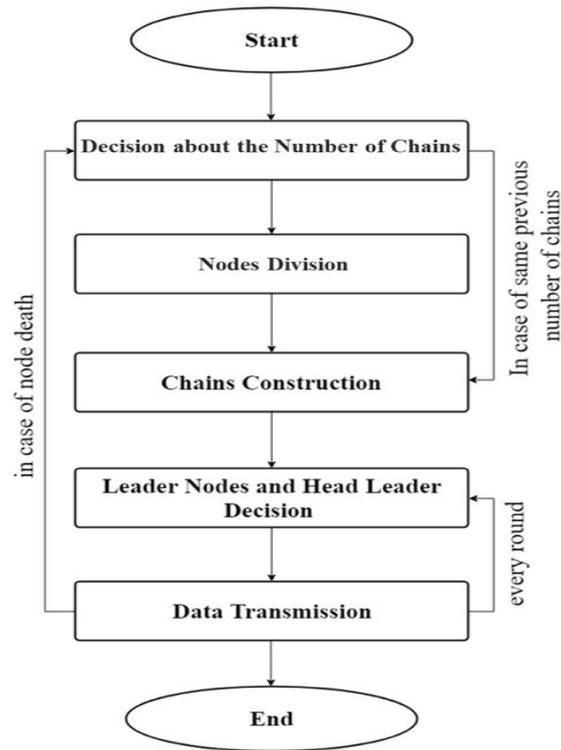


Figure 4. Phases of IMCRP protocol

### 5.1. Decision about the number of chains

Forming the right number of chains ( $K$ ) is an important step in hierarchal routing protocols since it manages less energy consumption. The optimal  $K$  value is affected by many factors, including the used energy model, the nodes' density, the BS location, the size of the sensing field and other factors [25]. In our protocol, we adopted an analytical model that is applicable to our network and energy models that were derived by N. Amini et al. in [24]. The model gives a range of the optimal  $K$  values by

$$k_{opt} = \sqrt{\frac{N \times E_{fs} \times M^2}{2 \pi (E_{amp} d_{toBS}^n - E_{elec}^{Rx})}} \quad (3)$$

where  $M$  is the side of a square sensing field and  $R$  is the distance from the BS to the center of the sensing field, as presented in Figure. 3. The minimum and maximum values are given by the following equations, which are based on the free space and the multipath radio models, respectively.

$$\min(k_{opt}) = \sqrt{\frac{N \times E_{fs} \times M^2}{2 \pi \left( E_{fs} \left( \frac{M^2}{6} + R^2 \right) - E_{elec}^{Rx} \right)}} \quad (4)$$

$$\max(k_{opt}) = \sqrt{\frac{N \times E_{fs} \times M^2}{2 \pi \left( E_{mp} \left( \frac{7 \times M^4}{180} + \frac{2}{3} M^2 R^2 + R^4 \right) - E_{elec}^{Rx} \right)}} \quad (5)$$

To find the optimal  $K$  value among the given range, a simulation has to be performed on the network before the actual deployment of the protocol. However, Figure 5 shows that our simulation with different scenarios and all the various  $K$  values gave similar network lifetimes. Therefore, to dynamically find the optimal  $K$  value by the BS during the network operation, we calculate the optimal  $K$  value in our protocol as the average of  $k_{opt}$  given by To find the optimal  $K$  value among the range given by the pervious equations, a simulation has to be performed on the network before the actual deployment of the protocol. However, we conducted a simulation with different network sizes using all the various  $K$  values. Figures 5

and 6 show the results where all the k values in the range result in similar network lifetimes. According to the results, the optimal K value vary as the nodes density change. Therefore, to dynamically find the optimal K value by the BS during the network operation, we calculate the optimal K value in our protocol as the average of  $k_{opt}$  given by:

$$K = \text{ave} \left( \sum_{i=\min(k_{opt})}^{\max(k_{opt})} i \right) \tag{6}$$

Note that the K value will change throughout the network lifetime depending on number of live nodes N.

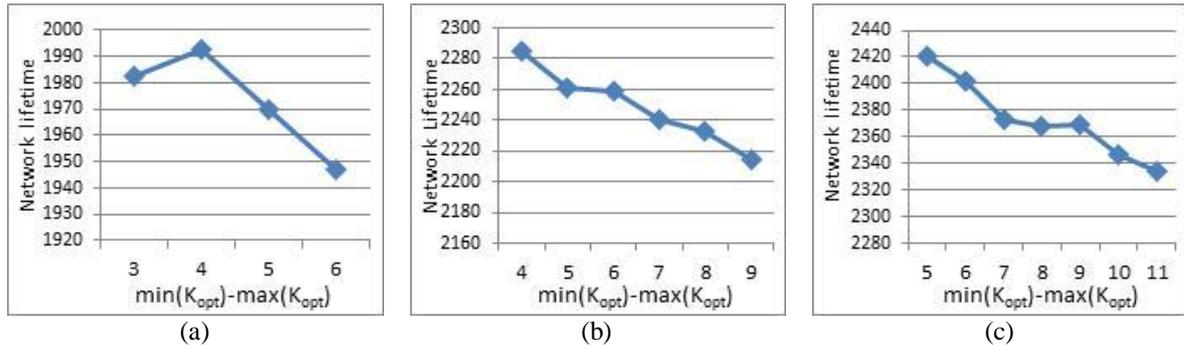


Figure 5. The lifetime of 1002 meter network vs. the number of constructed chains: (a) 100-node network; (b) 200-node network; (c) 300-node network

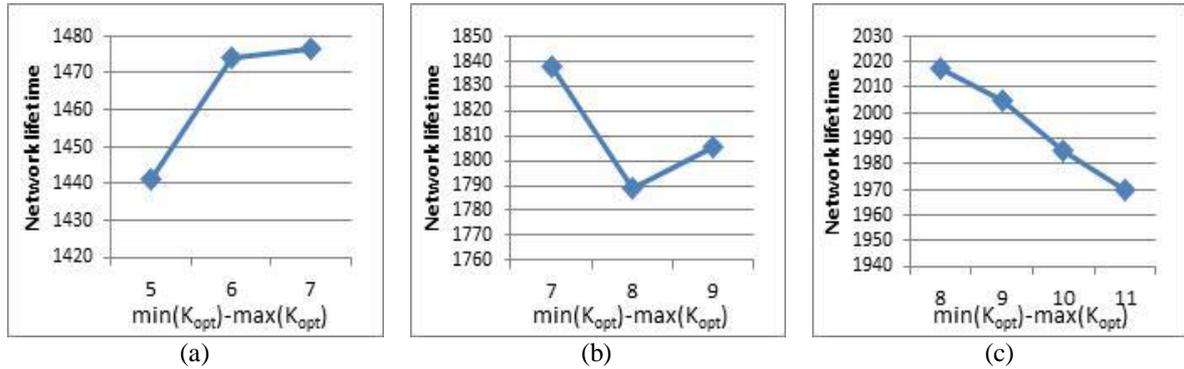


Figure 6. The lifetime of 2002 meter network vs. the number of constructed chains: (a) 100-node network; (b) 200-node network; (c) 300-node network

**5.2. Nodes division**

The K-means clustering algorithm is used in this protocol to divide the nodes in the area into K groups based on the Euclidean distances between the nodes. Therefore, the result of this clustering algorithm is each node joining its nearest group of nodes. At first, K random virtual points are chosen to be the initial centroid points, and each centroid will present a group. Then, the Euclidean distance between nodes and all K centroid points is calculated in order for each node to join the group with the nearest centroid. Note that the Euclidean distance in the two-dimensional space between 2 points  $p = (x_p, y_p)$  and  $q = (x_q, y_q)$  is given by:

$$d(p, q) = \sqrt{(x_p - x_q)^2 + (y_p - y_q)^2} \tag{7}$$

When all nodes are initially assigned into the K groups, a loop is executed where the positions of the centroid points are recalculated in each iteration based on the locations of the member nodes of the group. Node assignments keep changing since the nearest centroid identity may change during the process. The loop ends when no further changes in the centroid points' positions occur and the final assignment of K groups is broadcasted over the network. The centroid of a group of s member nodes is given by:

$$\text{Centroid}(x, y) = \left( \frac{1}{s} \sum_{i=1}^s x_i, \frac{1}{s} \sum_{i=1}^s y_i \right) \quad (8)$$

Note that the whole process of nodes division is performed remotely by the BS node at the start of the network operation and re-executed in case of the change of the optimal k value.

**Algorithm: Network division using k-means**

```

Choose K random points as the initial centroid points
Assign a group ID to each centroid
for each node i in {List of all alive nodes} do
    Calculate distance between node i and all K centroid points.
    Assign node i to the group of nearest centroid point
end for
for each group j in {List of all K groups} do
    Recalculate the centroid point j position based on the
coordinates of group j's nodes. Equation 5 applied.
end for
Compare the last and pervious centroid points positions.
0:
    Repeat steps 3-10 until no changes in the centroid points position.
1:

```

### 5.3. Chain construction

In contrast to previous phases, the chain construction is a distributed process in which all the nodes have to communicate and cooperate to construct the chain of their group. However, the nodes' communication during this phase is limited to only inside their group, and each chain is constructed independently of the rest of the K chains. The chaining mechanism is done using an enhanced greedy algorithm [5]. The algorithm allows the node to have multiple connections, and more nodes joining the chain leads to better choices for the remaining nodes to connect. Moreover, in the case of node death, only the affected chain is re-constructed.

**Algorithm: Chain formation by enhanced greedy algorithm**

```

for each group i in {List of K groups} do
    Set the node which is nearest to centroid as EndNode
    for each node j in {List of nodes didn't join the chain} do
        Calculate distance between node j and EndNode
    end for
    Select the nearest node to be NextNode waiting to join the chain
    for each node h in {List of nodes joined the chain} do
        Calculate distance between node h and NextNode
    end for
    NextNode join chain i by connecting to nearest node on chain
0:   i
    EndNode ← NextNode
1:
    Repeat steps 3-10 until all nodes in the group join the chain
2:
    end for
3:

```

### 5.4. Leaders and head leader selection

To choose better-fitting nodes to be the chain leaders (CL), the selection method depends on the weight of the node that considers two parameters: the distance of the node to the BS and its residual energy, as shown in (9). In every round, the combined weight of each node is compared with other nodes in the same chain, and the node with the minimum weight is selected as the chain leader of the round.

$$\text{weight}(i) = \frac{\text{Residual Energy}(i)}{\text{DistanceToBS}(i)} \quad (9)$$

Furthermore, one leader node out of the K chain leaders is selected as the head leader (HL) to be responsible for sending data to the BS. The HL collects the data from other leaders through a chain topology that was formed and started from the HL itself.

```

Algorithm: Leaders and head leader selection
  for each chain i in {List of K chains} do
  :
    for each node j in {List of nodes in chain i} do
    :
      Calculate weight of node i.
      weight[i]=residual energy[i]/distanceToBS[i]
    :
    end for
    :
    Select the node with minimum(weight) to be LeaderNode of chain i
  :
  end for
  :
  Select the leader node with minimum(weight) to be HeadLeader of
  : network
  
```

## 5. EVALUATION AND SIMULATION RESULTS

### 5.1. Simulation parameters

To evaluate the performance of the IMCRP, we simulated the IMCRP, IEEPB, and PEGASIS protocols by using MATLAB 2017a. The evaluation criteria adopted in this paper are the following:

- a) Network lifetime
- b) Delay
- c) Overhead

The network and energy parameters used in this simulation are shown in Table 2.

Table 2. Simulation parameters

Parameter	Value
Number of deployed nodes (N)	100, 200, and 300
Deployment area	100 * 100 m
Base Station location	(50,175)
Initial energy	0.5 J
$E_{elec}$	50 nJ/bit
$E_{fs}$	100 pJ/bit/m <sup>2</sup>
$E_{mp}$	0.0013pJ/bit/m <sup>4</sup>
Threshold distance	$\sqrt{E_{fs}/E_{mp}}$
Size of data packet	2000 bits
Data aggregation energy	5 nJ/bit

### 5.2. Simulation results

The results presented are based on the average of 10 simulations. The simulations are performed with three scenarios regarding the number of deployed sensor nodes. Therefore, the number of chains in the IMCRP protocol varies for each scenario since the number of nodes affects the optimal-K value, as mentioned in Section 4.

#### 5.2.1. Network Lifetime

The network lifetime may be defined in two ways:

- a) The time from when the network begins operating until the death of the first node, or
- b) The time from when the network begins operating until the death of the last node.

However, we adopt the second definition in this paper and the results are as follows. In Figure 7(a), the simulation of the 100-node network shows that PEGASIS operation ends at 1664 rounds and the IEEPB ends at 1881. Meanwhile, our IMCRP protocol continued for 1970 rounds with a 4.7% improvement over the IEEPB and an 18.4% over the PEGASIS. In Figure 7(b), the simulation of the 200-node network shows that the lifetime of PEGASIS stays the same as in the 100-node network with 1663 rounds and the IEEPB operation ended at 1831 with 2.7% less performance in comparison with the 100-node scenario. On the other hand, the IMCRP's lifetime increased as expected and continued to 2258 rounds, thus giving a 23.3% improvement over the IEEPB. Figure 7(c) shows the last scenario with the 300-node network's lifetime results. The PEGASIS protocol again gave a similar performance as in 100 and 200-node network scenarios. For the IEEPB, the lifetime ends at only 1785 rounds, which proves that the IEEPB performance decreases as

the number of nodes increases. The high node density in the IEEPB causes larger energy consumption by sensor nodes since each node will have a greater number of connected nodes or “child nodes”. Therefore, the energy spent in receiving data increases as well. However, the IMCRP does not have this problem due to the optimal-K model that takes into consideration the nodes’ density. The proposed protocol gave 2034 rounds with the 300-node network, thus giving a 9% improvement over the IEEPB and 17.4% over the PEGASIS.

The figures show that as soon as some nodes starting to die, the rest of nodes start to lose more energy and die as well. This happen due the distance between the nodes in the chain getting larger therefore losing energy becomes higher. Therefore, the re-clustering of the network as in IMCRP help a lot in this case where the still alive nodes get the opportunity to join their nearest alive nodes cluster after the death of their nearest dead nodes.

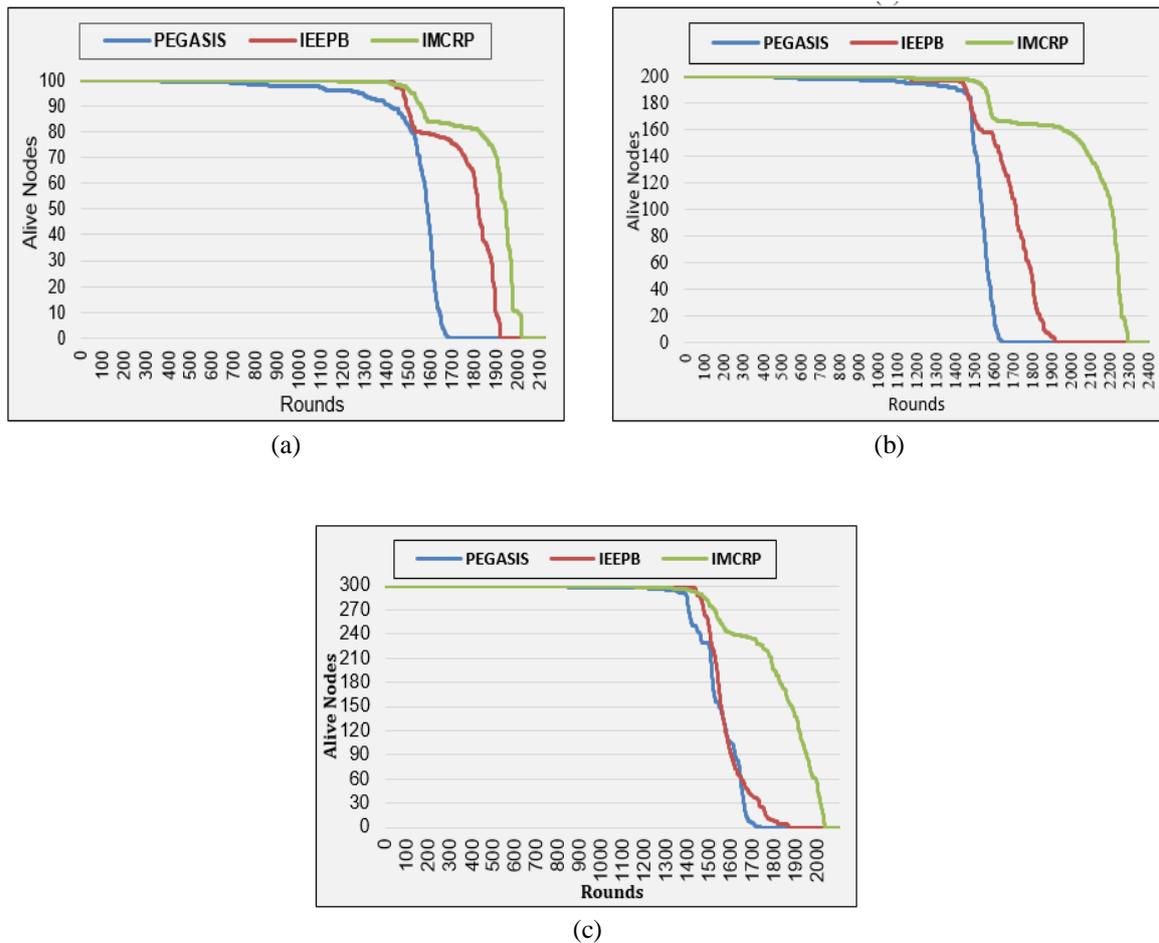


Figure 7. The network lifetime of 1002 meter network: (a) 100-node network; (b) 200-node network; (c) 300-node network

**5.2.2. Delay**

We measured the transmission delay as the maximum number of hops or "nodes" that a data packet travels in order to reach the chain leader. The simulation involved the delay in the IEEPB and IMCRP protocols in which the first is a single-chain protocol and the latter is a multiple-chain protocol. Thus, the delay result in the IEEPB is very large in all three scenarios compared to the IMCRP due to the single chain routing, as shown in Figure 8. In the 100-node network, the IMCRP network will start with dividing the 100 nodes into 4 chains based on (3), (4), and (5) while the IEEPB network will form a single chain containing all the 100 alive nodes. Therefore, Figure 8(a) shows that the IEEPB has an average delay of 40 hops and that the IMCRP has an average delay of 18 hops due the smaller length of chains compared to

IIEPB. For the second scenario with double the number of nodes, Figure 8(b) shows that the IIEPB has double the delay with an average of 80 hops due the single chain length increase, while the delay in IMCRP has increased only 38% with an average of 25 hops as the number of chains “k value” changed too and large chains are being avoided. Figure 8(c) shows that IIEPB delay keeps rising with an average of 110 hops and that the IMCRP managed to stay low at an average of 30 hops.

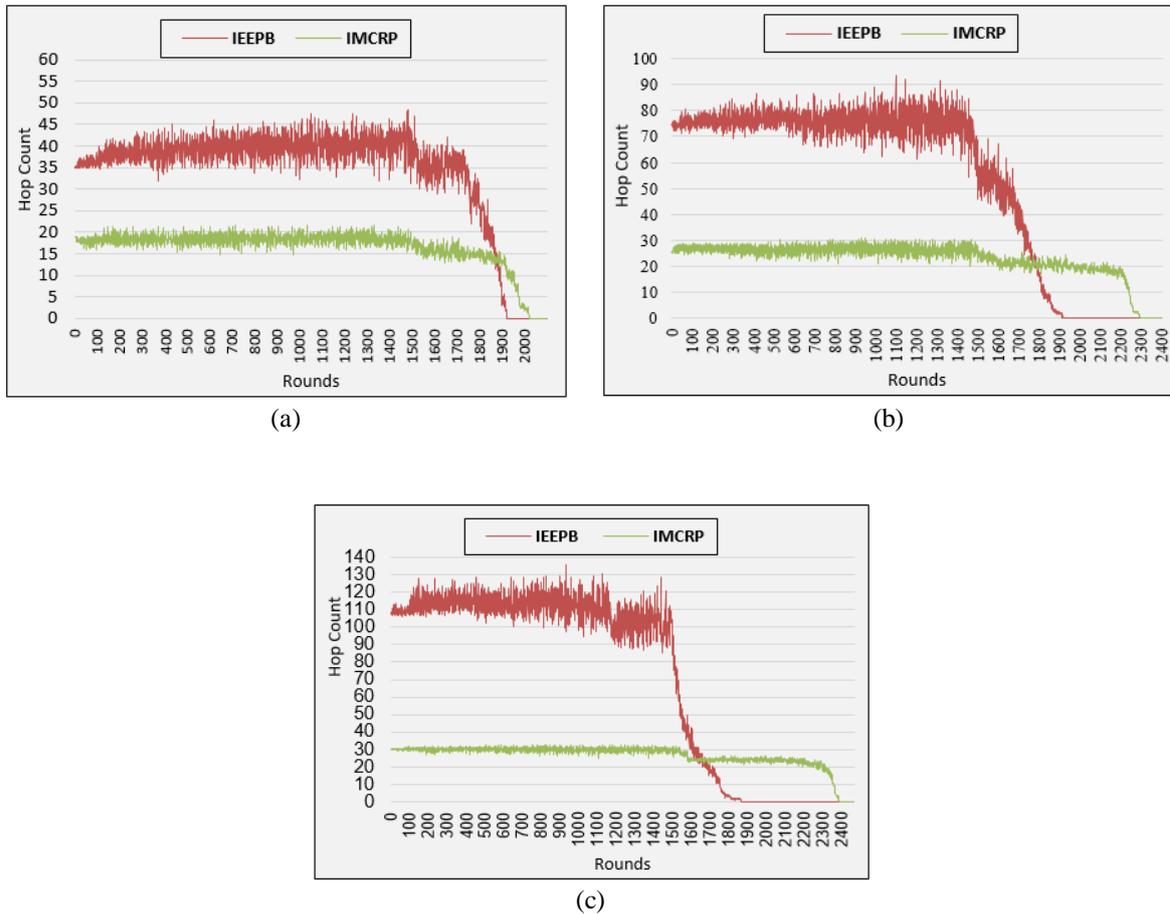


Figure 8. The transmission delay of 1002 meter network: (a) 100-node network; (b) 200-node network; (c) 300-node network

**5.2.3. Overhead**

Another important metric we considered in our evaluation is the overhead. The overhead that resulted from the control packets to establish the routing path in hierarchical routing protocols affects the overall performance and exhausts the sensors’ energy. In our simulations, we calculated the number of control packets exchanged between the sensor nodes and the sensor nodes and the BS during the chain construction phase. Note that the number of control packets is affected by the number of nodes in a single chain and the used algorithm to construct the chain. Figure 9 shows the total number of control packets throughout the network’s lifetime for all three scenarios. In the 100-node network, the results are close. PEGASIS had a total of 135014 control packets, IIEPB had 149029 control packets, and IMCRP had the lowest overhead with 62353 control packets. In the 200-node network, PEGASIS’s overhead increased dramatically with 675703 control packets, and the same occurred in the IIEPB with 786412. Meanwhile, in the IMCRP, a more reasonable increase occurred with 176100 control packets. Finally, with the 300-node network, PEGASIS had 1339722 control packets, and IIEPB had 1765133 control packets, while IMCRP maintained 324350 control packets. Since the advanced greedy algorithm used in IIEPB protocol has more complexity than the greedy algorithm used in PEGASIS, the number of control packets needed is higher. However, IMCRP protocol managed to get lower control packets than both PEGASIS and IIEPB due the fewer number of nodes in each chain.

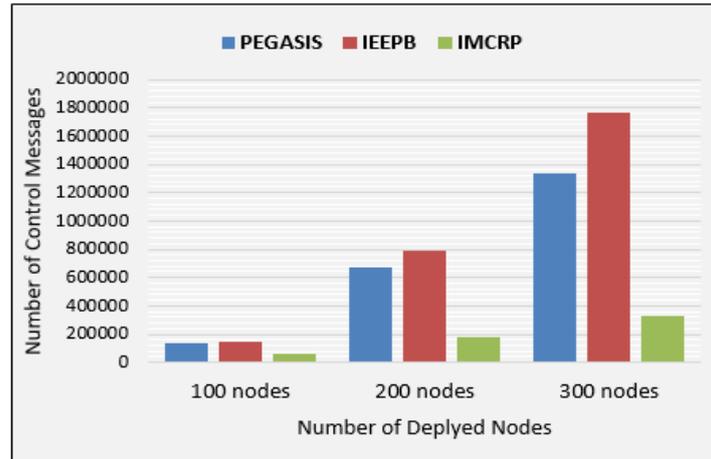


Figure 9. The overhead in 100-, 200-, and 300-node networks

### 5.3. Statistical inference

Pair wise comparison is done between our IMCRP and PEGASIS or IEEPB after each round regarding the number of alive nodes (AliveNodes) as it forms a bivariate variable. Paired t-test is used for statistical inference and the null hypotheses  $H_0$  and the alternative hypotheses  $H_1$  in both comparison cases for all three scenarios 100, 200, and 300 deployed nodes are as follows:

$$\begin{aligned}
 H_{0_{N=100}} &: (\text{AliveNodes}_{\text{IMCRP}} = \text{AliveNodes}_{\text{PEGASIS}}) \\
 H_{1_{N=100}} &: (\text{AliveNodes}_{\text{IMCRP}} > \text{AliveNodes}_{\text{PEGASIS}}) \\
 H'_{0_{N=100}} &: (\text{AliveNodes}_{\text{IMCRP}} = \text{AliveNodes}_{\text{IEEPB}}) \\
 H'_{1_{N=100}} &: (\text{AliveNodes}_{\text{IMCRP}} > \text{AliveNodes}_{\text{IEEPB}})
 \end{aligned}$$

$$\begin{aligned}
 H_{0_{N=200}} &: (\text{AliveNodes}_{\text{IMCRP}} = \text{AliveNodes}_{\text{PEGASIS}}) \\
 H_{1_{N=200}} &: (\text{AliveNodes}_{\text{IMCRP}} > \text{AliveNodes}_{\text{PEGASIS}}) \\
 H'_{0_{N=200}} &: (\text{AliveNodes}_{\text{IMCRP}} = \text{AliveNodes}_{\text{IEEPB}}) \\
 H'_{1_{N=200}} &: (\text{AliveNodes}_{\text{IMCRP}} > \text{AliveNodes}_{\text{IEEPB}})
 \end{aligned}$$

$$\begin{aligned}
 H_{0_{N=300}} &: (\text{AliveNodes}_{\text{IMCRP}} = \text{AliveNodes}_{\text{PEGASIS}}) \\
 H_{1_{N=300}} &: (\text{AliveNodes}_{\text{IMCRP}} > \text{AliveNodes}_{\text{PEGASIS}}) \\
 H'_{0_{N=300}} &: (\text{AliveNodes}_{\text{IMCRP}} = \text{AliveNodes}_{\text{IEEPB}}) \\
 H'_{1_{N=300}} &: (\text{AliveNodes}_{\text{IMCRP}} > \text{AliveNodes}_{\text{IEEPB}})
 \end{aligned}$$

The test statistic  $t$  with  $n-1$  degrees of freedom is defined as

$$t = \text{Mean} / (SD / \sqrt{n-1}) \quad (10)$$

Where Mean and SD are the mean and standard deviation of the difference of the number of alive nodes in two large samples of size  $n$ , as  $n$  is the total number of rounds. The 95% confidence interval is defined as

$$\text{Mean} \pm t_{0.05} * \left( \frac{SD}{\sqrt{n-1}} \right) \quad (11)$$

Table 3 shows the result of paired-t test obtained by pair wise testing of IMCRP with PEGASIS and IEEPB. It shows the mean, standard deviation and the lower and upper limits for the 95% confidence interval. Where  $p$  indicates the probability of the calculated value to obey the null hypotheses. A value of  $p < 0.05$  indicates that the null hypotheses are rejected at 5% significance level and the alternatives hypotheses are accepted at 95% confidence level.

Table 3. The results of paired t-test

Comparison Protocol	N	Mean	Standard Deviation	Std. Error Mean	95% Confidence interval		Significance level	t value	P value
					Lower	upper			
PEGASIS	100	16.46492	27.63920	.61481	15.25919	17.67065	Reject at 5%	26.780	<10 <sup>-3</sup>
IIEPB		5.23780	12.56323	.27946	4.68974	5.78586	Reject at 5%	18.743	<10 <sup>-3</sup>
PEGASIS	200	48.78091	67.95142	1.41781	46.00059	51.56123	Reject at 5%	34.406	<10 <sup>-3</sup>
IIEPB		34.53592	56.03568	1.16919	32.24314	36.82869	Reject at 5%	29.538	<10 <sup>-3</sup>
PEGASIS	300	40.23540	70.03119	1.52857	37.23773	43.23307	Reject at 5%	26.322	<10 <sup>-3</sup>
IIEPB		36.30231	66.83366	1.48154	33.39682	39.20780	Reject at 5%	24.503	<10 <sup>-3</sup>

## 6. CONCLUSIONS

Chain-based routing was discussed by including some of the algorithms concerning chain constriction and network partitioning. Then, we proposed an improved IMCRP protocol, which proved to provide efficient energy consumption, low transmission delay, and low control overhead in comparison with other protocols. The protocol constructed multiple chains using an advance greedy algorithm and k-mean clustering. Moreover, the main design improvement allowed the protocol to dynamically select the optimum number of chains throughout the network lifetime based on the current live nodes. At the end, paired t-test is used to prove the significance of data.

## ACKNOWLEDGMENT

This project was funded by King Abdulaziz City for Science and Technology (KACST), under grant number 0017-009-02-17-1. The authors, therefore, acknowledge with thanks KACST for technical and financial support.

## REFERENCES

- [1] A. R. Bhawna Sharma, "Energy Efficiency in Distributed Clustering of WSN using HEED Protocol," *International Journal of Engineering Research & Technology (IJERT)*, vol. 5, no. 06, Jun. 2016.
- [2] S. K. Singh, M. Singh, and D. K. Singh, "Routing Protocols in Wireless Sensor Networks - A Survey," *International Journal of Computer Science & Engineering Survey*, vol. 1, no. 2, pp. 63–83, Nov. 2010.
- [3] S. Lindsey and C. S. Raghavendra, "PEGASIS: Power-efficient gathering in sensor information systems," in *Proceedings, IEEE Aerospace Conference*, vol. 3, pp. 1125–1130, 2002.
- [4] Y. Yu and Y. Song, "An Energy-Efficient Chain-Based routing protocol in Wireless Sensor Network," in *2010 International Conference on Computer Application and System Modeling (ICCASM 2010)*, vol. 11, pp. V11-486–V11-489, 2010.
- [5] S. Feng, B. Qi, and L. Tang, "An improved Energy-Efficient PEGASIS-Based protocol in Wireless Sensor Networks," in *2011 Eighth International Conference on Fuzzy Systems and Knowledge Discovery (FSKD)*, vol. 4, pp. 2230–2233, 2011.
- [6] N. D. Tan and N. D. Viet, "SCBC: Sector-chain based clustering routing protocol for energy efficiency in heterogeneous wireless sensor network," in *2015 International Conference on Advanced Technologies for Communications (ATC)*, pp. 314–319, 2015.
- [7] M. Dorigo and M. Birattari, "Ant Colony Optimization," in *Encyclopedia of Machine Learning*, Springer, Boston, MA, pp. 36–39, 2011.
- [8] S. Ghosh, S. Mondal, and U. Biswas, "Enhanced PEGASIS using ant colony optimization for data gathering in WSN," in *2016 International Conference on Information Communication and Embedded Systems (ICICES)*, pp. 1–6, 2016.
- [9] M. Gupta and L. Saraswat, "Energy aware data collection in wireless sensor network using chain based PEGASIS," in *International Conference on Recent Advances and Innovations in Engineering (ICRAIE-2014)*, pp. 1–5, 2014.
- [10] S. M. Jung, Y. J. Han, and T. M. Chung, "The Concentric Clustering Scheme for Efficient Energy Consumption in the PEGASIS," in *The 9th International Conference on Advanced Communication Technology*, vol. 1, pp. 260–265, 2007.
- [11] K. H. Chen, J. M. Huang, and C. C. Hsiao, "CHIRON: An energy-efficient chain-based hierarchical routing protocol in wireless sensor networks," in *2009 Wireless Telecommunications Symposium*, pp. 1–5, 2009.

- [12] S. Mao and Y. T. Hou, "BeamStar: An Edge-Based Approach to Routing in Wireless Sensor Networks," *IEEE Transactions on Mobile Computing*, vol. 6, no. 11, pp. 1284–1296, Nov. 2007.
- [13] M. F. Shekh, A. Kumar, and G. Sharma, "Improved energy-efficient chain-based routing protocol for edge-based wireless sensor networks," in 2016 *International Conference on Communication and Signal Processing (ICCSP)*, pp. 2225–2231, 2016.
- [14] M. Hadjila, H. Guyennet, and M. Feham, "A Chain-Based Routing Protocol to Maximize the Lifetime of Wireless Sensor Networks," *Wireless Sensor Network*, vol. 05, no. 05, pp. 116-120, 2013.
- [15] T. M. Jawad and N. A. Ali, "Using K-means clustering algorithm with Power Efficient Gathering in Sensor Information Systems Protocol," *International Journal of Computer Science & Engineering Technology*, vol. 6, no. 1, p. 5, 2016.
- [16] T. Li, F. Ruan, Z. Fan, J. Wang, and J.-U. Kim, "An Improved PEGASIS Protocol for Wireless Sensor Network," 2015 *3rd International Conference on Computer and Computing Science (COMCOMS)*, pp. 16–19, 2015.
- [17] B. Patel and J. Munjani, "Chain based Routing Protocol for Wireless Sensor Network," *Int. J. Comput. Appl.*, vol. 134, no. 4, pp. 21–24, Jan. 2016.
- [18] G. K. Bhatti and J. P. S. Raina, "Cuckoo based Energy Effective Routing in Wireless Sensor Network," vol. 3, p. 4, 2014.
- [19] A. K. Mishra, R. U. Rahman, R. Bharadwaj, and R. Sharma, "An enhancement of PEGASIS protocol with improved network lifetime for Wireless Sensor Networks," in 2015 *IEEE Power, Communication and Information Technology Conference (PCITC)*, pp. 142–147, 2015.
- [20] B. Singh and E. S. Kaur, "An Improved Energy-Efficient BBO-Based PEGASIS Protocol in Wireless Sensors Network," 2014.
- [21] S. Mahajan, J. Malhotra, and S. K. Sharma, "Improved Enhanced Chain Based Energy Efficient Wireless Sensor Network," 2013.
- [22] M. A. Ali and A. K. Rajpoot, "Development of energy efficient routing protocol using Hop PEGASIS in Wireless Sensor Networks," 2014.
- [23] W. B. Heinzelman, A. P. Chandrakasan, and H. Balakrishnan, "An application-specific protocol architecture for wireless microsensor networks," *IEEE Transactions on Wireless Communications*, vol. 1, no. 4, pp. 660–670, Oct. 2002.
- [24] N. Amini, A. Vahdatpour, W. Xu, M. Gerla, and M. Sarrafzadeh, "Cluster Size Optimization in Sensor Networks with Decentralized Cluster-Based Protocols," *Comput Commun*, vol. 35, no. 2, pp. 207–220, Jan. 2012.
- [25] V. Kumar, S. B. Dhok, R. Tripathi, and S. Tiwari, "A Review Study on Analytical Estimation of Optimal Number of Clusters in Wireless Sensor Networks," *Transactions on Networks and Communications*, vol. 2, no. 5, pp. 75–103, Nov. 2014.