Journal of Theoretical and Applied Information Technology
31st March 2015. Vol.73 No.3

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

CENTROID DYNAMIC SINK LOCATION FOR CLUSTERED WIRELESS MOBILE SENSOR NETWORKS

[1]MANAR A. MIZHER, [2]SALEH H. AL-SHARAEH, [3]MEI CHOO ANG, [4]AYMAN M. ABDALLA, [5]MANAL A. MIZHER
[1,3,5]Institute of Visual Informatics, University Kebangsaan Malaysia, Malaysia
[2]King Abdullah II School of Information Technology, The University of Jordan, Jordan.
[4]Faculty of Science and I.T, Al-Zaytoonah University, Jordan.
E-mail: [1] manar@siswa.ukm.edu.my,[2] salsharaeh@yahoo.com,[3]amc@ukm.my,
[4]ayman@zuj.edu.jo ,[5] manal@siswa.ukm.edu.my

ABSTRACT

The wireless sensor network consists of three main components: a large number of small-sized sensors, a remote sink connected to the internet, and a cluster head whose existence depends on the overall network structure. The sensor in the wireless sensor network can be deployed through many ways, such as a simple model, a random walk model, and a random direction model. There were studies that examined networks and their various installation methods to save energy consumption and increase the network lifetime. These were usually achieved by formatting the network structure with one or multi-sinks, with or without clusters, or using static or mobile components such as sink, cluster heads, and sensors. In addition, using a homogeneous or heterogeneous environment implies using special devices as cluster heads or electing them from sensors periodically at specified times depending on different protocols. Previous studies did not focus on saving energy when all network's components were mobile. Our scheme, **C**entroid **D**ynamic **S**ink **L**ocation (**CDSL**), focuses on this case and aims to reduce the energy consumption through moving the sink to the optimal location with respect to the cluster heads. The simulation results indicated that the CDSL scheme increases the network lifetime by saving the cluster heads energy. When the sink is mobile, the network lifetime increased in all cases from 14.21% to 53.09% compared to that which use a fixed sink.

Keywords: *LEACH, Mobility model, Cluster Heads, Centroid algorithm, Power saving.*

1. INTRODUCTION

A wireless sensor network (WSN) consists of three main components. First, a large number of tiny sensors, Sensors have the ability to sense, process the acquired information, transmit messages to the sink and communicate to each other. The mobility model is designed to describe the movement pattern of mobile sensors and their locations during each round. These patterns may play a critical role in determining the protocol performance. The sensor nodes can be deployed in different models such as the random walk model (RWM), the random direction model (RDM), and the simple model (SM) [1], where nodes move independently with the same average speed as in [2]. In [3, 4], however, they move randomly. Mobile sensors in [5] are deployed depending on specific model. Authors in [6] deployed the stationary sensors randomly where the mobile sensor depended on the probabilistic detection model in the ABC algorithm. Second, a remote sink connected to the internet is engaged to give commands to all sensor nodes, and to gather information from the sensor nodes. The majority of authors used fixed sinks as in [1, 3-5, 7, 8]. Nonetheless, a minority of authors used moving sinks depending on different techniques with stable sensors, as in [9] where the sink movement depended on the Particle Swarm Optimization (PSO) algorithm. In [10], the sink moved according to the sink migration policy. In [11], authors compared the routing protocols in wireless sensor network using mobile sink. Third, a wireless sensor network may or may not have cluster heads (CH). In a clustering WSN nodes are grouped into clusters if the sensor nodes are homogeneous. In addition, if they have the ability of data-aggregation (data-fusion) and routing sensed information to the sink, the network needs the election of a cluster-head. However, in a heterogeneous sensor network, which consists of two types of nodes (sensor nodes and cluster head nodes), there is no need to elect cluster heads. Many algorithms have been introduced to deal with cluster issues such as:

cluster formation, cluster-head election, merging two or more clusters, cluster division, reformation of clusters, cluster-head reelection process, expanding cluster with a specific purpose like forwarding information, calculating the optimal number of clusters-heads in sensor networks, and gathering data in sensor networks and saving power. Authors in [7] grouped sensors into clusters dynamically in each tour as in LEACH. In [8], the BEE-C was algorithm proposed for formatting clusters, while in [5], nodes with higher remaining power were selected to act as cluster heads.

A WSN can be constructed according to many types such as static, hybrid, and mobile. A wireless network of mobile sensors can be employed in different fields, including military applications. Sensor networks can be used in surveillance missions and can be used to detect moving targets, the presence of micro-agents, or chemical gases [12]. Hence, data communicates between cluster heads and the sink is a critical issue in the network. If the distance is very long, it affects the signal quality of transmission and the total energy consumption of the WSN, which leads to the reduction of the network lifetime.

The rest of this paper is organized as follows. In Section 2, related works will be presented. Section 3 will discuss the problem statement, and our centroid dynamic sink location for clustered mobile WSN will be introduced in Section 4. Simulation results will be presented and analyzed in Section 5. Section 6 will provide the conclusion and suggests future work.

2. RELATED WORKS

Many research articles depended on the Low-Energy Adaptive Clustering Hierarchy protocol (LEACH), which is one of the most popular hierarchical routing algorithms for sensor networks [13]. its characteristics and benefits are summarized as follows: fixed base station is located far from the sensors, and all nodes in the network are homogeneous and energy constrained, radio channel is symmetric, nodes organize themselves independently from each other into local clusters where one node acts as the CH, so no extra negotiation is required to determine the CH, it includes randomized rotation of the CH position to balance the energy spent per round by each sensor node, it also compresses the amount of data being sent from the cluster heads to the base station by performing local data fusion, and sensors elect themselves to be CHs at any given time with a certain probability, where node n chooses a random number X between 0 and 1; if X less than T(n) then n becomes a CH. This is shown in Eq. (1).

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

Where p = the desired percentage of cluster heads (for example p = 0.05), r = the current round, G = set of nodes that have not been a CH in the last $1/p$ rounds.

CHs broadcast their status to other nodes, each node determines to which cluster it wants to belong by choosing a CH that requires the minimum communication energy. Transmission cost and receiving cost equations can be seen in Section 3. On the other hand, one essential issue in wireless sensor networks is how to gather sensed information in an efficient way to save energy, since the energy is a critical resource in a sensor node. There were many algorithms to improve the LEACH protocol to prolong the lifetime of the network. In [1], the authors considered the cluster-based architecture. They provided distributed clustering algorithms for mobile sensor nodes that minimize energy wasting for data gathering in a wireless mobile sensor network. The two main steps in their clustering algorithm are the cluster-head election step and the cluster formation step. They first proposed two distributed algorithms for cluster-head election: Algorithm of Cluster-head Election by Location (ACE-L), and Algorithm of Cluster-head Election by Counting (ACE-C). Then, considering the impact of node mobility by presenting the Clustering with Mobility mechanism (CM), they provided a mechanism to have a sensor node select a proper cluster-head to join in cluster formation. Their clustering algorithms achieved the following three objectives: (1) at least one cluster-head is elected, (2) number of cluster-heads are uniform, and (3) all clusters have the same cluster size. They validated their algorithms through an extensive experimental analysis with a RWM model, a RDM model, and a SM model.

The characteristics of cluster-head election algorithms used in [1] are: Clustering with Mobility (CM) mechanism is used to form clusters after electing all cluster heads for the current round, where each sensor does not neglect its mobility when deciding which cluster-head to join. ACE-C characteristics and benefits: ACE-C elects CHs in a round robin fashion, by giving each node a unique ID, a node decides independently whether it is a CH in the current round, ACE-C uses a loop to decide. During iteration, only one node has a suitable ID to become a CH, so the number of iterations is increased by increasing the number of CHs in the network. After determining all CHs, the CHs send an advertisement for non-CH nodes to join suitable clusters using the CM mechanism. ACE-L characteristics and benefits: different from LEACH protocol only in the CH election phase. Given some fixed main reference point (MRP) in the whole area, the closest nodes to MRP will be CHs. Each node will determine which MRP is the closest to it, and then will calculate the delay time depending on the distance to this MRP. When the election phase is started, and every node evaluates delay time to closest MRP, each node will wait until its delay time finishes. Then, the node with the shortest time will gain the link and send its advertisement to become a CH. Nodes with longer delay time will stop their timer, and will decide which CH they will join. Use CM in the cluster formation step.

3. PROBLEM STATEMENT

As mentioned before, one of the key challenges in WSN's environment is the limited battery power in each node. Despite the advancement made in battery technology regarding size and/or power capacity, power consumption remains an important factor to be considered. In cluster-based routing protocol, cluster heads consume more power than other nodes because they have special roles. Therefore, it is important to consider the power consumption metric in the process of cluster head election. Equations (2) and (3) were used in LEACH [13] to estimate the node consumed power where the transmission cost and receiving cost, respectively, for an *l*-bit message with a distance *r* are:

$$\text{Transmission: } E_{TX}(l,r) = E_{elec}\times l + E_{amp}\times l\times r^2 \quad (2)$$

$$\text{Receiving: } E_{RX}(l,r) = E_{elec}\times l \quad (3)$$

Where:

ETx(*l*,*r*): The cost of transmitting an l-bit message for a distance *r*.

E_{elec}: The power consumption of the circuit itself.

E_{amp}: The power consumed by the amplifier for transmitting packets.

l: The size of message in bits.

r: The distance between sender and receiver.

ERx(*l*,*r*): The cost of receiving an l-bit message for a distance *r*.

Consequently, the total power consumption by the cluster head can be calculated by Eq.(4) [1]:

$$TPC(CH_x) = ETCH_x(l,r) \quad (4)$$

Where:

TPC (*CHx*): The total powers consumed by the cluster head *X*

$ETCH_X(l,r)$: The summation of the cost of transmitting *l*-bit messages for a distance *r* by the cluster head *X*, where *l* is the size of message in bits, and *r* is the distance between sender and receiver.

As seen in Eq. (2), the distance between sender and receiver can affect the node lifetime. Therefore, we aim to find an appropriate approach that can reduce the value of this distance. It is clear that reducing CH's power consumption will increase the total lifetime of the whole network and keep it operating for as long as possible, which is desirable. In CDSL, we attempt to take advantage of the mobility sink concept, and at the same time to decrease power consumption in CHs. The position of a sink can greatly affect prolonging the lifetime of the network, since the power consumption is a function of the distance between the receiver and the transmitter. The position of the sink is very critical to the lifetime of the network. We consider the problem of dynamic sink movement where the sink is allowed to move in the sensing area. It moves to sense information about low power consumption for cluster heads to maximize the lifetime of the network. The sink changes its location depending on the centroid algorithm, which is the average coordinate among the cluster heads to choose the best among nearest locations for all cluster heads, as shown in Eq.(5) . Before the sink decides where to move, it needs to know where it can move to, and what the best location is. The cluster head locations are set at each round, when the elected cluster head affects the decision on where the sink can move. For most applications, the sink is able to move periodically in two-dimensional, which has an infinite number of points. However, not every point can be a possible location for a sink because a sink that moves to a point near a cluster head must be close enough to communicate with it.

The centroid *C(x,y)* of a finite set of cluster heads is:

$$C(x,y)= (\frac{x_1 + x_2 + \cdots + x_n}{n} + \frac{y_1 + y_2 + \cdots + y_n}{n}) \quad (5)$$

Where:

$C(x,y)$:The average point between the coordinates locations for cluster heads.

$x_1+x_2 + \ldots + x_n$: Summation of x-axis coordinates for all cluster heads.

$y_1+y_2+\ldots.+ y_n$: Summation of y-axis coordinates for all cluster heads.

The CDSL protocol was based on the work in [1], but with a dynamic sink located in a new location at each round instead of a fixed one. This is done when the sink calculates a centralized suitable place between all cluster heads, and then the sink moves there and informs CHs to send aggregated data to it in his new location.

4. CENTROID DYNAMIC SINK LOCATION FOR CLUSTERED MOBILE WSN (CDSL)

Centroid **D**ynamic **S**ink **L**ocation for clustered mobile WSN (CDSL) aims to reduce the total amount of the power consumed by the sensors in the network. This should especially benefit cluster heads, which are more sensitive to energy drains for their roles in the network. Protocol CDSL will be used after the first round to move the sink to an optimal location between CHs based on centroid point after electing CHs. To be more efficient, we consider investigating CDSL for different clustering protocols, mobility models, and round duration time. The dynamic sink will be aware of new CH locations to move into the nearest optimal location between them. The network environment in CDSL as in [1], where all sensor nodes are homogeneous, mobile, and power limited. Each sensor is supported by an LFS. The sink in CDSL at the first round is located at (300,100) as in [1], but after electing CHs starting at round two until the end of system lifetime, the sink will keep moving into new locations. We used three mobility models RWM, RDM, SM, with CDSL to achieve the three objectives of a good clustering scheme. The objectives are: at least one cluster head is elected in each round, the generated number of cluster heads at each round is uniform, and they have the same cluster size. Protocol CDSL partially differs from previous related work in the data transmission phase between CHs and the sink. This difference will be discussed later. In order to implement the previously described scheme, the sink needs to have information about the new CH location. This information can be sent by new CHs periodically after being elected to sink by exchanging control messages. Every control message intends to provide a certain piece of knowledge or to invoke certain action. In our case, we have to exchange different types of these control messages to enable CDSL to work correctly. The important information a sink needs to know includes the new elected CH locations. The sink uses this information to obtain its new location. This control message offers the way to overcome this challenge. In the first round, the first aggregation data item is sent to the sink in its preliminary location because all sensors initially know its location. From the second round and above, after electing new CHs, each sensor must send its location to the sink in a control message called CH_ID. This field contains the identifier of the sending node. The CH_ID field gives information to the sink about new elected cluster heads. Although this control message sent by CHs at each round will dissipate low power from the CHs battery, it will be useful for saving CH total power at the end of the round. We shall discuss this point in depth later.

As mentioned earlier, after the sink gets information of all new cluster heads elected locations in the current round, it must also move to a new optimal centroid location between cluster heads to save their power. The distance between CHs and the sink is critical, as shown in Eq.(2). Depending on the centroid algorithm shown in Eq.(5), the sink calculates its new location, where the sink in this algorithm treats the location in the first round as an initial center of the algorithm and all cluster heads as points in the region that need to know their center. After the sink finds an optimal center, it must move to that center where it will have the nearest location for all CHs. When the sink moves to its new location, it must send a control message with its new location to all CHs. When reserving a sink-location control message, cluster heads start sending aggregated data to the sink. This control message also dissipates low power from the cluster head, but as a whole, CHs will save their battery power because the new best location for the sink in the current round reduces the distance between them and the sink as mentioned in Eq.(2).

Protocol CDSL runs clustering protocols based on LEACH, CM, ACE_C, and ACE_L; all with a dynamic mobile sink. In addition, CDSL uses the mobility models that are RWM, RDM, and SM. Comparisons between CDSL and static sink will be discussed in the next section. We assume that all sensor nodes have the same limited non-rechargeable battery power values at system activation time, which equals one Joule (1 J). The sink has rechargeable hug battery power. For that, we consider saving sensors battery, not for all of these sensors, but only for CHs. Our network is homogeneous. Five sensors or less will take the role of CHs as determined in [1]. They periodically change at each round to choose other sensors to become CHs depending on the clustering algorithm used. After the clustering formation phase, every CH will aggregate data, and then send them to the sink in its initial location. Only in the first round, the sink is located far away in (300,100). LEACH is the base protocol for all protocols used in [1] and in CDSL. Therefore, CHs in this step will dissipate energy depending on the radio model in LEACH as shown in Eq.(2) for transmitting and Eq.(3) for receiving. Protocol CDSL, after the first round, will increase the cost of CHs location control messages and sink new location control messages. Eq.(2) and Eq. (3) are used, but with changing l when sending CH location messages or when CHs receive sink new location messages. This is because a data packet has more bits than a control message. In addition, we must not forget changing r only in Eq.(2) when CHs are sending control messages or when CHs are sending data messages to the sink's new location. Consequently, the total power consumption of the cluster head can be calculated by Eq. (4), taking into account the difference from our equation in the meaning of $ETCHx(l,r)$.

```
CDSL ( )
{Declare: Number of sensors, CHs count, Data msg
  size, Control msg. size, Sink_CHs distance.
  Loop ( if sensor number !=0)
        Choose CHs, where CHs count <=5
  end loop
  CHs transmit there information to all nodes
  if (first round= true)
     CHs transmit its aggregated data to sink in its
     initial location
  else ( ! first round)
   { loop as CHs count as
     found CH location
     Calculate Centroid value
     Move sink to point evaluated by Centroid
     Sink transmit his new location to CHs
     CHs send aggregated data to sink in his new
            location}
```

Figure 1: CDSL Scheme

Pseudo code of the CDSL algorithm is shown in Fig.1. When the algorithm starts, it declares the number of sensors, CHs, data message size, control message size and the initial sink location. After that, while some sensors are still alive, CHs will be chosen from these sensors. Then, CHs send their information to the remaining sensors to join the best CH depending on one of the clustering algorithms used. After that, CHs aggregate data from sensors in their cluster. CHs at the end of the round need to send their aggregation data to the sink. At this stage, we have two cases: First round and other rounds. At the first round, all CHs send data to the sink in its initial location. After that, new CHs are chosen in the next round where they send their location to the sink. Depending on their location, the sink calculates its new location based on the centroid algorithm and sends the new location to the CHs. When CHs want to send their aggregation data, they send them to the sink at its new location.

5. RESULTS AND ANALYSIS

Experiments in [1] and CDSL were simulated using wsn_v1_7 wireless sensor network simulator under windows platform, and they were written in VC++ language. The wsn_v1_7 used a clustered mobility environment with a mobile homogeneous sensor and cluster heads. The original wsn_v1_7 simulator was designed with CM mechanisms to form a cluster after electing cluster heads. It had two algorithms to elect cluster heads: the ACE-C algorithm and the ACE-L algorithm. They were implemented in different mobility deployment models, which were the RWM, RDM, and SM models. The wsn_v1_7 simulator was able to simulate the LEACH protocol, since it is the base of the scheme in [1] and CDSL. We ran all simulations on the same machine to compare the performance of the scheme in [1] against CDSL. The choice of performance metrics was dependent on the goal we aimed to achieve. The system lifetime ends if all sensors are dead. When the system lifetime increases, that means sensors have been saving more battery power. To achieve this, we must prolong cluster heads lifetime by saving their power. To ensure that lifetime is increased, our study must be concerned with the first node to die, the end round at which the death of the last node takes place, and the average of dead nodes. The mobility models used in our simulator are RWM, RDM, and SM. The performance parameters are shown in Table (1) as used in [1] with same values. Each sensor transmission distance can reach up to a maximum of 300 m. We used this range because the sink was fixed away from the sensing area at a location (300,100) as a static sink, and it was the initial location for the sink in first round in CDSL. The elected CHs locations could be the farthest possible from the sink, so they may need their maximum transmission distance to send their aggregation data to the sink.

Table 1: WSN_v1_7 Simulation Parameters

Items	Values
Network area	200m ×200m
Sink initial location	(300,100)
Mobility Models	SM , RWM , RDM
Number of sensor nodes	100
Speed of sensor nodes	Between 0 and 1 m/s
Size of the packet(in a round)	2000 bits
Period of each round	5s , 10s, 15s, 20s
Maximum transmission	>300m
Reference points	(50,50), (150,150), (150,50), (50,150)
ID's of sensor nodes	0-99
Expected number of CHs	5 (or 4 for MRP)

We divided the simulation study sample into two cases. This helped us trace different scenarios and conditions. The first case in the simulator was when the sink was static. The other case was when it was mobile. In order to get consistent results, we used a different duration round time for each simulation study sample case. We chose these round times to study the effects of period length on network lifetime, and we determined rounds not to be too short to enable processing of all phases, and not too long to save elected CHs energy at each round. Hence, we ran each of the clustering protocols (LEACH, ACE-L, ACE-C, and CM) in CDSL and in the static sink with the different mobility models SM, RWM, and RDM, and all of them with different round duration times of 5, 10, 15, and 20 seconds. We ran cases with 5 seconds for 40 experiments, but cases with other times only for 10 experiments.

We started by testing the end round metric in CDSL against the static sink with SM, RWM, RDM run in different round duration times with LEACH protocol, as seen in (Fig.2(a)). At all duration times, the static sink at most reached below 600 rounds, but in CDSL, it reached above 640 rounds and below 720 rounds. In addition, we see that SM has the best system lifetime because in this mobility model, the CM mechanism can predict which CH can be the nearest for the sensor to join when the round starts. This prediction is more precise in SM than in other mobility models because the sensor has fixed speed and direction to move unless it reaches the boundary of the region where it will reverse its direction. Furthermore, we see that mobility models with CDSL have better system lifetimes than static sink because CDSL moves the sink to the best location for all CHs. This leads to saving CHs power consumption by reducing the distance between the sink and CHs. Each point in the chart refers to the average of experiments for that case. For the CM protocol, as seen in Fig.2(b), in all duration times, CDSL has a better average of rounds than the network with a static sink. When CDSL uses the SM model, it has a longer lifetime of the network than RWM and RDM because the prediction is more precise in SM than in other mobility models. In the CM_Counting protocol, as seen in Fig.2(c), all duration times with a static sink at most reached below 580 rounds. In CDSL, it has better rounds than static sink where mobility models gave the best average when using the SM model. Also, using the CM_Location protocol, as seen in Fig.2 (d), all duration times with SM had a better network lifetime with CDSL than using a static sink. As a result of comparing the effect of clustering protocols in CDSL with static sink, as shown in Fig.2, we conclude that the CM_Location protocol had the longest network lifetime, where the end round reached near 2000 rounds. We can also conclude that SM with all protocols had the highest end round compared to other mobility models because sensor speed and direction were less variable, since the sensor can reverse its direction with the same speed only when reaching the boundary of the network region. In contrast, RWM and RDM had the lowest end round numbers because the sensor node in the RWM model calculated new speed and direction when it reached the boundary of the network to modify its position, while in RDM model, it changed its direction. . This makes the CM mechanism less useful for predicting which CH is more helpful for the sensor to join as in SM.

A comparison of CDSL using three mobility models and with different round duration times is shown in Fig.3. From this comparison, we conclude that CDSL with CM_location and SM had the highest prolonging network lifetime compared to the others because CM_Location chose the best location for CHs between sensors to save their power. In addition, SM increased the CM mechanism to predict which CH is the best for the sensor to join. Lastly, CDSL saved CHs power by reducing the distance between CHs and the sink.

This is because CDSL moved the sink to the nearest location for all CHs.

When studying the effect of RWM on all clustering protocols, as seen in Fig.4(a), we found that CDSL with CM_Location saved sensors energy more than the others, while static LEACH had the lowest energy saving. When studying the effects of RDM and SM on the clustering protocol, we found that CM_Location protocol with CDSL, as seen in Fig.4(b), saved sensors energy more than other clustering protocols when using RDM as the mobility model. Also, CM_Location saved more energy with SM, as seen in Fig.4(c). Dynamic sink in CDSL with all duration times with SM saved more energy than when implemented with other mobility models and more than static sink, as seen in Fig.4; where the end round reached near 2000 rounds. Also from the same figure, we conclude that CM_location and CM_Counting are better protocols with both static sink and CDSL than other protocols, where LEACH and CM with CDSL have less network lifetime than CM_location and CM_Counting with static sink. However, these latter protocols are still better than LEACH and CM with static sink because CM_Location and CM_Counting used different algorithms to choose the CHs. They also used CM mechanism in each sensor to predict the best CH to join in the next round. The number of dead nodes in LEACH with RWM had an average of 400 to 500 rounds when the sink was static. When it was dynamic, however, it had an average between 500 to 650 rounds. When using RDM, the number of dead nodes for CDSL in LEACH had an average of 600 to 800 rounds, and with static sink, it remained at the RWM average of 400 to 600 rounds. When LEACH ran with static sink and SM, the dead node average was 200 to 700 rounds, and with CDSL the average was 400 to 800 rounds. Therefore, LEACH with CDSL had a longer network lifetime than with static sink because CDSL saved CHs battery power by reducing the distance between the sink and CHs. When the CM protocol ran with static sink and RWM, dead nodes had an average of 200 to 600 rounds, but with CDSL, the average was 500 to 800 rounds. When CM protocol ran with static sink and RDM, the dead node average was 200 to 500 rounds, and with CDSL, the average was 500 to 800 rounds. When CM was run with static sink and SM, the dead node average was 200 to 700 rounds, and when it was run with CDSL the average of dead nodes was 400 to 900 rounds. From the above results, we conclude that CM with CDSL can prolong the network lifetime more than CM with static sink. When CM_Counting was run with static sink and RWM the dead node average was 600 to 900 rounds, and with CDSL the average of dead nodes was 700 to 1700 rounds. When CM_Counting was run with static sink and RDM, the dead node average was 500 to 900 rounds, and with CDSL, the average of dead nodes was 1000 to 1500 rounds. When CM_Counting was run with static sink and SM, the dead node average was 400 to 1000 rounds, and with CDSL, the average was 600 to 1700 rounds. Protocol CM_Location with CDSL had a longer network lifetime than with static sink. When CM_Location was run with static sink and RWM, the dead node average was 500 to 900 rounds, and with CDSL, the average was 800 to 2000 rounds. When CM_Location was run with static sink and RDM, the dead node average was 500 to 900 rounds, and with CDSL, the average was 1200 to 1700 rounds. When it was run with static sink and SM, the dead node average was 200 to 1000 rounds, and with CDSL, the average was 600 to 2400 rounds.

A comparison is made of static sink with CDSL using different protocols, different mobility models, and different round duration times and the metrics: first node death round and the end round. We found that: LEACH with RDM and 10 seconds had the longest difference in network lifetime between static sink and CDSL, while LEACH with SM and 20 seconds had the shortest difference in network lifetime. CM with RDM and 10 seconds had the longest difference in network lifetime between static and CDSL; while SM and 20 seconds had the shortest difference in network lifetime. CM_Counting with RWM and 5 seconds had the longest difference in network lifetime between static sink and CDSL; while with RDM and 10 seconds, it had the shortest difference in network lifetime. CM_Location with SM and 15 seconds had the longest difference in network lifetime between static sink and CDSL; while with RWM and 15 seconds, it had the shortest difference in network lifetime.

We conclude that SM had the first dead node in all clustering protocols because sensors that are near to reference point (RP) have more priority for being chosen to be CHs than others, so they consume their power more than other sensors in the network. In general, RDM is better than RWM when the mobility models chosen. This is because in RDM the CM mechanism can predict the locations of CHs more accurate than when using RWM. In addition, the ACE_L protocol is better than other protocols because CHs have the best location and sensors can join the nearest CH to save more energy. While in ACE_C, the CHs are chosen in round robin fashion depending on their ID, and that has an effect on two aspects. The first effect is increasing cluster size where more sensors can join the same CH, and that leads to consuming its energy faster. The second effect is increasing the distance between the CH and its sensors, and that consumes sensor and CH energy faster. When the sink was mobile, the lifetime was prolonged in all cases by (14.21% to 53.09%). The lowest increment was obtained when CDSL is compared to static sink with LEACH, Simple Model, and 20 seconds as show in Fig.5, while the best increment was achieved with CM, RDM, and 10 Second as shown in Table (2).

Table 2: Difference in Network Lifetime Between CDSL and Static Sink with CM.

Round	5 Sec.	10 Sec.	15 Sec.	20 Sec.
Mobility Model	The first node death (Difference %)			
Random Walk	17.36	55.79	23.98	23.53
Random Direction	24.81	60.00	23.52	23.48
Simple	27.53	54.33	27.32	23.50
Mobility Model	The end round (Difference %)			
Random Walk	21.00	47.60	22.72	23.14
Random Direction	23.47	**53.09**	23.65	23.49
Simple	18.59	48.25	17.21	**16.78**

6. CONCLUSION AND FUTURE WORK

In this study, we proposed an advanced dynamic sink location method to reduce the power consumption in mobile-clustered WSN, and we aimed to increase the network lifetime in this network. To achieve this goal, we considered the concept of mobile sink in CDSL for moving it to the nearest location between all CHs in each round based on a centroid algorithm. Different clustering protocols were implemented in CDSL, all of which were run on different mobility models and run for different round duration times. In order to compare the performance of static sink to CDSL, the system lifetime metric was used under different scenarios. This metric includes the end round, the first dead node round number, and the average number of dead nodes. The simulation results indicated that CDSL increased the network lifetime by saving the cluster heads energy. The effect of modifying various distance measures or clustering algorithms on prolonging the network lifetime can be studied in the future.

ACKNOWLEDGMENTS

The authors would like to thank Prof. Howard C.-M. Liu for his help.

REFRENCES

[1] C.-M. Liu, C.-H. Lee, L.-C. Wang, Distributed clustering algorithms for data-gathering in wireless mobile sensor networks, Journal of parallel and distributed computing, 67 (2007) 1187-1200.

[2] V.M. Valsalan, Dynamic Deployment of Wireless Sensor Networks Using Enhanced Artificial Bee Colony Algorithm, International Journal, (2013).

[3] S. Bhuvaneshwari, P.S. Balamurugan, A bee-hive optimization approach to improve the network lifetime in wireless sensor networks, International Journal on Computer Science and Engineering (IJCSE), 5 (2013).

[4] W. Zheng, J. Shu, An Improved Artificial Bee Colony-Based Relocation Algorithm in Mobile Sensor Network, Information Engineering Letters, 2 (2012).

[5] F. Xia, X. Zhao, H. Liu, J. Li, X. Kong, An energy-efficient and load-balanced dynamic clustering protocol for ad-hoc sensor networks, in: Cyber Technology in Automation, Control, and Intelligent Systems (CYBER), 2012 IEEE International Conference on, IEEE, 2012, pp. 215-220.

[6] C. Ozturk, D. Karaboga, B. Gorkemli, Probabilistic dynamic deployment of wireless sensor networks by artificial bee colony algorithm, Sensors, 11 (2011) 6056-6065.

[7] D. Karaboga, S. Okdem, C. Ozturk, Cluster based wireless sensor network routing using artificial bee colony algorithm, Wireless Networks, 18 (2012) 847-860.

[8] A. da Silva Rego, J. Celestino, A. dos Santos, E.C. Cerqueira, A. Patel, M. Taghavi, BEE-C: a bio-inspired energy efficient cluster-based algorithm for data continuous dissemination in Wireless Sensor Networks, in: Networks (ICON), 2012 18th IEEE International Conference on, IEEE, 2012, pp. 405-410.

[9] R. Alageswaran, R. Usha, R. Gayathridevi, G. Kiruthika, Design and implementation of dynamic sink node placement using Particle Swarm Optimization for life time maximization of WSN applications, in: Advances in Engineering, Science and

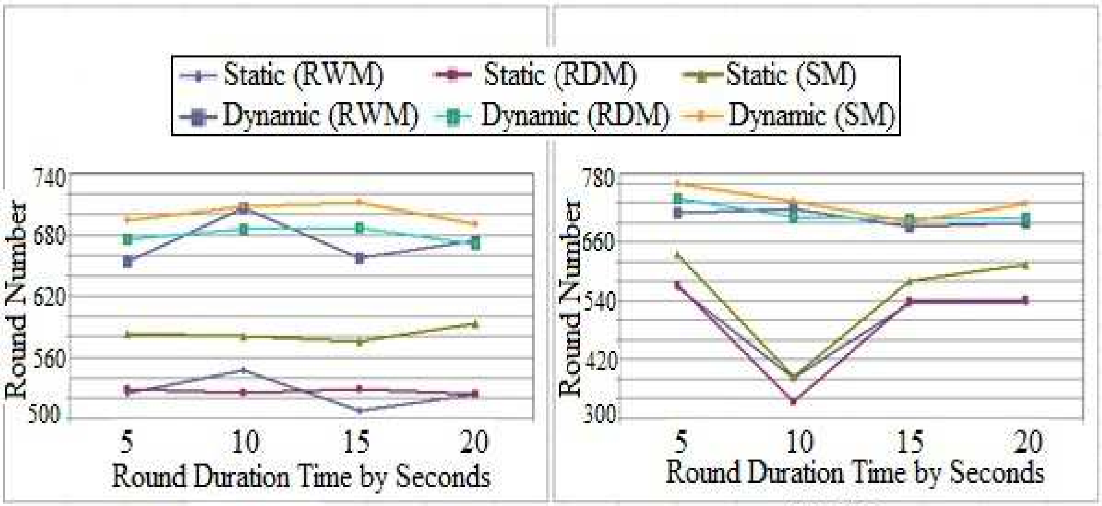
Management (ICAESM), 2012 International Conference on, IEEE, 2012, pp. 552-555.

[10] K. Oikonomou, S. Aissa, Dynamic sink assignment for efficient energy consumption in wireless sensor networks, in: Wireless Communications and Networking Conference (WCNC), 2012 IEEE, IEEE, 2012, pp. 1876-1881.

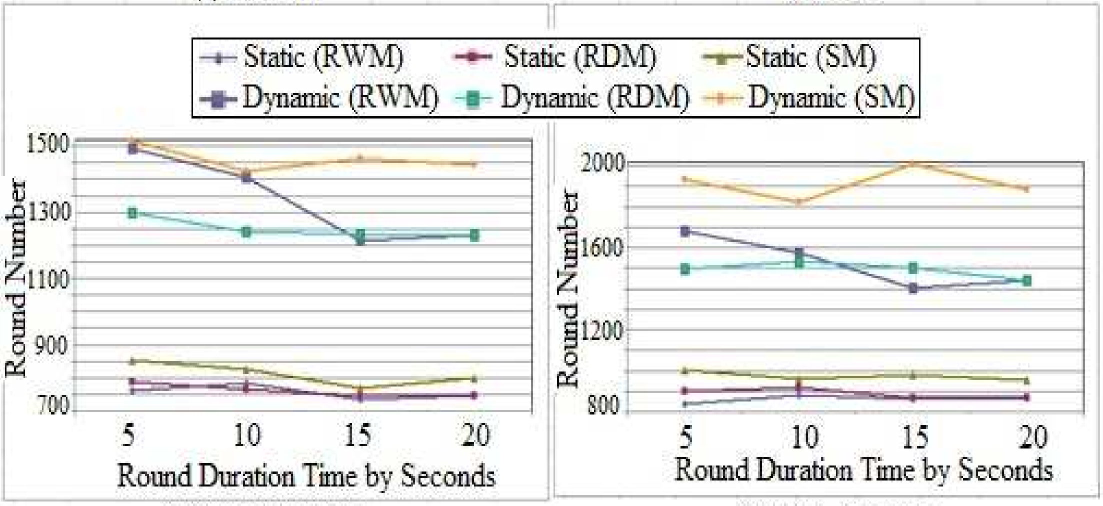
[11] T. Aseri, Comparison of routing protocols in wireless sensor network using mobile sink-A survey, in: Engineering and Computational Sciences (RAECS), 2014 Recent Advances in, IEEE, 2014, pp. 1-4.

[12] A.A. Abbasi, M. Younis, A survey on clustering algorithms for wireless sensor networks, Computer communications, 30 (2007) 2826-2841.

[13] W.R. Heinzelman, A. Chandrakasan, H. Balakrishnan, Energy-efficient communication protocol for wireless microsensor networks, in: System Sciences, 2000. Proceedings of the 33rd Annual Hawaii International Conference on, IEEE, 2000, pp. 10 pp. vol. 12.



(a)LEACH (b) CM



(c) CM_Counting (d) CM_Location

Figure 2: Clustering Protocols with Static Sink and CDSL Using (a) LEACH (b) CM (c) CM_Counting (d) CM_Location.

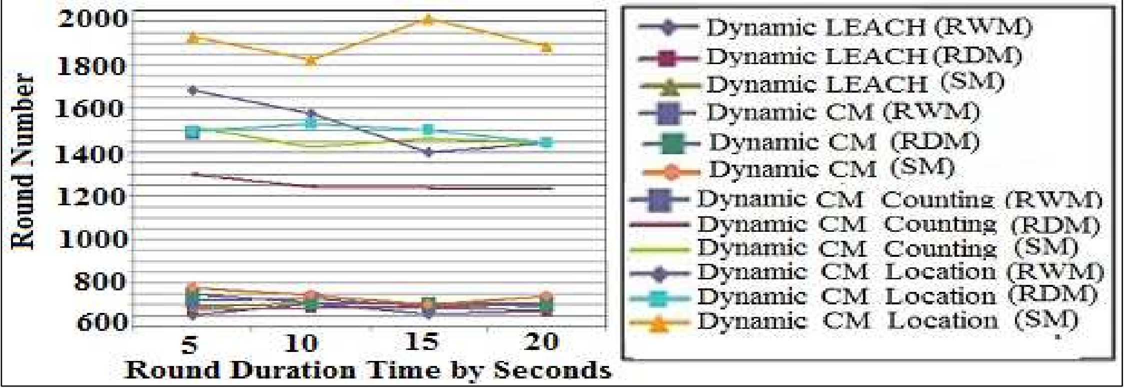


Figure 3: Clustering Protocols and Mobility Models with CDSL.

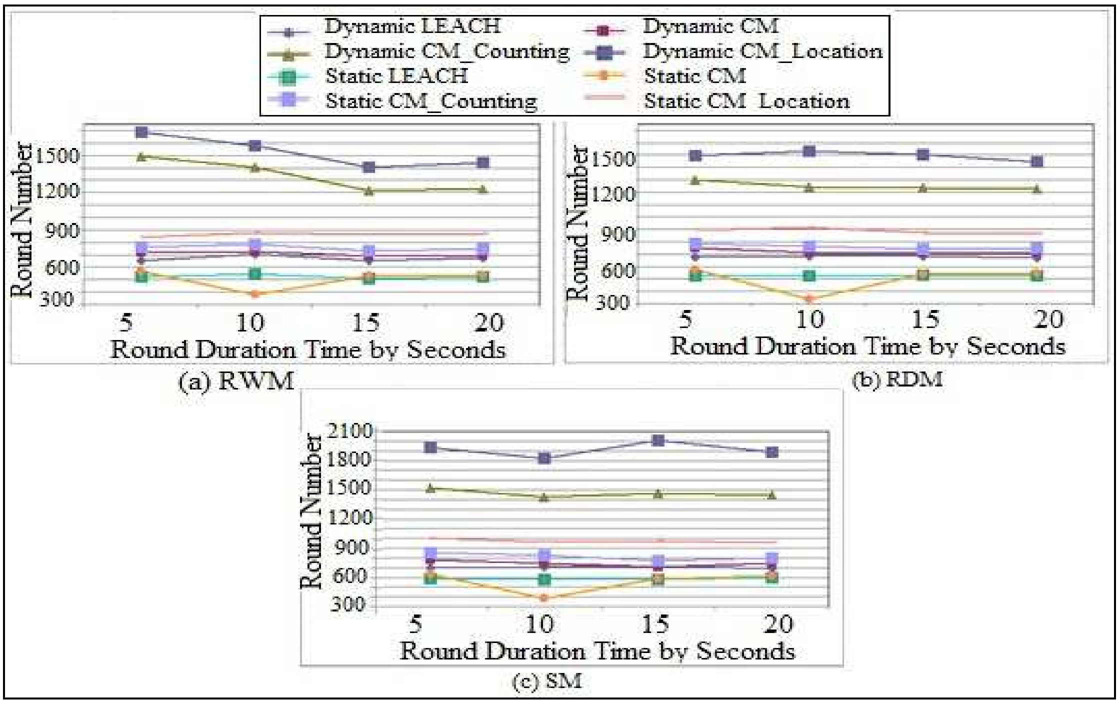


Figure 4: Clustering Protocols with Static and CDSL Using (a) RWM (b) RDM (c) SM.

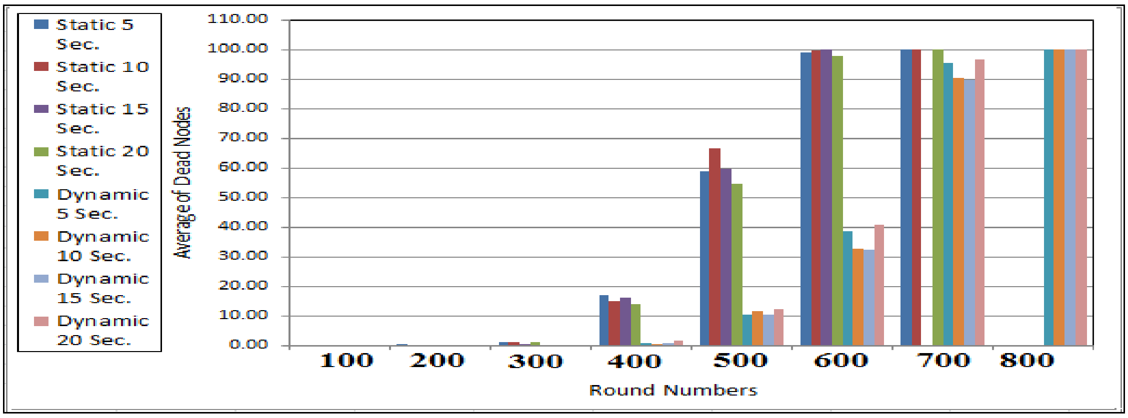


Figure 5: LEACH Average of Dead Nodes with SM When Using Static Sink and CDSL.