Wireless Personal Communications (2019) 105:1083–1109
https://doi.org/10.1007/s11277-019-06137-z

Energy and Network Balanced Distributed Clustering in Wireless Sensor Network

Srijit Chowdhury[1] · Chandan Giri[1]

Published online: 13 February 2019

Abstract

Balancing energy consumption of sensor nodes to extend the network lifetime is a major concern for the energy constrained wireless sensor network. Improper load balance and disproportionate energy consumption of sensor nodes during network activities such as transmitting and receiving of data cause energy hole problem and shorten the lifetime of the network. Many research works cited that clustering mechanism and the use of mobile sink can be effective to mitigate the energy hole problem and can improve the network lifetime. But using a mobile sink causes extra delay which is a major concern if the network is delay bound. In this work, we give deep insight into the problem of disproportionate energy consumption and aim to improve the network load balance and increase the network lifetime by applying efficient distributed clustering method with the help of a mobile sink. The proposed scheme named Energy Balanced Distributed Clustering Protocol (EBDCP) guarantees to transmit the sensed data to the base station within the tour deadline with the aid of a mobile sink. For this purpose, an efficient sojourn point determination algorithm has also been proposed. The simulation results prove that the proposed scheme performs significantly better than the existing works in terms of the energy distribution in the network, clustering overhead, residual energy of the network, number of alive nodes and network lifetime.

Keywords Wireless sensor network · Distributed clustering · Mobile sink · Energy efficient · Network lifetime

1 Introduction

Wireless sensor network (WSN) is an important scientific research and development area in recent times. It is widely utilized in studies of environmental phenomena, military surveillance, structural health monitoring, medical fields etc [23]. WSN consists

✉ Srijit Chowdhury
csrijitc@gmail.com; srijitc@it.iiests.ac.in

Chandan Giri
chandangiri@gmail.com; chandan@it.iiests.ac.in

[1] Department of Information Technology, Indian Institute of Engineering Science and Technology, Shibpur, Howrah 711103, India

of energy constrained sensor nodes which are deployed in hundreds or thousands over a large space. As the sensor nodes possess limited and non-rechargeable battery power, energy saving is the primary concern when designing a WSN.

Lifetime of a WSN is inversely related with the energy dissipation of the sensor nodes in the network. Researches show that clustering is one of the efficient data gathering techniques to increase the network lifetime of a WSN. But disproportionate energy consumption of the sensor nodes in the clustered network may lead to the early death of the network. This imbalanced energy consumption occurs in the clustered network due to (1) consumption of more energy by the cluster heads (CHs) when compared to a cluster member; and (2) for multi-hop inter-cluster communication, the CHs which are closed to the base station (BS) receive and relay more data as opposed to the CHs which are away from the BS. In single hop communication, the CHs which are at a significant distance from the BS expend more energy to send data directly to the BS as compared to the CHs near to it. In both of the above mentioned cases (1 and 2), some sensor nodes die prematurely and creates the hot-spot [19] problem leading to the disconnection of the network. To mitigate the causes of imbalanced energy consumption, the rotation of the CHs can be considered as a feasible option.

Also, the use of a mobile sink (MS) can successfully reduce the hot-spot problem. The MS follows a controlled or an uncontrolled trajectory either on demand basis or in a certain time interval, stops at some sojourn points (*SJP*s) and collects data from the CHs before returning back to the starting point. The problem gets more complicated if the network has a tour deadline by which the collected data must reach the BS. In that case, the *SJP*s and the path of the MS should be optimized with respect to the tour deadline. A group of centralized and distributed clustering algorithms have been proposed by the researchers to balance the energy load and to increase the network lifetime. But still there is scope to improve the performance of all these proposed schemes. The main problem of the centralized clustering scheme is the high overhead in each round. This is because the sensor nodes have to send some control messages to the BS in each round regarding the parameter values which are required to choose the CHs. Also the sensor nodes receive control messages which are broadcasted by the CHs. The problem of high overhead can be overcome by using distributed CH selection methods. But these approaches also face certain issues. Time synchronization between a BS and the sensor nodes is a major concern in distributed approaches. Time synchronization is needed to conduct the clustering process within a certain time period and also to complete each data gathering round within a specific time interval.

Bearing these problems in mind, this work aims to provide their solutions. The main contributions of this proposed scheme are as follows:

1. Balancing the energy consumption of the sensor nodes and minimising the energy consumption of the network to prolong the network lifetime.
2. Selecting optimal *SJP*s and path of the MS to ensure that the tour time of the MS does not exceed the tour deadline.
3. Connectivity of the network throughout the network lifetime.
4. Perfect time synchronisation between the BS and all the sensor nodes to ensure that a specific time interval is allocated for each round.

The paper is thus organised as follows: Sect. 2 discusses the related studies; In Sect. 3, network characteristics and problem definition has been described. The proposed scheme is discussed in detail in Sect. 4. The experimental results of the proposed scheme and the performance analysis are shown in Sect. 5. Finally, Sect. 6 concludes the work.

2 Related Studies

In WSN, a group of clustering methods have been proposed for energy efficient data collection and saving the network lifetime. But, these approaches have some limitations and thus failed to achieve the desired level of efficiency. In this section we discuss a few of such relevant techniques.

Low-Energy Adaptive Clustering Hierarchy (LEACH) protocol [7] is acknowledged as the first distributed clustering method. In this approach the cluster heads are selected randomly and probabilistically in a distributed fashion with the guarantee that all nodes are selected at least once within *N*/*k* rounds, where *N* denotes the total number of nodes in the network and *k* is the expected number of CHs in a round. Due to the absence of energy factor in selection of the CHs, LEACH becomes inefficient in terms of energy consumption and network lifetime. Later on some variants of LEACH have been proposed in [1, 4, 18, 20] to improve the performance of LEACH protocol.

In [9], the authors presented a dynamic cluster head selection method which tries to eliminate the overlapping of network coverage due to the random selection of CHs. Simulated results in [9] show improved performance in terms of energy consumption and network lifetime than LEACH [7] and some other existing clustering methods.

Equally Distributed Cluster Heads (DECH) technique is presented in [17] which is an improvement over some conventional clustering protocols like LEACH [7], Threshold sensitive Energy Efficient sensor Network protocol (TEEN) [12], Distributed Energy-Efficient Clustering scheme for heterogeneous wireless sensor networks (DEEC) [14] and Balanced Clustering Algorithm with Distributed Self-Organization (DSBCA) for Wireless Sensor Networks [11] in terms of energy consumption and network lifetime. DECH tries to balance the distribution of CHs in the network so that the energy consumption could be minimised. At first DECH selects CHs with the help of one of the conventional clustering protocols. These conventional clustering processes select CHs in a random manner and thus some CHs may be placed close to each other which degrades the performance of the network. DECH finds closely placed CHs and replace those CHs by new CHs through a re-selection process where a node having minimum average link cost is selected as new CH. The main problem of this approach is that the re-selection process increases the clustering overhead.

A recently proposed clustering process named Energy Efficient Probabilistic Clustering Technique (EEPCT) [21] also tries to improve the performance of LEACH protocol in terms of energy consumption and network lifetime. Both of a centralised and a distributed algorithm have been proposed in EEPCT. It selects CHs on the basis of closeness of nodes. It modifies the CH selection threshold of the LEACH protocol for improving the performance.

In [3], Sleep-awake Energy Efficient Distributed (SEED) clustering algorithm is proposed which divides the network field into some energy regions. The CHs of the high energy region communicate with the base station directly and thus consume extra energy in communication when compared to the CHs in low energy region. SEED implements awake and sleep mechanism to save energy. Optimal Clustering in Circular Networks (OCCN) [5] replaces one hop communication between CHs and the base station by multi-hop communication. OCCN finds optimal number of clusters and tries to partition the network with equal-size clusters.

In all of these approaches the following two instances may arise which cause disproportionate network load and unbalanced energy consumption:

- A sensor node may not found any CH within its low power transmission range. According to the above mentioned algorithms including LEACH protocol the node has to increase its power to reach its nearest CH and thus the node will consume extra energy in data communication.
- According to the above mentioned protocols a node chooses its nearest CH to join as a member node. In this process, it may happen that a CH may not be selected by any other nodes and thus may become a lone CH node. In this case the lone CH forms no cluster and has to send its own data individually to the BS. This causes uneven load distribution in the network.

Researches [10, 15] show that multi-level and/or multi-path routing based clustering methods can be adopted to reduce the communication distance between the sensor nodes and the base station which helps in reducing energy consumption and increase in the network lifetime.

All the above mentioned clustering techniques consider a static WSN structure where the whole network follows a converge-cast traffic pattern [6]. This traffic pattern directs data from all the source nodes towards a specific set of sensor nodes (for example, CHs in the clustered network) and thus may result in hot-spot problem.

Many existing data gathering schemes [2, 25] use MS to eliminate the hot-spot problem [6]. Tree-cluster-based data-gathering algorithm (TCBDGA) [26] uses a mobile sink to mitigate the hot-spot problem. The MS stops at some RPs (rendezvous points) and SRPs (subrendezvous points) to collect data. The work targets to balance the network load and increasing the network lifetime.

One of the main problems of using the MS is the delay occurred in data collection due to the low speed of the MS. So, finding the optimal trajectory of the MS especially in a delay sensitive network is a prime challenge to the researchers. In [16], a rendezvous point (RP) based data collection strategy with energy efficient path selection of the mobile sink has been proposed. The weighted rendezvous planning (WRP) algorithm proposed in [16] tries to find out optimal RPs in a delay bound WSN on the basis of hop distance of a node from the tour and the number of neighbour nodes of that node. In each round, the WRP algorithm runs a TSP solver [16] for several times to include a RP in the mobile tour. This increases the run time complexity of the algorithm and thus it is not a realistic approach.

3 Network Characteristics and Problem Definition

Figure 1 describes the cluster based network topology for the proposed scheme. Let us consider a wireless sensor network which comprises of N sensor nodes. The nodes are distributed over a $L \times L$ square field. Sensor nodes periodically generate data packets and these packets are delivered to the BS. In every periodic round, sensor nodes have to form clusters to rotate the CHs so that the network load can be evenly distributed over the network and the hot-spot problem [22] can be avoided. This helps in balancing the energy

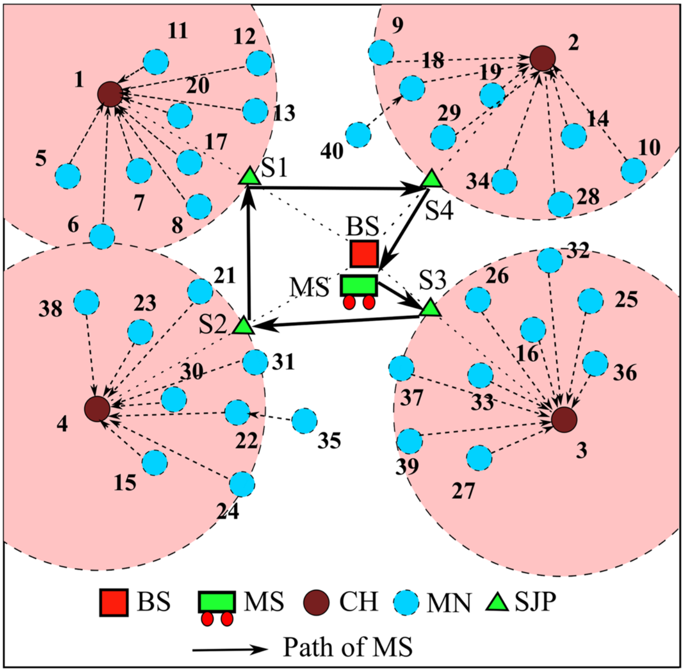


Fig. 1 Topology of the network

consumption of the network and increasing the network lifetime. A sensor node may have any of the three types of node status—ordinary node (ON), cluster head (CH) and member node (MN). Before cluster formation in every round, all the alive nodes have the ON status. During the clustering process some nodes of ON status become cluster heads and change their status to CH (See nodes 1, 2, 3 and 4 in Fig. 1). Each of the rest of the nodes of ON status joins a neighbour CH node (a CH node which belongs to its transmission range) and changes its node status to MN (See dotted circle nodes in Fig. 1). If a node has no neighbour CH node, then it joins a MN near it and forms a multi-hop path to reach a CH (See nodes 35 and 40 in Fig. 1).

An MS is employed to collect data packets from the CHs. In each round, the MS starts its journey from the BS and stops at *SJP*s near the CHs for the purpose of acquiring data from the CHs. After completion of the data collection process, the MS returns and delivers all collected data to the BS. The MS has a tour deadline. The *SJP*s and the tour path of the MS are to be selected in such a way that the MS can be able to complete its tour within the tour deadline. In Fig. 1, the *SJP*s (*S*1, *S*2, *S*3 and *S*4) are shown by triangles and the tour path of the MS is denoted by solid arrow lines.

To formulate the clusters selection process and finding the optimal sojourn points to meet the tour deadline, the following assumptions are considered in this work:

3.1 Assumptions

- Homogeneous and static sensor nodes are randomly deployed in the field.
- Each sensor node sends a single data packet in each round.

- At the time of deployment, all sensor nodes have the same initial energy (I_e).
- Sensor nodes set low power level for transmission. But, a sensor node may tune its power level to the maximum to communicate with another sensor node.
- The MS has enough energy to perform all of its tasks throughout the network lifetime.
- The static BS has enough power to communicate with all nodes. It finds the sojourn points and the optimal tour path of the MS.
- The static BS is positioned at the center of the square field, but it may be placed at any place even outside of the sensing field.
- The BS has the knowledge of locations of all the sensor nodes. By using any of the standard location finding techniques [8], the locations of sensor nodes can be determined.
- Similar to [16], when calculating delay, we do not consider the communication time between the MS and the sensor nodes, the transmission, queuing and propagation delays as because these are negligible as compared with the traversing time of the MS.
- Cluster member nodes send highly correlated data and thus those data can be fused at the associated CH.

3.2 Description of Packets

In the network, a group of control packets and data packets are exchanged among the BS, the MS and the sensor nodes. Brief descriptions of such packets are depicted in Table 1.

The energy consumption model for the proposed WSN is described next.

3.3 Energy Consumption Model

For simplicity and fair comparison with other existing works, the well known first order radio model [7] has been used in our work. According to [7], the amount of energy dissipated by the transmitting circuit to transmit k bits message to the distance d is:

$$\begin{aligned} E_{TX}(k,d) &= k \times E_{elec} + k \times E_{fs} \times d^2 \quad \textit{if } d < d_0 \\ &= k \times E_{elec} + k \times E_{mp} \times d^4 \quad \textit{if } d \geq d_0 \end{aligned} \tag{1}$$

where E_{elec} is the dissipated energy of the electronic circuit, $d_0 = \sqrt{E_{fs}/E_{mp}}$ is the threshold distance, E_{fs} is the dissipated energy of the amplifier circuit in free space model and E_{mp} is the dissipated energy of the amplifier circuit in multi path fading model.

The amount of energy dissipated by the receive circuit to receive k bits message is:

$$E_{RX}(k) = k \times E_{elec} \tag{2}$$

For sensing of k bits data, the energy dissipated by the sensing circuit is:

$$E_s(k) = k \times E_{sense} \tag{3}$$

where E_{sense} is the energy dissipation of the sensing circuit for sensing one bit of data.

To aggregate k bits data, the energy dissipation is calculated as:

$$E_{Agg}(k) = k \times E_{DA} \tag{4}$$

where E_{DA} is the per bit energy consumption in data aggregation.

Table 1 Packet description

Packet name	Packet description	Transmission mode	Content
CP1	Start message which BS broadcasts at the beginning of each round	Broadcast	p, w_1, w_2, w_3, α
CP2	CH advertisement message which is transmitted by a CH	Broadcast	*CH_id*
CP3	Joining message which is transmitted by MN to a CH	Unicast	*ON_id*, *CH_id*, number of hops
CP4	Scheduling message which is transmitted by a CH to its MNs	Broadcast	*CH_id*, time slots of all MNs
CP5	Lone CH declaration message which is transmitted by a lone CH	Broadcast	*CH_id*
CP6	Joining invitation message which is transmitted by an ON to a lone CH	Unicast	*ON_id*, *Lone_CH_id*
CP7	Joining message which is transmitted by a lone CH to its NH node	Unicast	*MN_id*, *parent_id*, *Lone_CH_id*, hop
CP8	Upload data request message which is transmitted by the MS to nearby CHs	Broadcast	*MS_id*
DP1	Data message which is transmitted by MN to the CH	Unicast	*MN_id*, *CH_id*, data
DP2	Aggregated data message which is transmitted by a CH to the MS	Unicast	*CH_id*, *MS_id*, *aggregated_data*

A CH node dissipates its energy for sensing data, receiving data and control packets, aggregating received data packets and transmitting data and control packets. Equation (5) measures the dissipated energy of a CH i.

$$\begin{aligned} E_{CH}(i) = E_s(k_d) + E_{RX}(k_d) \times m_i + E_{Agg}(k_d) \times (m_i + 1) \\ + E_{TX}(k_d, d(i, NH)) + n_{TX} \times E_{TX}(k_c, d(i,j)) + n_{RX} \times E_{RX}(k_c) \end{aligned} \tag{5}$$

Here k_d is the size of a data packet, k_c is the size of a control packet, m_i is the number of member nodes associated with the CH i, $d(i, NH)$ is the distance between the CH i and its next hop node, n_{TX} is the number of control packets transmitted by the CH i and n_{RX} is the number of control packets received by the CH i.

A non-CH node dissipates its energy for sensing data, transmitting data and control packets and receiving data and control packets. Equation (6) measures the dissipated energy of a non-CH node j associated with cluster head i.

$$\begin{aligned} E_{mem}(j) = E_s(k_d) + m_{RX} \times E_{RX}(k_d) + (m_{RX} + 1) \times E_{TX}(k_d, d(j, i)) \\ + m_{TX} \times E_{TX}(k_c, d(j, i)) + n_{RX} \times E_{RX}(k_c) \end{aligned} \tag{6}$$

Here $d(j, i)$ is the distance between the non-CH node j and CH node i, m_{RX}, m_{TX} and n_{RX} are respectively the number of received data packets, number of transmitted control packets and number of received control packets.

From the Eqs. (5) and (6), the energy consumption of the network in a round (E_{Round}) can be calculated as:

$$E_{Round} = \sum_{i=1}^{Q} E_{CH}(i) + \sum_{i=1}^{Q} \sum_{j=1}^{m_i} E_{mem}(i,j) \tag{7}$$

where Q is the number of CHs.

4 Proposed Scheme

The proposed Energy Balanced Distributed Clustering Protocol (EBDCP) tries to distribute the energy consumption load evenly over the network. Even distribution of energy consumption among the sensor nodes aids to mitigate the hot-spot problem as discussed in Sect. 1. EBDCP also tries to select the optimal set of CHs so that the overall energy consumption of the network can be reduced. The proposed scheme is divided into two main phases: cluster building phase and the data transmission phase. To abate the hot-spot problem, the rotation of CHs in every round is needed and thus the cluster building phase is executed in each round. The proposed scheme is described with the help of a flowchart (Fig. 2).

At the initial stage of network formation the BS broadcasts a start message ($CP1$) containing values of some network parameters. The broadcast is made with sufficient power to reach all the sensor nodes in the network. Initially all the sensor nodes set their eligibility status (*E_Status*) of being a CH to *true*. All the sensor nodes including the BS set the count of number of rounds (*Round*) to zero. On receiving the $CP1$ message, each sensor node sets their node status (*Node_Status*) to *ON* and participates into the cluster head selection process. For the selection of CHs and the association of member nodes, a distributed clustering protocol named Distributed Cluster Selection Protocol (*DCSP*) has been proposed which is described in Algorithm 1 in Sect. 4.1.

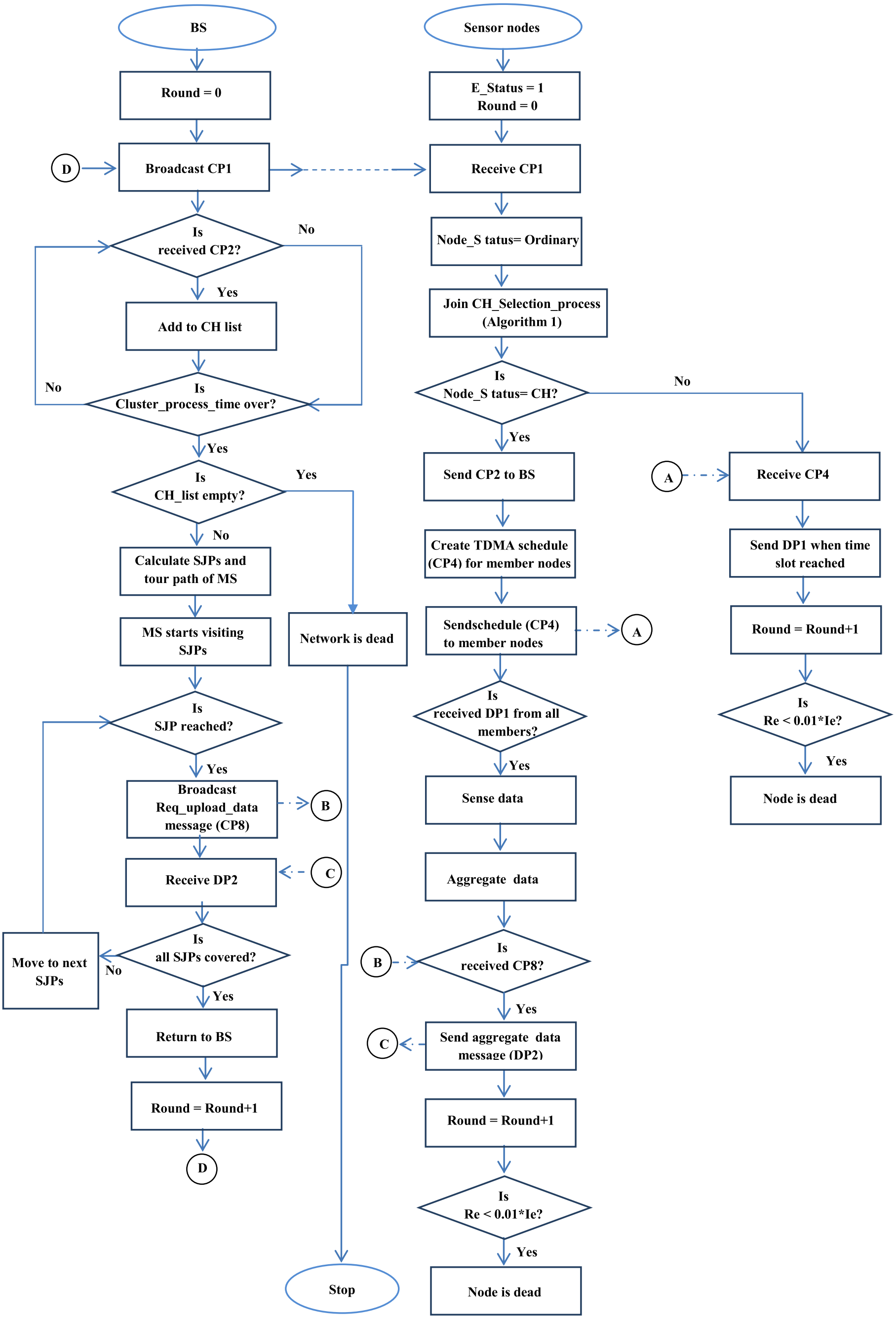


Fig. 2 Flowchart of the proposed scheme

4.1 Cluster Building Phase

In the proposed scheme, *DCSP* (Algorithm 1) is used to select the CHs and the cluster members in each round in such a way that the total energy consumption is minimised in each round while maintaining the network connection throughout the lifetime. Each alive sensor node having *E_Status* = *true* runs *DCSP* to calculate their Cluster Selection Factor (*CSF*) value. *CSF* is the chance factor of a node to be a CH. The range of *CSF* is [0,1]. The higher the *CSF* value, the greater the chance to be a CH. To calculate the *CSF* value, three strong determination factors are considered, they being residual energy(R_e) of the node, the LEACH threshold (threshold used in LEACH protocol [7]) and distance (d_B) between a node and the BS. The calculation of *CSF* is given in Eq. (8).

$$CSF = w_1 f_1 + w_2 f_2 + w_3 f_3 \tag{8}$$

where

$$f_1 = \frac{R_e}{I_e} \tag{9}$$

$$f_2' = \frac{p}{1 - p(Round \ mod \ 1/p)} \tag{10}$$

$$f_2 = f_2' + \alpha(1 - f_2') \tag{11}$$

$$f_3 = \frac{d_B}{\sqrt{2} \times L} \tag{12}$$

and

$$w_1 + w_2 + w_3 = 1 \tag{13}$$

The value of f_2 is calculated in two steps. At first, the LEACH threshold [7](f_2' in Eq. 10) is calculated and then it is incremented by using a constant factor (α) which is supplied by the BS along with $CP1$ message. In f_2, p denotes the probability of a node to be a CH. Thus, p measures the desired percentage of CHs in the network (see Section 2.1 of [21]). In the second step, each node generates a random number (r) and if $r < f_2$ then $f_2 = r$, otherwise $f_2 = 0$ (See line 15 to 21 in Algorithm 1). The factor f_3 depicts that the distant nodes from the BS are preferable for the selection of CHs. This is because the nodes away from the BS deplete their energy quickly due to either the direct transmission of data to the BS or the use of large multi-hop relay to reach a CH or the BS. Thus, the selection of distant nodes as CHs helps in reducing the average transmission distance of the nodes which are distant

Table 2 List of timers

Timer	Description
T_{advCH}	CH advertisement timer
$T_{CH-process}$	Timer to indicate the ending of CH selection process
$T_{membership}$	Timer to indicate the ending of joining process of member nodes
$T_{membership\_extra\_time}$	Timer for extra time allocation to handle the joining of lone CHs

from the BS. This also aids the *SJP*s (See Sect. 4.3) to be evenly spread all over the region instead of concentrating the *SJP*s around the BS.

In the cluster building process, some timers are used to measure the time instances of the occurrence and ending of various events (e.g. selection of CHs, joining of member nodes etc.). The list of timers and their descriptions are given in Table 2. The time unit of all the timers is measured in seconds.

After calculating the *CSF* value, each node sets the advertisement timer $T_{advCH}(t_1)$, where $t_1 = \frac{1}{CSF}$seconds. The advertisement timer T_{advCH} is used by the nodes to advertise themselves as a CH. The timer T_{advCH} takes an expiry time t_1 as input. After time t_1 the timer T_{advCH} expires and the corresponding node declares itself as a CH. According to the proposed scheme, the node which has the highest *CSF* value will be selected as a CH first. For this purpose, each node sets the expiry time (t_1) of the advertisement timer T_{advCH} to the reciprocal of the *CSF* value. On expiry of T_{advCH} timer, a node broadcasts the *CP*2 message to advertise itself as a CH. The advertising node then changes the *Node_Status* to *CH*. It also sets *E_Status* = *false*. This helps the *DCSP* protocol to select a new set of CHs in the next round. If a node *j* receives a *CP*2 message, then it stops its T_{advCH} timer to withdraw itself from the CH selection process. This helps in selecting evenly distributed CHs over the network. The node *j* then marks the corresponding CH as its next hop node (*NH*) and changes its *Node_Status* to member node (*MN*). A node may get more than one *CP*2 messages. In that case, it selects the CH node on the basis of strongest received signal strength (RSSI) value among all the received *CP*2 messages (See line 35 in algorithm 1).

To ensure that all the nodes start their timers at the same time, the nodes wait for $(\delta_{advCH} - (T_c - T_s)))$ seconds, where δ_{advCH} and T_s respectively are the waiting time and the broadcasting time of a round as declared by the BS through *CP*1 message and T_c is the current time measured at the node. Each node also starts the timer $T_{CH-process}(t_2)$ along with the starting of T_{advCH} timer to calculate the end time of the CH selection process, where $t_2 = \frac{1}{CSF_{min}} + C$. Here, CSF_{min} is the minimum value of *CSF* for which a node has the chance to be selected as a CH, and *C* is a constant. In this work, $CSF_{min} = 0.1$ is considered. This is reasonable as in this work the number of nodes has been considered to be 100 where out of these 100 nodes, approximately 10 nodes ($p = 0.1$) can be selected as CHs in each round. Thus, assuming that a single node takes 1 s to declare itself as a CH, the total time required for the declaration of all 10 CHs will be 10 s. Thus, the node which declares itself as the 10th CH requires its CH advertisement timer $T_{advCH}(t_1)$ to be expired within 10 s i.e. $t_1 = \frac{1}{CSF} = 10$ which yields $CSF_{min} = 0.1$. The CH selection process ends when $T_{CH-process}$ expires. Now, the joining process of the member nodes to the CHs starts. For this purpose, each node starts $T_{membership}$ timer. Each *MN* sends the joining message (*CP*3) to its selected CH and the CH adds *MN*s to its member list (See lines 49–56 in Algorithm 1). After the end of the aforementioned process, some CHs may exist with no member nodes (lone CH). According to the proposed scheme, these lone CHs change their status to *MN* and select a nearby member node as its *NH* node (See lines 57–75 in algorithm 1). For this purpose, all nodes start $T_{membership\_extra\_time}$ timer and before it expires, a lone CH broadcasts a *CP*5 message. A member node (member to other CH), on hearing the *CP*5 message, replies by sending a *CP*6 message to the lone CH. The lone CH selects the node as its *NH* node from which it receives *CP*6 message earlier than the rest (if it receives more than one *CP*6 messages). The lone CH (which has now changed to a member node) sends the joining message (*CP*7) to the selected *NH* node which then informs its CH about this new member node by sending a *CP*3 message.

Algorithm 1: DCSP

```
Input  : $p, w_1, w_2, w_3, \alpha, Round, I_e, R_e, \delta_{advCH}, E\_Status, CSF_{min}, C, d_B, L$
Output: $Node\_Status$ ($CH$ or $MN$),
        $NH$ (Next hop node),
begin
    $NH$=id($i$); // $i$ is the current node
    $RSSI$=0;
    if $Round\ mod\ 1/p = 0$ then
        $E\_Status = true$;
    end
    if $E\_Status = false$ then
        $CSF$=0.01; //$CSF$ is set to a very small number
    end
    else
        $f_1 = \frac{R_e}{I_e}$;
        $f_2' = \frac{p}{1-p(Round\ mod\ 1/p)}$;
        $f_2 = f_2' + \alpha(1 - f_2')$;
        $f_3 = \frac{d_B}{\sqrt{2} \times L}$;
        $r_1 = rand(0, 1)$;
        if $r_1 \leq f_2$ then
            $f_2 = r_1$;
        end
        else
            $f_2 = 0$;
        end
        $CSF = w_1 \times f_1 + w_2 \times f_2 + w_3 \times f_3$;
    end
    $t_1 = \frac{1}{CSF}$;
    wait($\delta_{advCH} - (T_c - T_s)$);
    $t_2 = \frac{1}{CSF_{min}} + C$;
    Start timer $T_{advCH}(t_1)$;
    Start timer $T_{CH-process}(t_2)$;
    if $T_{CH-process}$ is running then
        if $T_{advCH}$ expired then
            Broadcast $CP2$;
            $Node\_Status$ = CH;
            $E\_Status = 0$;
        end
        if $Node\_Status = ON$ and received $CP2$ from node $j$ then
            if $RSSI(j) \geq RSSI$ then
                $NH = id(j)$;
                $RSSI = RSSI(j)$;
                $Node\_Status = MN$;
                if $T_{advCH}$ is running then
                    Stop $T_{advCH}$;
                end
            end
        end
    end
    else
        Start timer $T_{membership}$;
    end
    if $T_{membership}$ is running then
        if $Node\_Status = CH$ and $CP3$ from node $j$ received then
            Add node $j$ to the $member_list$;
        end
        if $Node\_Status = MN$ then
            Send $CP3$ to $NH$;
        end
    end
    if $T_{membership}$ expired then
        Start $T_{membership\_extra\_time}$;
    end
    if $T_{membership\_extra\_time}$ is running then
        if $Node\_Status = CH$ and $member\_list$ is empty then
            Broadcast $CP5$;
            if $CP6$ is received then
                Send $CP7$;
            end
        end
        if $Node\_Status = MN$ then
            if $CP5$ is received then
                Send $CP6$;
            end
            if $CP7$ is received then
                Send $CP3$;
            end
        end
    end
end
```

4.1.1 Time Synchronisation Between the BS and all the Sensor Nodes

In the proposed work, sensor nodes send data packets to the BS with the aid of the MS in each round. Thus, the calculation of the start time and the end time of a round must be the same for the BS and all the other sensor nodes. Also, in a WSN, the sensor nodes have a different local clock time. Therefore, the time synchronisation between the local time of the BS and that of all the other sensor nodes is necessary. For this purpose, Flooding Time Synchronisation Protocol (FTSP) [13] can be used which accurately synchronises (per-hop synchronisation error is within 1 μs range [13]) the local clock time of sensor nodes with respect to a reference clock (in our case, the clock of the BS has been considered as the reference clock). Further to ensure that each round commences at the same time in all the sensor nodes, the following approach has been adopted.

Before starting each round the BS sends the round start message ($CP1$) which consists of δ_{advCH} and T_s, where δ_{advCH} denotes how much time (in seconds) a sensor node shall have to wait before starting the CH selection process and T_s denotes the broadcasting time stamp of $CP1$ message. If a sensor node receives $CP1$ message from the BS at time T_c, then the sensor node calculates the propagation time of the $CP1$ message which is $(T_c - T_s)$. Thus, the sensor node starts T_{advCH} and $T_{CH-process}$ timers (for CH selection process) after waiting $(\delta_{advCH} - (T_c - T_s))$ seconds. This ensures that all the sensor nodes start a round at the same time although they receive the round-start message ($CP1$) at different local time.

Lemma 1 *The message complexity of EBDCP in cluster building phase is* $O(N)$, *where N is the number of nodes in the network.*

Proof Let us suppose in a round, there are N number of sensor nodes in the network. Among those nodes, number of CHs is Q and number of lone CHs is ϕ, where $\phi << N$. Hence, the total number of MNs is $(N - Q - \phi)$. Each CH sends a $CP2$ message to the BS and broadcasts a $CP2$ message and a $CP4$ message to its member nodes. Thus, the total number of $CP2$ and $CP4$ messages transmitted by all CHs is $(3 \times Q)$. The BS sends a $CP1$ message to all sensor nodes. The total number of such messages received by all the sensor nodes is N. Each MN transmits one $CP3$ message to the associated CH and thus the total number of $CP3$ messages is $(N - Q - \phi)$. To associate a lone CH node as a member node to another CH, the control messages $CP5$,$CP6$, $CP7$ and $CP3$ are exchanged. So, the total number of these messages will be $(\phi + \phi \times N + \phi + \phi = 3\phi + \phi N)$. Hence, the total number overhead control messages in cluster building phase is $(3 \times Q + N + (N - Q - \phi) + 3\phi + \phi N = 2N + 2Q + 2\phi + \phi N)$.

Thus the message complexity of EBDCP in cluster building phase is $O(2N + 2Q + 2\phi + \phi N) \approx O(N)$ since $\phi << N$. □

4.2 Data Transmission Phase

After the execution of *DCSP*, the node status of all the alive nodes becomes either CH or MN. The nodes having *Node_Status* = *CH* send $CP2$ message to the BS. Then CHs create time schedules for their MNs and send $CP4$ message to those MNs. An MN gets activated according to its scheduled time slot and sends sensed data by transmitting $CP4$ message to its corresponding CH. In rest of the time the MNs go to the idle state to save energy. A CH after getting all the data messages ($CP4$) from their respective MNs, senses its own data and then aggregates all the data including its own. When a CH receives $CP8$ message (upload data request message) from the MS, it transfers the aggregated data through $DP2$ message. The MS stops at each SJP on its travelling path and collects aggregated data from all the CHs before returning to the starting point. When the MS delivers all the collected data to the BS, a network round is considered to be completed. This process is continued till the network becomes dead. The network is considered dead if in a round the BS receives no $CP2$ message. In the next Sect. 4.3, the calculation of determining SJPs is described.

4.3 Selection of *SJP*s

After getting the information of all the CHs, the BS starts finding the set of SJPs such that the total tour time of the MS will not exceed the delay constraint (τ_d). For this purpose the SJP Determination Algorithm (See Algorithm 2) is proposed. The position of the BS is considered to be the first SJP and it is added to the SJP_list. A sorted list of all CHs is prepared in descending order on the basis of d_B. Then for each CH in the sorted list, the BS calculates the co-ordinates of the SJP (SP_x, SP_y) by using Section formula (See lines 13 and 14). It is assumed that the SJP divides the line segment between the BS and the CH in the ratio $m : n$, where $m = d_B - Tr$ and $n = Tr$. Then the new SJP is added to the SJP_list. The cost of the tour (in terms of travelling time of the MS) is calculated. After calculating the cost, if it is seen that the cost exceeds τ_d then the new SJP is removed from the SJP_list. The process continues until all the CHs in the sorted list are examined. Finding the minimum cost of the tour of the MS maps to a travelling salesman problem (TSP) [24] if we consider the CHs as the cities and the MS as a salesman. As TSP is a NP-hard problem, a heuristic method is required to find out a near optimal cost of the tour. For this purpose, a simple heuristic called Nearest Neighbour (NN) algorithm is applied as in [24]. In NN algorithm, the salesman starts visiting from a city and then moves to the next city among all the unvisited cities which has the shortest route (edge) from the current city. This continues until the salesman visits all the cities before returning to the starting point. Figure 1 shows the tour path of the MS based on the NN algorithm. The MS (salesman) starts from the BS (starting city) and then moves to $S3$ which is the nearest unvisited SJP (city) among all the unvisited SJPs ($S1$, $S2$, $S3$, $S4$) from the BS (current city). Following the same process the MS then visits $S2$, $S1$ and $S4$ respectively and finally returns to the BS.

```
Algorithm 2: $SJP$ Determination

Input  : $CH\_list$ with locations
Output: $SJP\_list$,
        $tour\_path$ of the MS
begin
    $Sort\_descend(CH\_list)$;
    Add $(X_{BS}, Y_{BS})$ to $SJP\_list$;
    $x1 = X_{BS}$;
    $y1 = Y_{BS}$;
    for $i$=1 to $L$ do
        $x2 = CH\_list(i).x$;
        $y2 = CH\_list(i).y$;
        $d_B = \sqrt{(x1 - x2)^2 + (y1 - y2)^2}$;
        if $d_B > T_r$ then
            $m = d_B - T_r$;
            $n = Tr$;
            $SP_x = (mx2 + nx1)/(m + n)$;
            $SP_y = (my2 + ny1)/(m + n)$;
            Add $(SP_x, SP_y)$ to $SPJ\_list$;
            (cost, tour_path) = run TSP($SPJ\_list$);
            if $cost > \tau_d$ then
                Remove $(SP_x, SP_y)$ from $SPJ\_list$;
            end
        end
    end
end
```

5 Performance Evaluation

For the purpose of performance evaluation of the proposed scheme, we have simulated the proposed scheme and other existing algorithms (EEPCT [21], DECH [17] and LEACH [7]) using Matlab 2015a and compared the simulated results in Sects. 5.2 and 5.3. Performance evaluation is done with respect to the energy distribution in the network, clustering overhead, residual energy of the network, number of alive nodes and network lifetime. Balanced energy distribution in the network and less clustering overhead lead to high residual energy of the network, increased number of alive nodes and increase in the network lifetime. Again, balanced energy distribution occurs when the clustering method selects optimal or near optimal set of CHs. Selection of optimal CHs leads to less energy consumption of CHs. Variance of the energy consumption of nodes is a good parameter to measure the distribution of energy over the network. For the fair comparison, we consider two scenarios. In Scenario 1, to measure the efficiency of the clustering process of the proposed scheme, we assume a WSN which is delay tolerant and no MS is employed to collect the data. The CHs have to send aggregated data directly to the BS which is located at the center of the square field. In Scenario 2, we consider a delay sensitive network where the MS, in every round, starts from the BS and travels towards some *SJP*s to collect data from the CHs. The MS should return to the BS within a specified tour deadline. The metrics for the simulations are summarised in Table 3. For the fair comparison with the existing works, the value of p is set to 0.1 as in [17, 21].

The value of the tour deadline (τ_d) is set to 100 s for the simulation purpose, though in a real scenario the tour deadline could be set to any time duration as per the application's requirement.

Table 3 Parameters used in simulation mode

E_{elec}	50 nJ/bit
E_{fs}	100 pJ/bit/m^2
E_{mp}	0.0013 pJ/bit/m^4
E_{sense}	0.2 µJ/bit
E_{DA}	5 nJ/bit/signal
Initial energy(I_e)	2 J
Data packet size (k_d)	2000 bits
Control packet size (k_c)	100 bits
Number of nodes (N)	100
Position of BS (X, Y)	(50 m, 50 m)
Sensing area ($L \times L$)	$100\,m \times 100\,m$
Tour deadline (τ_d)	100 s
Speed of the MS (v_{MS})	1 m/s
$w_1 + w_2 + w_3$	0.4 + 0.3 + 0.3
Probability of a node to be a CH (p)	0.1

5.1 Determining Optimal Values of Weights w_1, w_2 and w_3

w_1, w_2 and w_3 are the weight coefficients for the factors f_1, f_2 and f_3 respectively to calculate CSF (Eq. 8). Here, the weight coefficient determines how much importance is given to the respective factor for selecting a cluster head (CH). To determine the optimal combination of the values of the weights, we consider the first node death (in rounds) of the WSN consisting of 100 nodes. Figure 3 shows the results obtained for all possible combinations of w_1, w_2 and w_3. Each line shows the 1st node deaths (in rounds) for all possible combinations of w_2 and w_3 against each fixed value of w_1. It is seen from Fig. 3 that at each line the number of rounds increases to a certain point initially and then it decreases. We call all

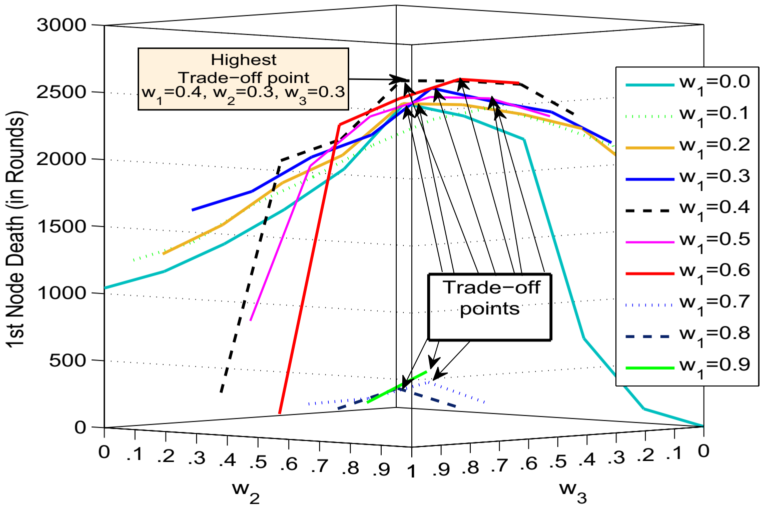


Fig. 3 1st Node death for varying combinations of w_1, w_2 and w_3

such points as the trade-off points. From all such trade-off points it can be seen that the highest number of rounds exists at the point where $w_1 = 0.4$, $w_2 = 0.3$ and $w_3 = 0.3$. (See Highest Trade-off point in Fig. 3). Thus, in this work, the aforementioned values of the weight coefficients are used.

The experimental results and performance analysis are described in the next section.

5.2 Experimental results for Scenario 1

Figure 4 shows the residual energy of nodes in different remaining network energy status (75%, 50% and 25% respectively of the total network energy) for the different algorithms. As can be seen from the figures, LEACH and DECH have comparatively uneven energy distribution among the nodes when compared with the rest of the protocols as they do not consider any energy parameter while choosing the CHs. The distributed version of the EEPCT algorithm considers residual energy of nodes as the final selection criteria of the CHs and thus shows better energy distribution than LEACH and DECH. EBDCP showcases the most balanced energy distribution in all the three cases of remaining energy status of the network. This is due to the efficient choice of parameters and the use of the CH selection function *CSF* as described in Sect. 4.1. This proves that the CH selection protocol (*DCSP*) used in the proposed scheme selects the best set of CHs in each round as compared with the other protocols. This is the key factor for the consumption of less energy by the CHs which is illustrated in Fig. 5a. This also helps in maintaining a steady and balanced ratio of the energy consumption of CHs ($\Sigma_{i=1}^{Q} E_{CH}$) and the network energy consumption of a round (E_{Round}). This is depicted in Fig. 5b. For further investigation of the even distribution of energy over the network, the variances of energy consumed by the CHs ($Var(E_{CH})$) and the MNs ($Var(E_{mem})$) respectively from the mean energy consumption of the nodes (μ_{E_n}) are depicted in Fig. 6. The variances of CHs and MNs are calculated respectively by the Eqs. 14 and 15.

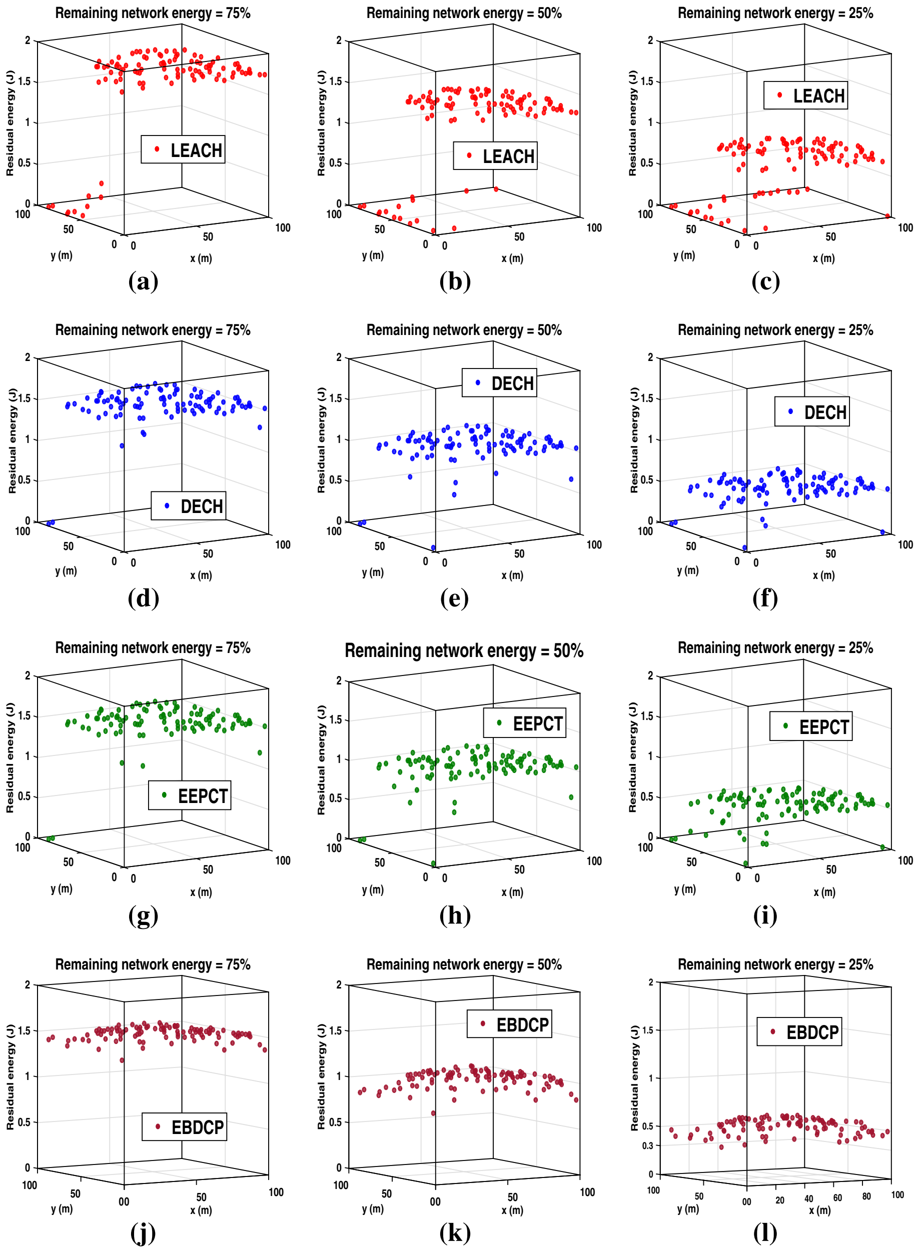
$$Var(E_{CH}) = \Sigma_{i=1}^{Q} (E_{CH}(i) - \mu_{E_n})^2 \ / \ Q \tag{14}$$

$$Var(E_{mem}) = \Sigma_{j=1}^{n_A - Q} (E_{mem}(i) - \mu_{E_n})^2 \ / \ n_m \tag{15}$$

where Q is the number of CHs, n_A is the number of alive nodes and $n_m (= n_A - Q)$ is the number of member nodes.

It can be observed from Fig. 6a, b that EBDCP yields comparatively low and steady variances ($Var(E_{CH})$ and $Var(E_{mem})$) as compared to the variances of other protocols. These results further favour EBDCP scheme for well balanced energy dissipation over the network as compared with other protocols.

Figure 7a calculates the overhead energy consumption in clustering process of all the algorithms. Overheads are calculated by the number of control messages which are involved in the clustering process. Less overhead is desired for longer network lifetime. It is clearly seen from Fig. 7a that DECH consumes very high overhead energy than the rest of the protocols. The main factor of such high overhead is the implementation of re-selection of CHs which involves extra control packets in the clustering process. Figure 7b provides the ratio of overhead energy ($E_{Overhead}$) and the network energy consumption (E_{Round}) to show the percentage of overhead energy to the network energy consumption. As expected



(a) (b) (c)

(d) (e) (f)

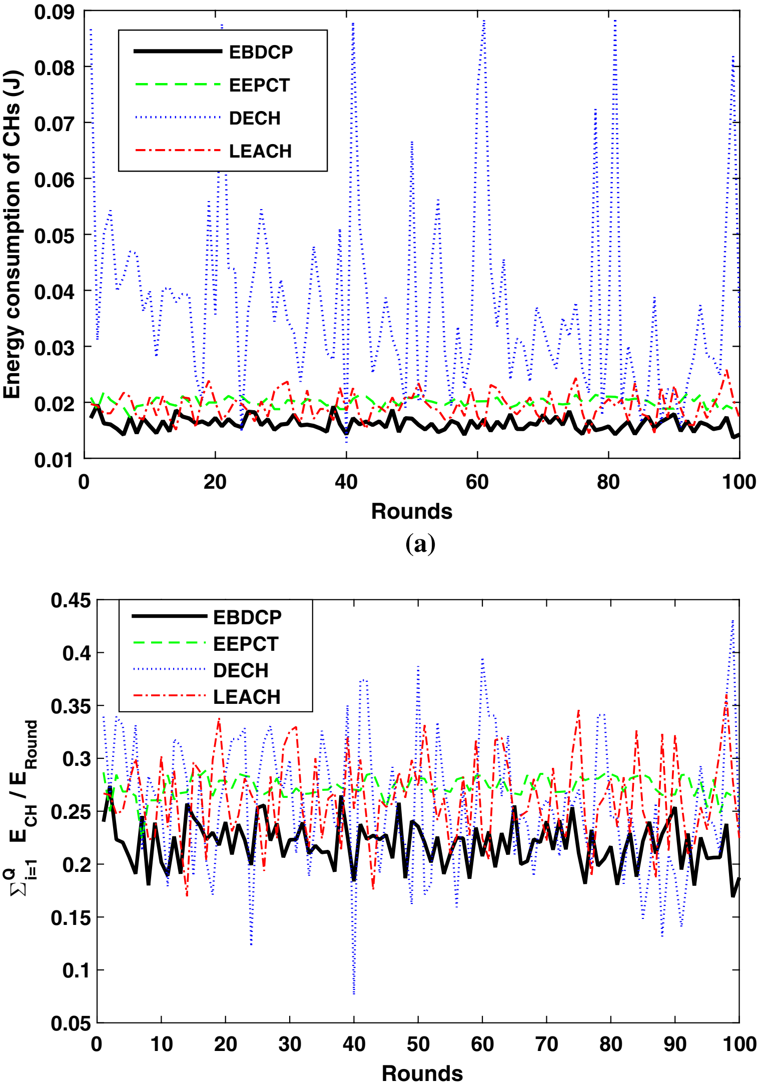
(g) (h) (i)

(j) (k) (l)

Fig. 4 Residual energy of nodes in different remaining network energy status

from the discussion, it is clearly visible from the figure that DECH yields high overhead percentage as compared with the other schemes. The rest of the schemes including the proposed EBDCP protocol maintain a reasonably low overhead percentage.

According to the results as discussed till now, it can be invariably concluded that EBDCP shows a balanced energy consumption and less clustering overhead as and



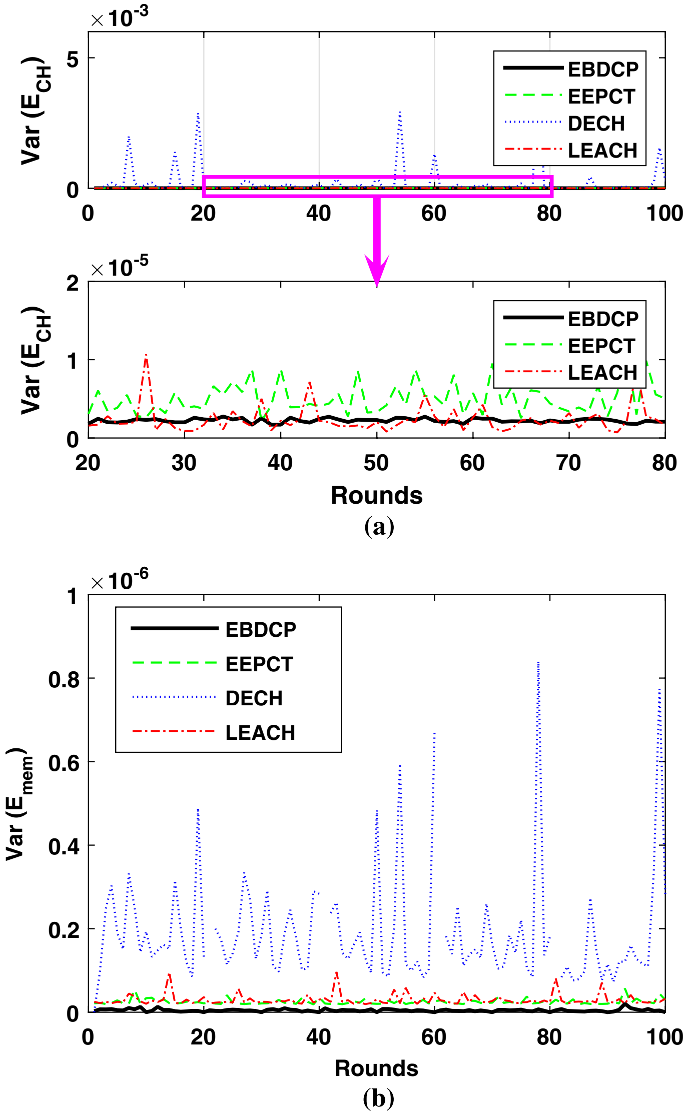
(a)

(b)

Fig. 5 **a** Energy consumption of CHs in rounds. **b** Ratio of CHs energy and network energy consumption in rounds

when compared to the other algorithms. Going by the consequences of these conclusions, Fig. 8 shows the residual energy of the network for different rounds till the last node is dead. As expected EBDCP shows better performance than the other existing algorithms.

The same conclusion and results are reflected in Fig. 9 as well. Fig. 9 shows that there are more number of alive nodes per round for EBDCP as compared to other existing algorithms. The stability period which is defined as the number of rounds till the



(a)

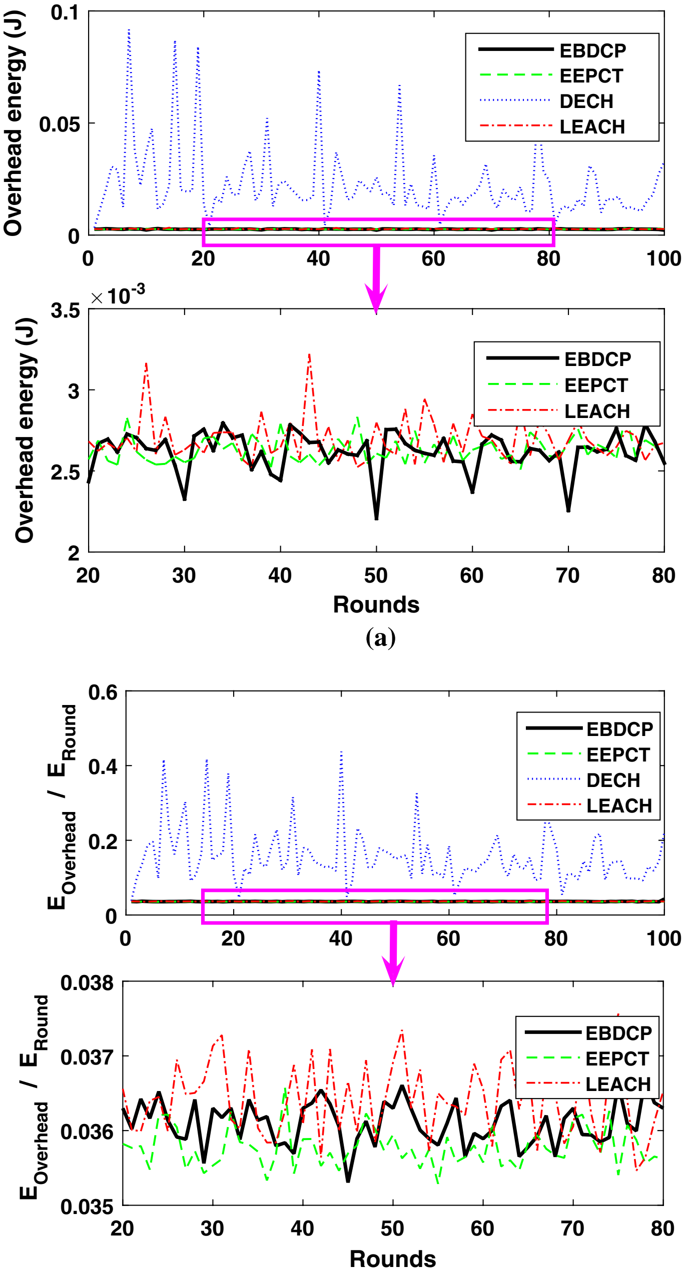
(b)

Fig. 6 Variance of energy consumptions for **a** CHs. **b** MNs

death of the first node is also higher in the proposed algorithm. To further verify Figs. 9, 10 shows the comparison for the different algorithms in terms of percentage of dead nodes.

5.3 Experimental Results for Scenario 2

In case of Scenario 1, as the CHs have to send their aggregated data to the BS directly thus the CHs which are away from the MS may have to increase their transmission power to the



(a)

(b)

Fig. 7 **a** Overhead energy consumption. **b** Ratio of overhead energy and network energy consumption in rounds

Fig. 8 Residual energy of the network in rounds

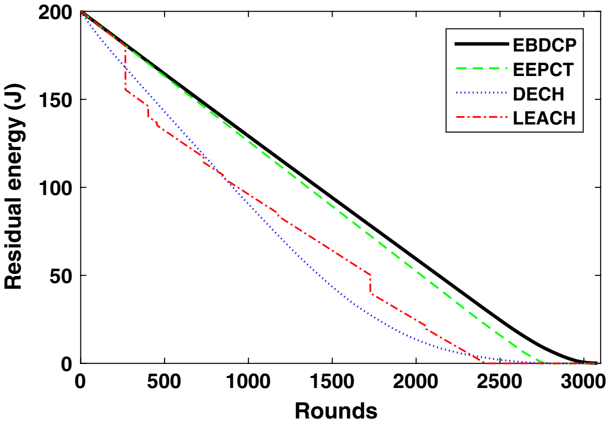


Fig. 9 Number of alive nodes in rounds

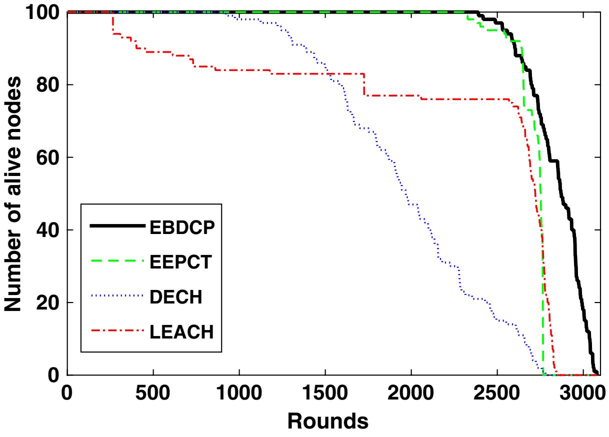
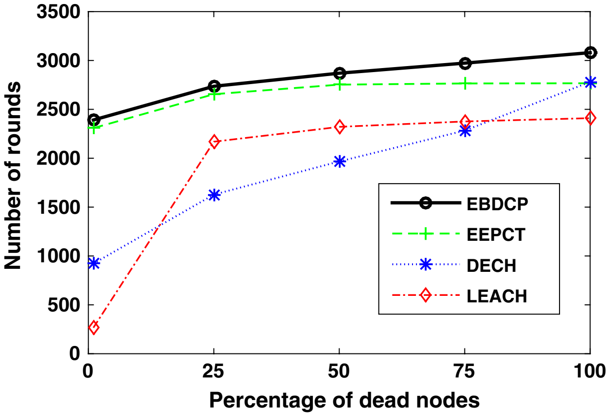


Fig. 10 Percentage of dead nodes versus rounds



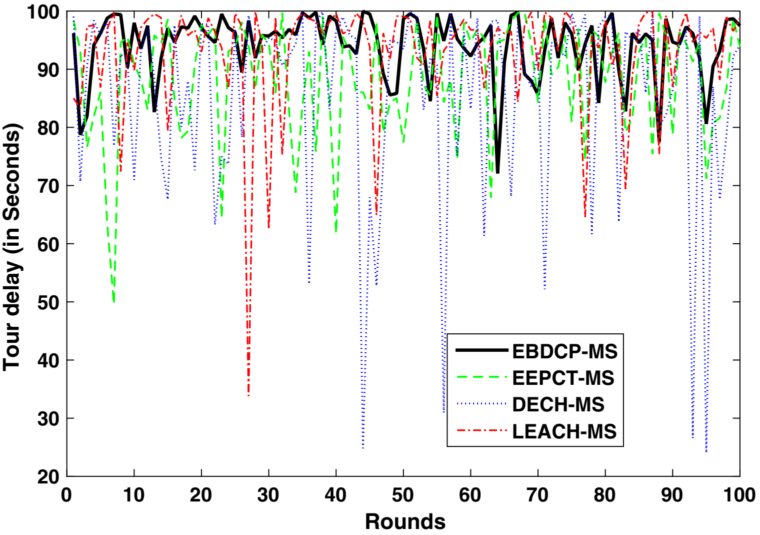


Fig. 11 Tour time of the MS in different rounds

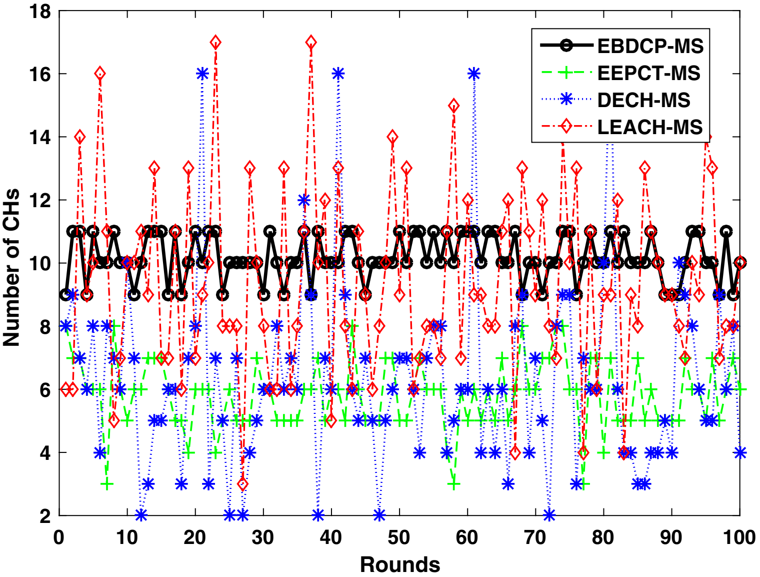


Fig. 12 Number of CHs in different rounds

maximum to reach the BS. Consequently this may cause unbalanced energy consumption over the network. To mitigate this problem an MS is employed in Scenario 2 where the MS travels some *SJP*s near the CHs. The CHs deliver their data to the MS when the MS comes within their proximity. In Scenario 2 it is also assumed that the low-speed MS should complete its journey within the tour deadline. For the purpose of fair comparison the *SJP*

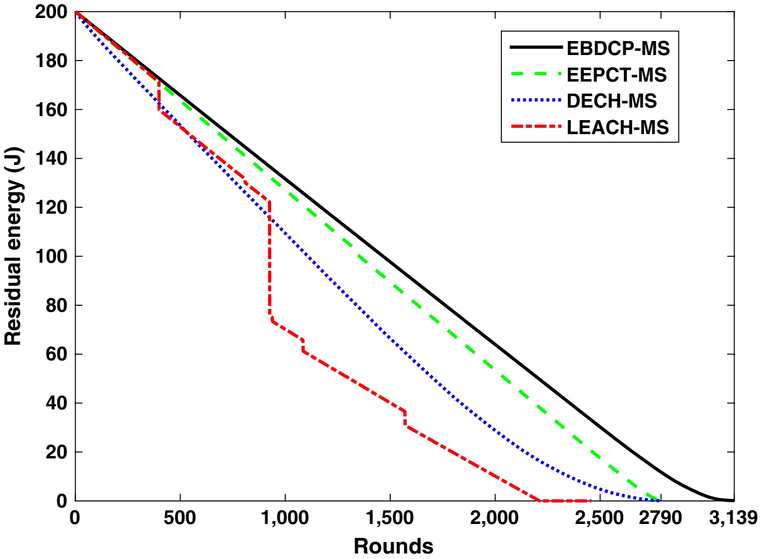


Fig. 13 Network residual energy in rounds

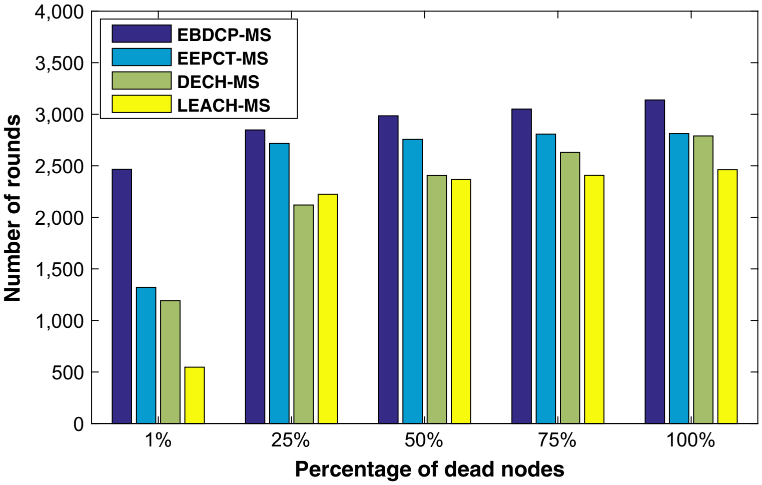


Fig. 14 Percentage of dead nodes

determination algorithm (Algorithm 2) is implemented to all the algorithms (EBDCP, EEPCT, DECH and LEACH) and they are named as EBDCP-MS, EEPCT-MS, DECH-MS and LEACH-MS respectively. In Figs. 11, 12, 13 and 14, *N* is set to 100 and *L* is set to 100 m. Fig. 11 shows the tour time of the MS in different rounds. The tour deadline (τ_d) is set to 100 s. From the figure it is seen that the proposed EBDCP-MS has steady and relatively higher tour time than the others. This is due to the fact that EBDCP-MS produces 9 to 11 number of CHs (See Fig. 12) all along the rounds which is very close to the optimal

Table 4 Percentage of improvements in the network lifetime

Percentage of dead nodes (%)	Improvement over LEACH-MS	Improvement over DECH-MS	Improvement over EEPCT-MS
1	350%	107%	86.6%
25	28%	34.3%	4.8%
50	26.1%	24.1%	8.3%
75	26.7%	16%	8.65%
100	27.5%	12.5%	11.63%

number of CHs (according to [7] the optimal number of CHs (k_{opt}) is $N \times p = 100 \times 0.1 = 10$). The maximum deviation of the number of CHs from the k_{opt} is 1. In case of the other existing algorithms, the range of the number of CHs varies from 3 to 17 (Fig. 12). Thus, the maximum deviation is 7 which is much higher as compared to EBDCP-MS. Maintaining a steady near optimal set of CHs all through the network rounds helps EBDCP-MS to achieve higher network residual energy in various rounds as compared to the other algorithms. This is depicted in Fig. 13. The high network residual energy leads to high network lifetime. The network lifetime is measured in terms of number of rounds upto which the network is functioning and connected. So, the number of rounds for any percentage of nodes' death (1% to 100%) can be considered as the network lifetime.

In Fig. 14, network lifetime is shown in rounds for different percentage of dead nodes. It is clearly seen from the figure that in all the cases EBDCP-MS showcases better performance by achieving higher number of rounds when compared to the existing methods. The percentage of improvements achieved by EBDCP-MS over existing methods are shown in Table 4.

6 Conclusion

In this work, a novel distributed clustering method (EBDCP) has been proposed which balances the distribution of energy consumption over the whole network. Balancing of energy distribution helps in reducing the network energy consumption and increases the network lifetime. A time synchronized optimal cluster selection method named *DCSP* is used in EBDCP which selects near optimal set of CHs in each network round. The network round time is calculated by the sensor nodes by using an efficient time synchronization process. To overcome the hot-spot problem, the CHs are rotated in every round and an MS is also employed in the network. To ensure the applicability of the proposed scheme to a delay sensitive WSN, an optimal sojourn point selection algorithm named *SJP* Determination algorithm has been proposed which restricts the tour time of the MS within the tour deadline. Simulation results show that the proposed scheme outperforms the existing methods in terms of the balance of energy distribution, reduction in energy consumption, low clustering overhead and increase in the network lifetime.

References

1. Abasıkeleş-Turgut, İ., & Hafif, O. G. (2016). Nodic: A novel distributed clustering routing protocol in WSNS by using a time-sharing approach for CH election. *Wireless Networks*, *22*(3), 1023–1034.

2. Abo-Zahhad, M., Ahmed, S. M., Sabor, N., & Sasaki, S. (2015). Mobile sink-based adaptive immune energy-efficient clustering protocol for improving the lifetime and stability period of wireless sensor networks. *IEEE Sensors Journal*, *15*(8), 4576–4586.
3. Ahmed, G., Zou, J., Fareed, M. M. S., & Zeeshan, M. (2016). Sleep-awake energy efficient distributed clustering algorithm for wireless sensor networks. *Computers and Electrical Engineering*, *56*(Supplement C), 385–398.
4. Al-Ma'aqbeh, F., Banimelhem, O., Taqieddin, E., Awad, F., & Mowafi, M. (2012). Fuzzy logic based energy efficient adaptive clustering protocol. In *Proceedings of the 3rd international conference on information and communication systems, ICICS '12* (pp. 21:1–21:5). New York, NY: ACM.
5. Arghavani, M., Esmaeili, M., Esmaeili, M., Mohseni, F., & Arghavani, A. (2017). Optimal energy aware clustering in circular wireless sensor networks. *Ad Hoc Networks*, *65*(Supplement C), 91–98.
6. Gu, Y., Ren, F., Ji, Y., & Li, J. (2016). The evolution of sink mobility management in wireless sensor networks: A survey. *IEEE Communications Surveys Tutorials*, *18*(1), 507–524. Firstquarter.
7. Heinzelman, W. B., Chandrakasan, A. P., & Balakrishnan, H. (2002). An application-specific protocol architecture for wireless microsensor networks. *IEEE Transactions on Wireless Communications*, *1*(4), 660–670.
8. Jamalabdollahi, M., & Zekavat, S. A. R. (2015). Joint neighbor discovery and time of arrival estimation in wireless sensor networks via ofdma. *IEEE Sensors Journal*, *15*(10), 5821–5833.
9. Jia, D., Zhu, H., Zou, S., & Hu, P. (2016). Dynamic cluster head selection method for wireless sensor network. *IEEE Sensors Journal*, *16*(8), 2746–2754.
10. Laouid, A., Dahmani, A., Bounceur, A., Euler, R., Lalem, F., & Tari, A. (2017). A distributed multipath routing algorithm to balance energy consumption in wireless sensor networks. *Ad Hoc Networks*, *64*(Supplement C), 53–64.
11. Liao, Y., Qi, H., & Li, W. (2013). Load-balanced clustering algorithm with distributed self-organization for wireless sensor networks. *IEEE Sensors Journal*, *13*(5), 1498–1506.
12. Manjeshwar, A., & Agrawal, D. P. (2001) Teen: A routing protocol for enhanced efficiency in wireless sensor networks. In *Proceedings 15th international parallel and distributed processing symposium. IPDPS 2001* (pp. 2009–2015).
13. Maróti, M., Kusy, B., Simon, G., & Lédeczi, Á. (2004). The flooding time synchronization protocol. In *Proceedings of the 2nd international conference on embedded networked sensor systems, SenSys '04* (pp. 39–49). ACM: New York, NY.
14. Qing, L., Zhu, Q., & Wang, M. (2006). Design of a distributed energy-efficient clustering algorithm for heterogeneous wireless sensor networks. *Computer Communications*, *29*(12), 2230–2237.
15. Sabet, M., & Naji, H. (2016). An energy efficient multi-level route-aware clustering algorithm for wireless sensor networks: A self-organized approach. *Computers and Electrical Engineering*, *56*(Supplement C), 399–417.
16. Salarian, H., Chin, K. W., & Naghdy, F. (2014). An energy-efficient mobile-sink path selection strategy for wireless sensor networks. *IEEE Transactions on Vehicular Technology*, *63*(5), 2407–2419.
17. Sundareswaran, P., Vardharajulu, K. N., & Rajesh, R. S. (2015). Dech: Equally distributed cluster heads technique for clustering protocols in wsns. *Wireless Personal Communications*, *84*(1), 137–151.
18. Tong, M., & Tang, M. (2010 Sept) Leach-b: An improved leach protocol for wireless sensor network. In *2010 6th international conference on wireless communications networking and mobile computing (WiCOM)* (pp. 1–4).
19. Wang, J., Cao, Y., Li, B., Kim, H., & Lee, S. (2017). Particle swarm optimization based clustering algorithm with mobile sink for WSNS. *Future Generation Computer Systems*, *76*(Supplement C), 452–457.
20. Wang, W., Du, F., & Xu, Q. (2009 Sept) An improvement of leach routing protocol based on trust for wireless sensor networks. In *2009 5th international conference on wireless communications, networking and mobile computing* (pp. 1–4).
21. Yadav, R. K., Gupta, D., & Lobiyal, D. K. (2017). Energy efficient probabilistic clustering technique for data aggregation in wireless sensor network. *Wireless Personal Communications*, *96*(3), 4099–4113.
22. Yan, J., Zhou, M., & Ding, Z. (2016). Recent advances in energy-efficient routing protocols for wireless sensor networks: A review. *IEEE Access*, *4*, 5673–5686.
23. Yick, J., Mukherjee, B., & Ghosal, D. (2008). Wireless sensor network survey. *Computer Networks*, *52*(12), 2292–2330.
24. Zhao, M., & Yang, Y. (2012). Bounded relay hop mobile data gathering in wireless sensor networks. *IEEE Transactions on Computers*, *61*(2), 265–277.
25. Zhao, M., Yang, Y., & Wang, C. (2015). Mobile data gathering with load balanced clustering and dual data uploading in wireless sensor networks. *IEEE Transactions on Mobile Computing*, *14*(4), 770–785.

26. Zhu, C., Wu, S., Han, G., Shu, L., & Wu, H. (2015). A tree-cluster-based data-gathering algorithm for industrial wsns with a mobile sink. *IEEE Access*, *3*, 381–396.

Srijit Chowdhury is currently working towards the Ph.D. degree in Information Technology at Indian Institute of Engineering Science and Technology, Shibpur, India. His current research is focused on efficient data gathering techniques in wireless sensor network. He received B.Sc degree from Calcutta University in 1995 and had received Master in Computer Application from Sikkim Manipal University, India in 2010 and subsequently Master of Technology in Information Technology from Bengal Engineering and Science University, Shibpur, India in 2013. He is a student member of IEEE and member of ISOC, India, Kolkata Chapter.

Chandan Giri has been at IIEST, Shibpur, India (formerly Bengal Engineering and Science University, Shibpur) since 2008 as Assistant Professor, Department of Information Technology. His current research is focused on testing and design for testability of integrated circuits (especially 3D and multicore chips) and wireless sensor network. He received B.Sc (H) degree in Physics from Calcutta University in 1997. Then he received his Bachelor of Technology (B.Tech) in Computer Science & Engineering from Calcutta University, India in 2000 and subsequently Masters of Engineering (M.E.) in Computer Science & Engineering with specialization in Image processing from Jadavpur University, Kolkata, India in 2002. He was awarded PhD degree form Department of Electronics and Electrical Communication Engineering of Indian Institute of Technology, Kharagpur in 2008. He served as programme committee member of International conferences ICIC 2009 and ICIC 2010. He also presented his research papers in several International Conferences and Journals. He is a member of IEEE and ACM.