Computers and Electrical Engineering 67 (2018) 96–113

Contents lists available at ScienceDirect

Computers and Electrical Engineering

journal homepage: www.elsevier.com/locate/compeleceng

Hybrid energy-efficient multi-path routing for wireless sensor networks☆

Mohit Sajwan[a,*], Devashish Gosain[b], Ajay K. Sharma[a]

[a] National Institute of Technology Delhi (NITD), New Delhi, India
[b] Indraprastha Institute of Information Technology Delhi (IIITD), New Delhi, India

ARTICLE INFO

Article history:
Received 27 July 2017
Revised 11 March 2018
Accepted 12 March 2018
Available online 23 March 2018

Keywords:
Energy efficiency
Routing
Clustering
Network lifetime
WSN

ABSTRACT

The effectiveness of a wireless sensor network relies on the underlying routing protocol. In this paper, we propose a novel algorithm which leverages both flat and hierarchical routing schemes for maximizing energy efficiency. It designates some desired number of nodes as cluster heads leading to cluster formation in the network. Inside clusters, nodes adopt multi-hop routing scheme to communicate with cluster head, which on reception of data packets from all cluster members, transmits the aggregated data along the precomputed path to the sink. Intra-cluster communication can happen in two modes *viz., philanthropist*—maximal residual energy neighbor node is selected, and *selfish*—nearest node is selected as next hop. Our approach refrain nodes from transmitting along long links, thus minimizing the energy consumption of the network. We simulated our algorithm against established protocols, and results indicate that it outperforms other protocols for network characteristics like energy minimization and scalability.

1. Introduction

A wireless sensor network (WSN) consists of tiny, low-powered sensors communicating with each other possibly through multihop wireless links and collaborating to accomplish a common task [1]. As a bridge between physical and virtual information worlds, WSN collects data from it surrounding environment and communicate it to the digital world, such as computers. To accomplish their tasks, WSNs should address two needs: (i) sensing in the target area and (ii) communication between the sensor nodes. Since they operate on limited power supplies for pervasive computing it becomes essential to keep them functional as long as possible. It has been already established that a sensor node expends very less energy in sensing in comparison to communication [2]. In literature, many routing protocols have been proposed and specifically tailored to minimize the energy consumption of sensor nodes. They can be broadly classified into flat and hierarchical algorithms.

In flat routing like [3,4], a node generally transmits its packets to neighboring nodes within its communication range. Whereas in hierarchical routing like LEACH (low energy adaptive clustering hierarchy) [5] and HEED (a hybrid energy-efficient distributed clustering) [6], a node transmits its data to its nearest cluster head (CH) which in turn sends it to the sink. Both the approaches have their own merits and demerits. The foundational principle of flat routing is cooperative

☆ Reviews processed and recommended for publication to the Editor-in-Chief by Associate Editor Dr. M. H. Rehmani.

* Corresponding author.
E-mail addresses: mohitsajwan@nitdelhi.ac.in (M. Sajwan), devashishg@iiitd.ac.in (D. Gosain), director@nitdelhi.ac.in (A.K. Sharma).

https://doi.org/10.1016/j.compeleceng.2018.03.018

multi-hop forwarding, but in doing so, a large volume of traffic is generated (in simplistic case a packet from each node is generated and forwarded to the sink) and it results in energy depletion of many nodes. Whereas, in hierarchical routing scheme, there are some designated cluster head nodes which are responsible for data aggregation from their cluster members and finally sending the aggregated information to the sink themselves. This conserves the energy of cluster members but puts a heavy toll on *CHs* [7]. Also, since all the sensor nodes are bound to latch themselves to some *CH*, they may do so by communicating out of normal radio range. This further results in poor quality of service and degraded performance. Our proposed algorithm hybrid energy-efficient multi-path routing protocol (*HEEMP*), attempts to provide a solution to the aforementioned problems. It primarily aims at creating clusters in the network and within each cluster, nodes transmit their sensed data to CH in a multi-hop manner. Later, CH aggregates the received data and transmits it to the sink along the pre-computed path.

HEEMP is a hybrid approach as it incorporates both hierarchical and flat routing schemes. Initially, sink designate CHs based on *node degree* and *residual energy* of the sensor nodes. Once the CHs are elected, sink broadcast CH advertisement message, which leads to different cluster formation. But inside each cluster, cluster members refrain from transmitting their sensed data directly to CHs, rather adopt co-operative forwarding with fellow cluster members to communicate with CH.

In order to establish the efficacy of HEEMP, we compared it with four well-established routing algorithms *viz.*, LEACH [5], PEGASIS [8], GSTEB [9] and TBC [10] and observed that HEEMP outperformed all of them for various performance metrics like *network lifetime, scalability and residual energy etc.*

The major contributions of this research are:

- We propose a novel routing protocol HEEMP (hierarchical in design), which aims at *increasing the network lifetime* of the network.
- Important tasks like CH selection, route construction from sink to CH are carried out by sink itself; thereby reducing the load on the sensor nodes.
- Under its operation no node communicates more than d_0 distance (long link—discussed in Section 3.3) as it results in tremendous energy conservation.
- We also propose M-TBC (Modified tree based clustering) protocol, which is an improvement on existing TBC (Tree based clustering)[10] routing protocol.
- We compared HEEMP with other established routing protocols under various simulation settings (varying area, number of sensor nodes and sink locations) and found that it outperformed all. We observed 920% performance gain against LEACH[5], 290% against PEGASIS[8], 38% against GSTEB[9] and 761% against M-TBC.

The rest of the paper is organized as follows. In Section 2, we present our related research and in Section 3, we formulate our system model and the data aggregation schemes. The HEEMP protocol is described in detail in Section 4. Next, in Section 5, we address some shortcomings of TBC routing scheme and propose Modified-TBC (M-TBC) which is an improvement over TBC. In Section 6, we describe comparative analysis and simulation results of HEEMP protocol with other established protocols. Later, we describe how M-TBC is better protocol than TBC in Section 7 and in Section 8, we discuss our limitations and future work. Finally, in Section 9 we present our concluding remarks.

2. Related research

In WSN, the main task of each sensor is to transmit its sensed data periodically to the Base station (BS) or sink[1]). The simplistic approach to achieve this is Direct Transmission, which allows nodes to directly communicate with BS [5]. However, it leads to uneven energy depletion among the sensor nodes. Therefore, the nodes which are placed far from the BS, would drain out faster in comparison to the nodes which are placed closer to the BS. The high disparity in energy consumption of nodes, ultimately shortens the overall network lifetime, violating the basic criteria of WSN (*viz.*, energy conservation of sensor nodes). To overcome such issues, Low Energy Adaptive Clustering Hierarchy (LEACH) protocol [5] was proposed, where network is divided into various clusters, while network operation is divided into various rounds. Each round is further divided into two phases: the setup and the steady state phase. In the setup phase, each node computes a threshold value followed by a random number. If this random number has a lesser value than the threshold, it will elect itself as a cluster head. Each node latches itself to the nearest cluster head, leading to a cluster formation. During steady state phase, cluster head aggregates the data packets received from its cluster members and by adopting single hop communication it send data packets to BS. LEACH, improved the network lifetime, eight times more than the direct transmission.

In [11], Centralized Energy Efficient Distance (CEED) routing protocol was proposed (an enhancement of LEACH), which aimed at improving the cluster head selection and cluster formation. In CEED, CH selection is based on residual energy and distance of each node from the sink. In cluster formation also, each node chooses its CH on the basis of residual energy and distance parameter. It then constructs a chain between the cluster heads for transmitting data packets to the sink (in a multihop manner).

Later, authors in [12] propose a new CH selection scheme where each node competes to become a CH. The nodes having the high residual energy are given preference over low residual energy nodes. Once the CHs are elected and clusters are

[1] In this article, sink and BS are interchangeably used.

formed, again the CH having the high residual energy and in close proximity to the sink are elected as parent CHs for all other remaining CHs. It is a two-step process; first clusters are formed and later among all CHs their parent CHs are elected. The parent CH collects data of all the CHs (*viz.*, the entire networks sensed data) and transmits it to the sink using multi-hop forwarding.

All routing protocols, in the family of hierarchical routing schemes, suffer the problem of the early death of CHs (due to high load on cluster members), as a solution, Power Efficient Gathering in Sensor Information Systems (PEGASIS) [8] was proposed. Similar to the aforementioned algorithms, PEGASIS also operates in rounds. At the beginning of each round, all the nodes virtually align themselves in one single chain, with any one node being a leader. Each node communicates only with a close neighbor and takes turns transmitting to the BS, thus reducing the amount of energy spent per round. Though it outperforms LEACH by 100% to 300% (for FND_Statistics[2]), but its practical deployability is not a trivial task. PEGASIS relies on a far fetched assumption that nodes have global knowledge of the network, *viz.*, every node knows the location of all other nodes in the network, which makes it poorly scalable. Also, it is not suited for delay sensitive applications because whenever a node dies, the entire chain is reconstructed, which incurs, large delays. Similar to PEGASIS, in [13], authors proposed 'green' and 'udreen' algorithms for Gaussian and uniform distributed sensor networks respectively.

Recently, in [14] chain based and tree-based routing schemes were merged. Authors propose a novel multi-branch tree-based clustering approach to extend the lifetime of the WSNs. This protocol incorporates the concept of independent node set (INS)[3] and dominant set in the construction of routing tree. The main idea of this approach is to create 'n' levels and then for each level designate an INS which leads to the formation of the backbone of the tree. The levels start from sink and end at leaf nodes. Sink acts as a parent for INS of level 1, and level 1 INS nodes act as parents for INS nodes of level 2 and so on. This leads to the tree construction in the network. The remaining nodes at each level create virtual chains terminating at some INS of the same level. These chains can be visualized as sub-branches to the main routing tree.

In GSTEB [9] authors proposed to make a single node as the root node (node having maximum residual energy) rather than electing multiple CHs. On the beginning of each round, sink broadcasts the node ID of the elected root node. Each node in the field has only two alternatives; if there is some node present in between the transmitting node and the root node, transmitting node elects the intermediate node as next hop otherwise it assumes root node to be its next hop directly.

Tree Based Clustering (TBC) [10] is also an improvement of LEACH. Initially, *p%* of nodes are elected as CH followed by cluster formation phase. In LEACH each cluster member communicates directly to its CH, which leads to high energy dissipation of those sensor nodes which resides far from the CH. To avoid long link communication inside each cluster, TBC divides each cluster into α levels, where α is a design parameter. Node residing in Lth level elects the closest node belonging to $(L-1)$th level. Eventually, a tree like structure is created inside the cluster rooted at CH which is assigned the 0 level. TBC can also be regarded as a hybrid approach as it incorporates both hierarchical (clusters are formed) and flat routing scheme (inside cluster multi-hop forwarding takes place).

3. System model

3.1. Network model

In our model, we have assumed that N number of nodes are randomly deployed in the terrain, and each node has an *ID* (unique identity) associated with it. A node is represented as i (its ID), and $N(i)$ is a set of alive neighbors of node i. Nomenclature of all the symbols are tabulated in Table 1.

Assumptions

1. Sensor nodes are stationary and are randomly deployed in the terrain.
2. The sensor nodes are aware of their locations through some localization techniques [15].
3. Each node is capable of aggregating data, received from its neighbors.
4. The communication channel is reliable and error free.
5. There is only one Base Station (or Sink) which is fixed and can be placed at the center, corner, or at far from the terrain.

3.2. Aggregation model

In HEEMP, each node is capable of doing data aggregation, as it significantly decreases the energy consumption of the network, Where each node correlates receiving data with itself, discard the redundant data and summarize the whole data in a single packet [16]. Fig. 1 depicts the simplistic scenario, where nodes are placed in linear order. Each node has data packets to be transmitted to the sink. In the first case Fig. 1(a) shows, since each node transmits (without data aggregation),

[2] The time from the start of the network operation to the death of the first node in the network.
[3] set of nodes in which no node is the immediate neighbor any other node.

Table 1
Nomenclature table.

Symbol	Description
FND	First node dead
LND	Last node dead
BS	Base station
CH	Cluster head
d	Distance
d_o	Threshold value of distance
E_{elec}	Energy consumed by transmitter and receiver circuitry
E_{fs}	Energy consumed in free space model
E_{mp}	Energy consumed in multi path model
k	Data packet size
NCM	Non cluster member
CM	Cluster member
N	Total number of nodes
D_{agg}	Set of nodes performing data aggregation
ID	Identity of a node
ROI	Region of interest
$\|A\|$	Cardinality of set A
NS	Neighbor set
PNS	Progressive neighbor set
R	Route set
LR	Legitimate route set
BR	Best route set
α	The number of levels

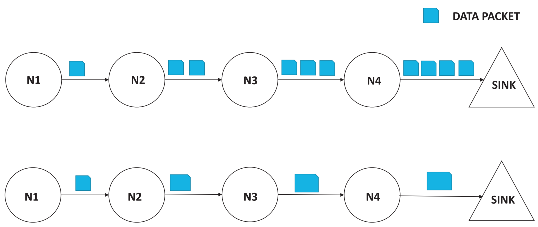


Fig. 1. Data transmission with (a) No Data aggregation (b) Data aggregation.

a total of $N(N+1)/2$ data packets are observed in the network. While for another case (using data aggregation) Fig. 1(b) depicts, a total of N data packets are observed in the network (each node aggregate its sensed data with the data received from the preceding node). Hence using data aggregation, $(N+1)/2$ data packets would be reduced in the network.

3.3. Energy model

In wireless sensor networks, energy scavenging is of utmost importance as each sensor node has a limited battery supply. WSN once deployed is left undisturbed with an intention of periodic (or event driven) data collection. A sensor node consists of many functional units constituting sensor, processor, memory, battery and transceiver unit. It is an established fact that among all, transmitter consumes maximum energy [17]. The first order radio model suggests that if a node i has to transmit k bit data to node j, which are d distance apart, then energy consumed by node i is given as

$$E_{Tx}(k,d) = \begin{cases} E_{elec} * k + E_{fs} * k * d^2 & if(d < d_0) \\ E_{elec} * k + E_{mp} * k * d^4 & if(d \geq d_0) \end{cases} \tag{1}$$

And energy consumed by node j is given as

$$E_{Rx}(k,d) = E_{elec} * k \tag{2}$$

In the above equations, E_{elec} represents the energy that is consumed by transmitter or receiver circuitry. E_{fs} and E_{mp} indicate the energy consumed by the transmitter amplifier for free space and multipath model respectively. And d_0 is the threshold value equals to $\sqrt{\frac{E_{fs}}{E_{mp}}}$.

4. HEEMP algorithm

HEEMP is a self-sustained routing protocol, which primarily aims at energy scavenging by refraining communication along long links. At any instance of time, each node stores the information about its neighbors[4] only, thereby reducing the memory requirements. Initially, each node with in a network sends their *Nbr_Table*[5] information to the sink using multicasting instead of flooding. The initiator node starts sending the neighbor information to the base station through neighboring nodes. Any sensor node will forward the *Nbr_Table* information packet only once for any source node to avoid the looping in the network. For doing this, each node maintains a received neighbor information list. Therefore, it reduces the traffic in the network and conserves the energy. The base station stores the state of each sensor node to elect cluster heads and to determine the path between cluster heads and itself.

In HEEMP, network management task (such as CH selection) has been taken away from nodes and are given to sink, to reduce the overall complexity of the network. Since the sink has complete topology information, it elects CHs (based on residual energy and node degree) and broadcasts their node ID and location to other nodes. When a node receives these messages from the sink, it sends the join-request message (Join_REQ) to the cluster-head which is located nearest to itself. This message includes the node's ID, location information and remaining energy (neighbors on reception of this message store the CH ID of the transmitting node). Once the clusters are formed, cluster members adopt multi-hop routing (among the members of the same cluster) to communicate with CH. In HEEMP, CHs do not transmit directly to the sink, rather they communicate along the pre-determined path (received from the sink) avoiding the long links as described in Fig. 2. HEEMP works in two phases *viz., setup* and *routing* phase.

4.1. Setup phase

Sink has complete topology information (ID, location and remaining energy of each node). All those nodes for which, the sink is in communication range form the *Direct_Set*.[6] Amongst the remaining nodes sink designates some nodes as CHs. For this, it calculates a parameter, *chance of election (CE)* (for each node) based on normalized values of remaining energy and node degree. It also ensures that elected CHs are spatially far so as to have balanced distribution of clusters. Thus, it always verify that all the selected CHs are at least *dist_threshold*[7] apart from each other. Primarily, the node having the highest CE parameter is directly elected as CH. The node having the second highest CE value should be elected as next CH, but if its distance from first CH is less than *dist_threshold*, it will not be included in the selection and next node based on the third highest CE value will be evaluated. This process continues until the desired number of CHs (*viz.*, 5–6% of total nodes excluding direct set nodes) are not found (formally described in procedure *ClusterHead_Election*). Since all the CHs must be at least *dist_threshold* apart from each other, it may happen that all the nodes have been checked but the total number of desired CHs are still not elected. Then operation proceeds with the selected number of CHs only (which are less than the desired number of CHs).

Once the CHs are elected, sink constructs routes between CHs and itself. Since it has the complete topology information, it selects the optimal route such that the total energy consumption of the routing path is minimum. For each CH, it creates a *Route Set (R)* containing all possible routes between CH and itself using depth first traversal operation. It then creates a *Legitimate Route Set (LR)* which contains those routes in which all the nodes have residual energies above the predefined threshold. Generally, a route consists of multiple sensor nodes which adopt multi-hop communication to transmit packets *from CH to sink*. If any route has sensor node (or nodes) which has remaining energy less than the predefined threshold, is not included in LR set. Then for all the *legitimate routes*, sink calculates the energy cost, which is the sum of the energy cost of each node belonging to that route. The energy cost of a sensor node is the ratio of energy expended in transmission (to its immediate neighbor) to the remaining energy of the node. The route which has least energy cost is considered as the best route between CH and sink as shown in Fig. 3.

After all the CHs are elected and paths have been constructed, sink broadcasts the CH ID's and location information in the network. Nodes on reception of the CHs information send join request to the nearest CH (and neighboring nodes store its CH ID in their Nbr_Table). Thus, at the end of the setup phase, each node has a complete information about its neighbors (its ID, location, remaining energy, and its CH ID), and each CH is aware of all its cluster members.

---

[4] Neighbors are those nodes which lie in the communication range of a given node.

[5] The *Nbr_Table* of each node has only four fields *viz.*, Neighbor ID (*Nbr_ID*), Location of neighbor (*Nbr_Loc*), Residual energy of the neighbor (*Nbr_Residual_Energy*) and Neighbor's Cluster Head ID (*Nbr CH ID*).

[6] These nodes are not elected as CH, and transmits their sensed information directly to sink.

[7] which is twice the communication range of a node [18].

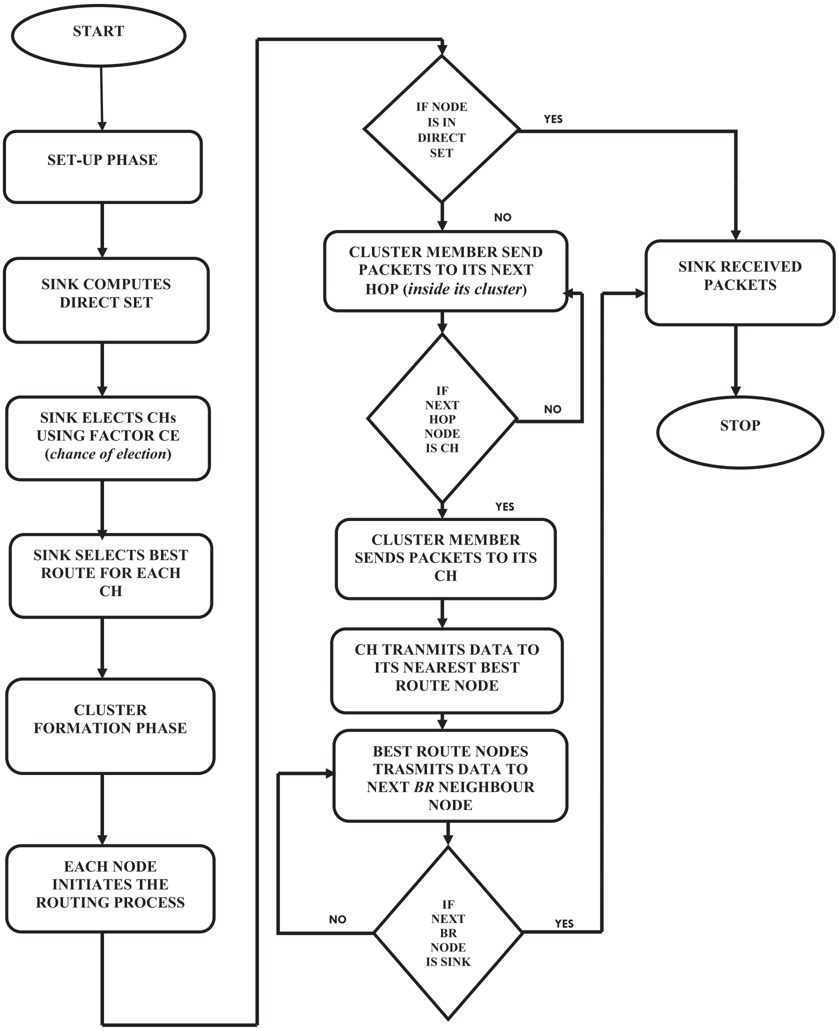


Fig. 2. Flow chart of HEEMP protocol.

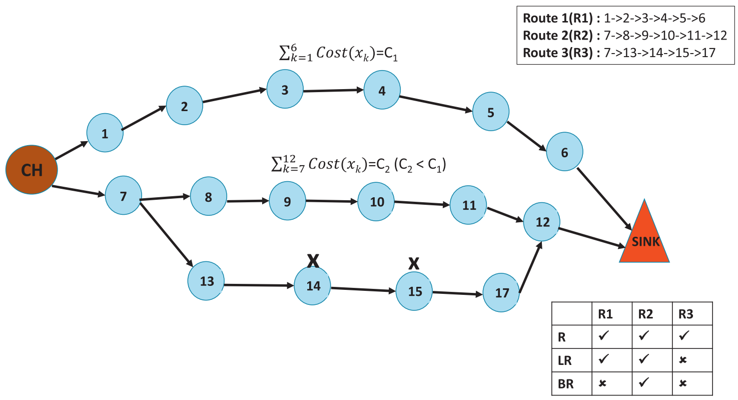


Fig. 3. Route selection by sink: R1, R2 and R3 represents possible *routes (R)* from CH to sink. R1 and R2 qualify to become *legitimate routes* (represented as LR) whereas R3 could not, as nodes 14 and 15 have residual energies, less than threshold. Among *legitimate routes*, R2 qualifies as *best route* (represented as BR), because it has least energy cost value.

Algorithm 1 HEEMP: setup phase.

```
procedure SETUP PHASE(Total number of nodes N, Desired number of Cluster Heads Total_CH)
    declare Direct_Set = {}          ▷ Set containing those node ID's for which sink is a direct neighbor
    declare CH = {}                  ▷ Set containing node ID's of cluster heads
    for every node n ∈ N do
        if Sink is in T_x then
            Add node n to Direct_Set
        end if
    end for
    CH = ClusterHead_Election(N, Direct_Set)       ▷ Sink elects cluster heads (CH)
    BR=Route_Selection(CH)           ▷ Sink select best route (BR) from each CH to itself.
    Sink broadcasts CHs information (ID and location) in the network.   ▷ Each nodes receives the ID of CHs from sink
    for each node n ∈ N and n ∉ bestroute do
        Node n selects closest CH to latch with
        Node n sends Join_Req to CH   ▷ It adjusts its transmission power to max(dist_to_CH, dist_to_farthest_Nbr). All the Nbrs of node n stores its CH ID in their Nbr Table.
    end for
end procedure
```

```
procedure CLUSTERHEAD_ELECTION(Array of node N, Direct set Direct_Set)
    for every node n ∉ Direct_Set do
        Calculate chance of election CE
```

$$CE = \left(\frac{ResidualEnergy}{InitialEnergy}\right) * \left(\frac{NodeDegree}{max(nodedegree)}\right)$$

```
    end for
    Sort nodes N in descending order of their CE values
    declare itr = 0
    while |CH| < Total_CH do
        if ∀ch ∈ CH : dist(ch, N(itr)) > dist_threshold then     ▷ it ensures CH elected are spatially far
            CH = CH ∪ N(itr)
        end if
        itr = itr+1
        if itr == |N| then
            break
        end if
    end while
    Return CH
end procedure
```

4.2. Routing phase

In this phase, each node transmits its sensed data to the sink. Direct_Set nodes send their data directly (to the sink), whereas rest of the nodes adopt the following routing scheme. At this stage, each node has a filled Nbr_Table. Each cluster member iterates through its Nbr_Table and all those neighbors whose CH ID (fourth field of the Nbr_Table) matches with its own CH ID are shortlisted. Amongst the shortlisted candidates, all those nodes which are present in between itself and CH (by comparing their location with its own location) are stored into *ProgressiveNodeSet* (*PNS*) depicted in Fig. 4. Nodes *m*, *j*, and *k* constitute the Neighbor Set (*NS*) of node *i*. Any node which belongs to *NS*, and is also present in between Node *i* and sink, constitutes the *PNS* (nodes *j* and *k*). For instance, node *j* must satisfy the following two conditions, to become the member of *PNS* of node *i*, (1) node $j \in NS$ (2) $dist_i^{sink}$ - $dist_j^{sink} > 0$.

Now, a cluster member can select its next hop in two different ways *viz.*, being *philanthropist* or being *selfish*. In the former approach node belonging to *PNS* having highest residual energy is elected as next hop. Whereas, in latter approach the node closest to the transmitting node is selected as next hop. In philanthropist mode, transmitting node takes neighboring nodes' remaining energy into consideration whereas in selfish mode, transmitting node only aims at reducing its own energy expenditure by sending packets to the nearest progressive node. If *PNS* is empty it performs the same process for entire Nbr_Set. From all the neighboring nodes, it selects the node either on the basis of residual energy (in philanthropist mode)

procedure ROUTE_SELECTION(Elected Cluster Head nodes CH)
declare R = {} ▷ **represents set of all possible routes between CH and sink
declare LR = {} ▷ represents set of legitimate routes between CH and sink (a route consisting of nodes having remaining energy greater than threshold). Any legitimate route is always a route thus $LR \supseteq R$.
declare BR = {} ▷ represents set of best route between each CH and sink
for each CH **do**
$R = R_1, R_2, R_3, \ldots .R_j$ ▷ number of n available routes between CH and sink where R_j represents set of nodes $x_1, x_2, \ldots, x_i$ is in Route R
for each route $R_j \in R$ **do**
if $\forall x \in R_j : E_{res}(x) < E_{threshold}$ **then**
$LR = LR \cup R_j$ ▷ LR represents set of legitimate routes
end if
end for
for each route $LR_j \in LR$ **do**

$$cost(LR_j) = \sum_{\forall x \in LR_j} \frac{Tx(x)}{E_{res}(x)} \tag{3}$$

end for
$bestroute_{CH} = min(cost(LR))$ ▷ Legitimate Route with minimum cost value is selected as best route between CH and sink
$BR = BR \cup bestroute_{CH}$
end for
Return BR
end procedure

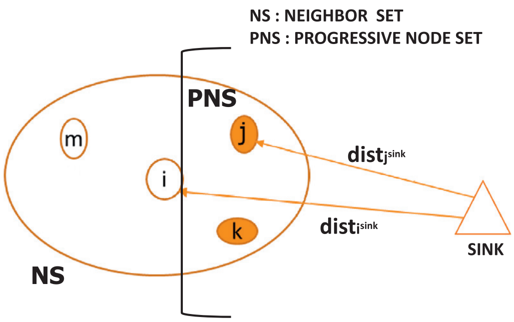


Fig. 4. Progressive node set.

or on the basis of distance to itself (in selfish mode). If a node has empty Nbr_Set, it transmits data directly to CH. After receiving data from all the neighbors, CH performs data aggregation and transmits data along the route received by the sink. Route nodes on reception of packets (from their preceding route node), aggregates their own sensed data and forwards it towards the sink (Fig. 5).

5. Modified-TBC (M-TBC)

Similar to HEEMP, tree based clustering (TBC) [10] aims at avoiding long links inside the cluster. Thus, it is an apt choice for performance comparison with HEEMP. While evaluating TBC, we observed some limitations in the protocol, and thus we improved TBC and termed it as M-TBC.

M-TBC is an extended version of TBC. In TBC, each node transmits its sensed information to the node residing in its preceding level. This approach was proposed to avoid direct transmission between cluster members and CH (which in many cases happens at long links resulting faster energy depletion). In M-TBC, we introduce the notion of *Intermediate_Node_Set*. Any node i when transmitting data to its preceding level node j, will check whether node j is present in between node i and CH. If node j, satisfies the condition it is added to *Intermediate_Node_Set*. Among all the nodes belonging to this set, node closest to node i, is selected as next hop. If *Intermediate_Node_Set* remains empty, node i, directly communicates to CH. M-TBC is formally presented as Algorithm 3.

The cluster formation phase is directly adopted from TBC, where 5% of nodes are randomly elected as CHs. Then protocols attempts for tree formation among the nodes. For this, each CH determines the level of each member node in the cluster. Let d_{max} denotes the distance of the farthest node from the CH and α the number of levels decided according to the size of

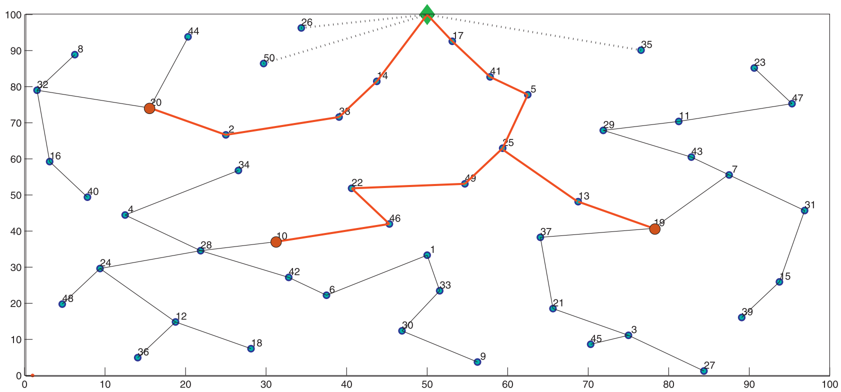


Fig. 5. MATLAB simulation of HEEMP routing process.

Algorithm 2 HEEMP: routing phase.

---

procedure ROUTING(Total Nodes N, Elected cluster heads CH, Best route (to CH) BR)
  declare *ProgressiveNodeSet* = {} ▹ Neighboring nodes present in between node *n* and the sink.
  for each node n ∈ N and n ∉ Direct_Set and n ∉ CH and n ∉ BR **do**
    for each *nbr* ∈ *NbrSet*(*n*) and $CH_{nbr} == CH_n$ **do**
      if dist(n,CH)-dist(*nbr*,CH)>0 **then**
        Progressive_Node_Set(n) = Progressive_Node_Set(n) ∪ *nbr*
      end if
    end for
    if Progressive_Node_Set(n) ≠ *Null* **then**
      ∀*nbrs* ∈ Progressive_Node_Set(n)
      if Philanthropist mode **then**
        Select *nbr* as next hop node which has maximum residual energy
      else ▹ Selfish mode
        Select closest *nbr* as next hop node
      end if
    else if NbrSet(n) ≠ *Null* **then**
      ∀*nbr* ∈ NbrSet(n) : $CH_{nbr} == CH_n$
      Select closest *nbr* (in selfish mode) or which has maximum residual energy (in philanthropist mode) ▹ Amongst all those nbrs which belongs to same cluster as of node n, it selects the best one depending upon the mode of operation
      Node n sends its data packets to *nbr*
    else
      Send packets directly to CH
    end if
  end for
  for each node n ∈ CH **do**
    Perform data aggregation
    Send packet towards sink along its *bestpath* (received earlier from sink)
  end for
  for each node n ∈ Direct_Set and n ∉ *bestpath* **do**
    Send packet directly to sink
  end for
end procedure

---

the network. Then the average data transmission distance between two adjacent levels of the tree, d_a, can be calculated as d_{max}/α. Nodes lying in d_a distance from CH, belongs to level zero (L(0)) and nodes present in between d_a and twice of d_a from CH belongs to level one (L(1)) and so on.

Always ith level node will try to latch with $(i-1)$th level node. But while doing so, it ensures whether $(i-1)$th level nodes belong to *Intermediate_Node_Set*. If there are no nodes in this set, it directly communicates with CH. The main idea of creating different levels is energy scavenging by cluster members, just by transmitting packets to their previous level nodes and not directly to the CH. Therefore each node computes its parent node from *Intermediate_Node_Set*. Parent node selection is based upon the minimum distance between node and its *Intermediate_Node_Set*. Fig. 6 clearly depicts why M-TBC is better than TBC. In Fig. 6a, *parent_node*(1) is 2, *parent_node*(2, 3) is 4, and *parent_node*(4) is node 5. Nodes 2 and 3 rather than

Algorithm 3 M-TBC (Total number of nodes N, Desired percentage of Cluster Heads P, Current round r).

Calculate Threshold

$$T(n) = \begin{cases} \left(\frac{P}{1-P*(rmod(1/P))}\right) & n \in G \\ 0 & n \notin G \end{cases} \quad (4)$$

```
for each node n ∈ N do
    if rand(n)<T(n) then                ▷ rand(n) is the random number generated by node n
        Elect node n as CH
    end if
end for
for each CH do
    //Cluster advertisement phase
    CH broadcasts its ID and location in the network.
    for each node n ∈ N do
        Node n latch with closest CH                ▷ It sends Join_Msg to CH
    end for
    //Intra cluster Level Formation
    for each node ∈ CH do
        d_α = d_max/α
        ▷ where d_max denotes distance between farthest node and the CH and α represents number of levels in the Cluster
        for every node of n ∈ N do
            L(α)=n                ▷ assign level to each node
        end for
    end for
    //Intra cluster tree formation
    min_dist=∞
    Intermediate_Node_Set = ∅
    for each node i ∈ CH do
        for each node j ∈ L(i) − 1 do
            if dist(i, CH) − dist(i, j) < 0 then
                Intermediate_Node_Set(i) = Intermediate_Node_Set(i) ∪ j
                if dist(i,j)<min_dist then
                    parent(i) = j
                end if
            end if
        end for
        if parent(i)==NULL then
            parent(i)=CH
        end if
    end for
end for
```

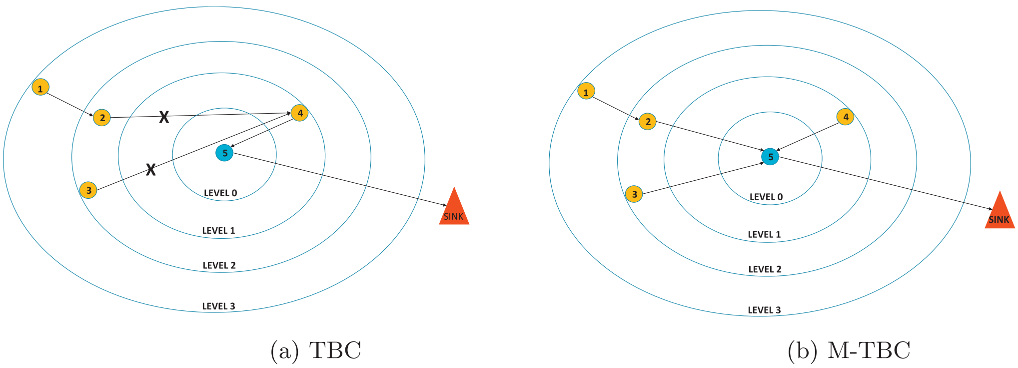
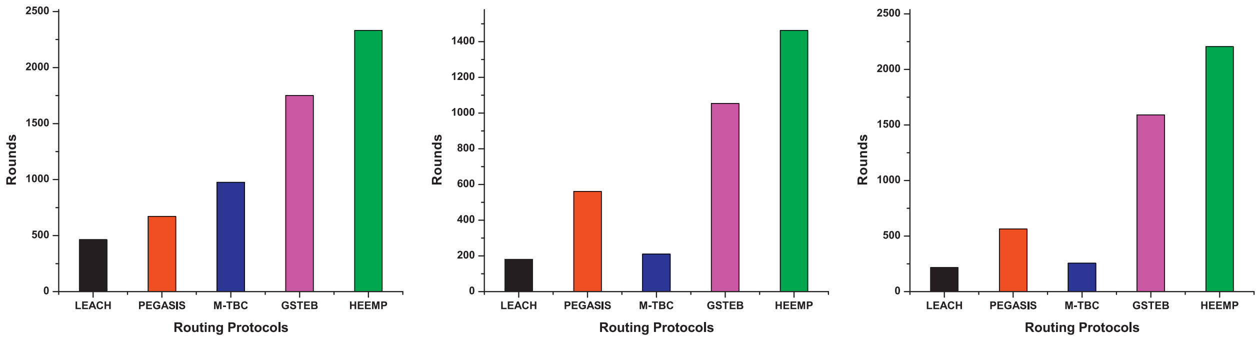


Fig. 6. Routing in TBC vs. M-TBC.

Table 2
Simulation parameters.

Parameters	Values
Free space energy Dissipation(E_{fs})	10 pJ/bit/m^2
Multi path energy Dissipation(E_{mp})	0.0013 pJ/bit/m^4
Transmitter energy Dissipation($E_{Tx-elec}$)	50 nJ/bit
Receiver energy Dissipation($E_{Rx-elec}$)	50 nJ/bit
Energy for Data aggregation (E_{DA})	5 nJ/bit/signal
Data packet size	2000 bits
Control packet Size	200 bits
Initial energy	0.5 J
Communication range	30 m
dist_threshold	60 m



(a) Sink at Center (b) Sink at Far (c) Sink at Corner

Fig. 7. FND_Statistics.

transmitting to node 4 must directly communicate with 5 (as for both nodes 2 and 4, *Intermediate_Node_Set*[8] is empty). M-TBC overcomes aforementioned problem and is shown in Fig. 6b.

In routing phase, each node is aware about its parent node. Hence, each cluster member transmits its data to its respective parent node and CH communicates directly to the sink.

6. Comparative analysis and simulation results

In this section, we validate our claims by extensive MATLAB simulations and compared the performance of HEEMP with LEACH[5], PEGASIS[8], TBC[10], M-TBC, and GSTEB[9] routing protocols. Throughout our simulations, we follow the same network parameters as, described in Table 2. To evaluate the performance of HEEMP, we use various performance metrics described as follows.

6.1. Network life time

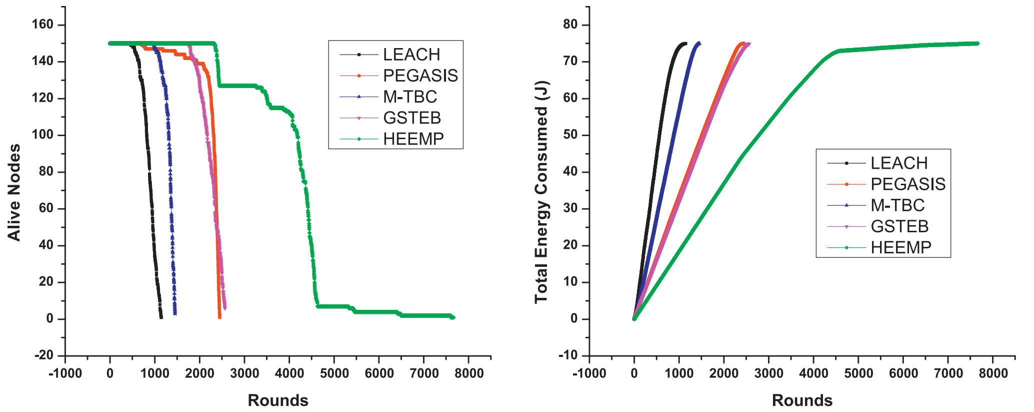
In this work, we adopt *First Node Dead Statistics (FND_Statistics)* as a metric for network lifetime. It is defined as the interval between the rounds where the first node starts transmitting data and the round where the first node gets dead [9,19]. For the evaluation of network life time 150 nodes are deployed randomly in the terrain size of 200 × 200 m^2, under the following scenarios:

(1) sink at center (2) sink at far[9] (3) sink at corner

The results of the aforementioned scenarios are depicted in Fig. 7a–c. It can be inferred from the figures, that for all three aforementioned scenarios HEEMP outperformed its other counterparts. The performance gain [20] as per FND_Statistics has

Table 3
Performance gain of HEEMP.

Protocols	Sink location		
	Far	Center	Corner
LEACH [5]	708%	403%	920%
PEGASIS[8]	160%	246%	290%
M-TBC	596%	139%	761%
GSTEB[9]	38%	33%	38%



(a) Alive node statistics (b) Total energy consumption of the network

Fig. 8. Alive node and total energy consumption Statistics.

been tabulated in Table 3 and is calculated by Eq. (3).

$$Performance\,gain = \frac{HEEMP - Protocol}{Protocol} * 100\% \quad (3)$$

The node death trend of the network (under various protocols) has been shown in Fig. 8a. It can be observed from the figure that for any network lifetime definition (1%, 10%,... ,100%) of nodes dead, HEEMP significantly performed better *viz.,* it nearly doubled the network lifetime (for HEEMP, more than 95% nodes die in 4625 rounds whereas for all others it is below 2550 rounds).

Table 3, summarizes the *performance gain* of HEEMP compared to other protocols, based on network lifetime. For different set of experiments (placing sink at different locations), with different routing protocols, HEEMP remarkably outperformed all its existing counterparts. For GSTEB, it achieves a minimum of 33% improvement, whereas for LEACH it achieves a maximum of 920% improvement.

6.2. Total energy consumption

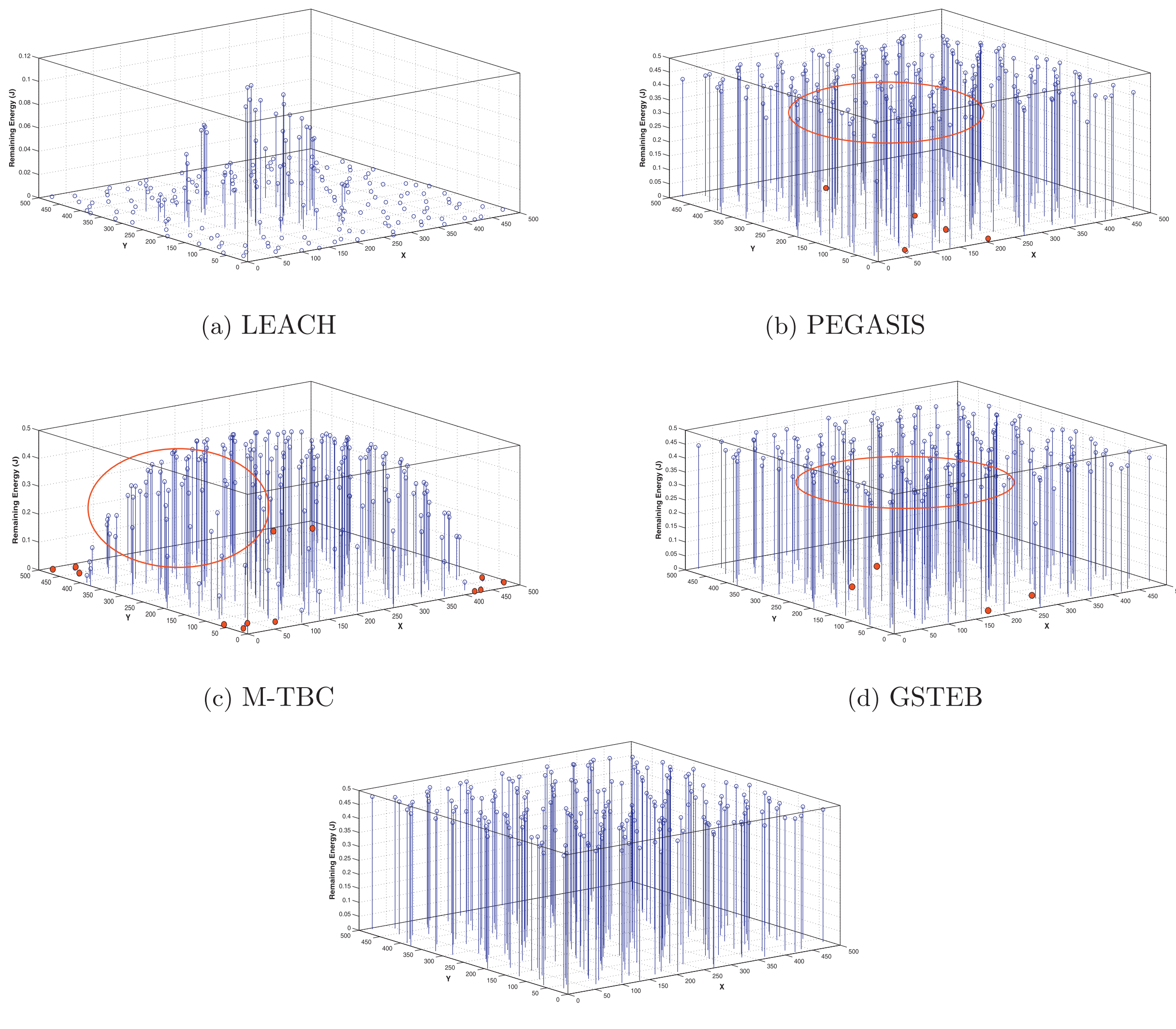
It is defined as the sum of energy consumed by all nodes in each round. Total Energy consumption (TEC) is expressed as:

$$TEC = \sum_{Rounds} \sum_{Nodes} (E_{Tx(Nodes)} + E_{Rx(Nodes)}) \quad (4)$$

From Fig. 8b it can be deduced that energy consumption rate of HEEMP is least as compared to all other routing protocol. When network is operated with HEEMP as underlying routing protocol, more than 94% of nodes expend their complete residual energy around 4600 rounds, whereas for all other protocols complete network is dead well before 2500 rounds. Thus, there is a huge discrepancy in the energy expenditure rate of HEEMP to its other existing counterparts.

In HEEMP, cluster members do no directly latch with CH, rather transmit packets in multi-hop manner. This reduces the *reception energy* of the CH. In LEACH, all the nodes directly transmit their packets to CH, but in HEEMP only the subset of the cluster members communicates directly with the CH reducing the reception energy. HEEMP aims at reduction in *transmission energy* of the sensor nodes also. No node transmit data along long links (distance greater than d_0) whereas

[8] $dist(2, 4) > dist(2, 5)$ and $dist(3, 4) > dist(3, 5)$ therefore $parent\_node(2, 3)$ must be 5.
[9] The nearest node to the sink is more than d_0 distance apart.



(a) LEACH (b) PEGASIS

(c) M-TBC (d) GSTEB

(e) HEEMP

Fig. 9. *Remaining energy* of sensors for different routing protocols after 200 rounds.

for all other protocols under consideration some nodes may communicate with nodes located far from them. It can be easily visualized in Fig. 8b, that total energy consumption graph ramps up slowly in comparison to other routing protocols, establishing the fact, that overall energy consumption of HEEMP is less than other routing protocols under consideration.

6.3. Energy balancing

Fig. 9, represents the residual energy of each sensor node (for different routing protocols) at 200 rounds. We placed 200 nodes in 500 × 500 m^2 field. Since the nodes are randomly dispersed in a larger terrain, they have to communicate along long links, expending their energy at a faster rate. In such situations, energy balancing plays an important role. Fig. 9e shows how well HEEMP achieves energy balancing over the network. Among all the routing protocols LEACH (Fig. 9a) achieves least energy balancing (with majority of the nodes dead) followed by M-TBC (Fig. 9c). Apparently, it seems that PEGASIS (Fig. 9b) and GSTEB (Fig. 9d) achieves energy balancing similar to HEEMP, but it is noteworthy, that after 200 rounds 5 nodes in PEGASIS and 4 nodes in GSTEB are dead ,[10] whereas for HEEMP all nodes are alive.

Table 4 provides further insights into the results, establishing the fact that HEEMP performs better in comparison to other protocols under consideration. We divided energy of each sensor node into 10 different levels with the step size of

[10] Nodes marked as red dots are dead nodes.

Table 4
Energy distribution level.

Levels	Energy range (J)	Node distribution (for various Routing Protocols)				
		LEACH[5]	PEGASIS[8]	M-TBC	GSTEB[9]	HEEMP
1	(0.45 – 0.5]	0	101	42	114	197
2	(0.40 – 0.45]	0	86	37	77	3
3	(0.35 – 0.40]	0	2	27	0	0
4	(0.30 – 0.35]	0	4	18	1	0
5	(0.25 – 0.30]	0	1	14	3	0
6	(0.20 – 0.25]	0	0	19	0	0
7	(0.15 – 0.20]	0	0	11	1	0
8	(0.10 – 0.15]	3	0	9	0	0
9	(0.05 – 0.10]	21	0	5	0	0
10	[0.0 – 0.05]	176	6	18	4	0

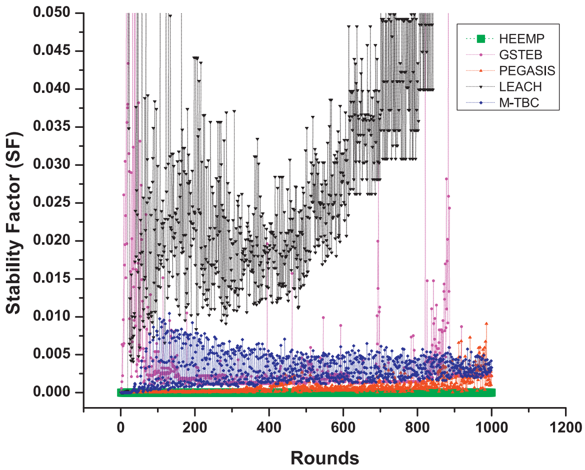


Fig. 10. Stability factor.

0.05 J (column two of the table). We recorded how many nodes are at which energy level operating under different routing protocols. Table 4 shows that with HEEMP 197 nodes are at level one (*viz.*, with least energy expense) followed by GSTEB (with 114 nodes) and PEGASIS (with 101 nodes). Except HEEMP, for all other protocols, some nodes are also present at last level (*viz.*, near complete depletion). These results ensure that HEEMP provides longer network lifetime with well-achieved energy balancing.

6.4. Stability factor

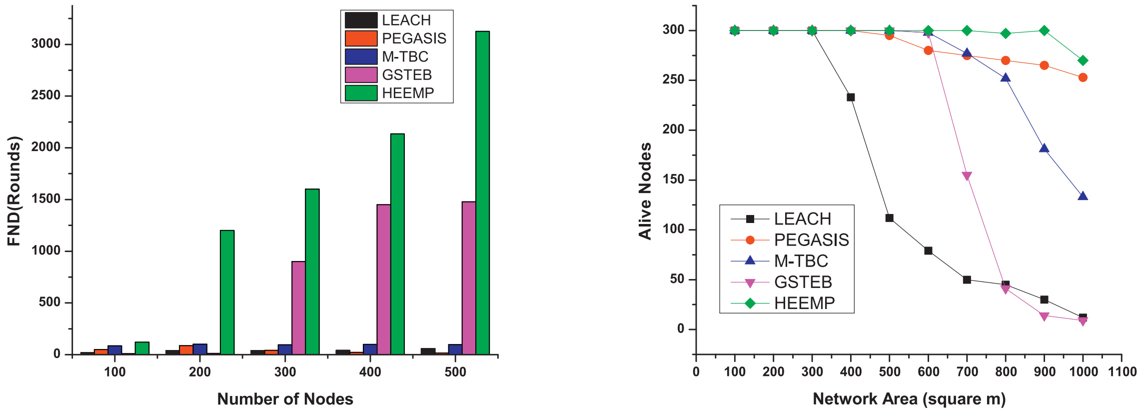
Section 6.3 provides insight for energy balancing specific upto one particular round (*viz.*, 200th round). In order to obtain a general trend for variability in energy consumption by the sensor nodes, we calculate standard deviation in energy consumption for each round and term it as *stability factor (SF)*:

$$SF = \sqrt{\frac{\sum_{i=1}^{N}(E_{avg} - E_{con}(i))^2}{N}} \quad (5)$$

where E_{avg} is the average energy consumption of the sensor nodes, and $E_{con(i)}$ is the energy consumed by the ith node in the current round.

We placed 500 nodes in 1000 × 1000 m^2 with sink at center and calculated SF for each round.[11] Again, HEEMP performed better than all its competing protocols as shown in Fig. 10. Extremely low and consistent SF of HEEMP can be attributed to the fact, that it leverages load balancing among sensor nodes. None of the sensor nodes, communicates along long links; intra cluster members adopt multi-hop communication towards CH, and CH transmits data along the predetermined path (in multi-hop manner) towards the sink. Thus, no node is overburdened with extremely high energy expenditure. Additionally, all the nodes perform data aggregation, which results in less packet generation in the network. For all

[11] Having observed the energy balancing characteristics in Section 6.3, we intentionally placed 500 nodes in large terrain size so as to test how well HEEMP (and other protocols) minimize the variability in the energy consumption by the sensor nodes, in comparatively larger terrain size.



(a) Scalability of routing protocols (FND_Statistics) (b) Nodes alive as a function of network area

Fig. 11. Scalability and network area of routing protocol.

other protocols, some nodes may transmit along long distances, expending their energy reserves faster than other remaining nodes.

In LEACH, all CHs transmit directly to sink and nodes far from sink[12] die out fast than rest of the nodes. Even in M-TBC and GSTEB, intra cluster members attempt to transmit their sensed data in a multi-hop manner, but if they fail to find a suitable neighbor, direct transmission is adopted for CH. In PEGASIS also, very often the leader elected is far from the sink. In its communication with sink, the energy consumption is very high in comparison to other nodes which transmit to their closest neighbors forming a chain. HEEMP does not suffer from these aforementioned problems and consistently maintains a very low SF, establishing the fact that energy consumption is uniform across all the sensor nodes under its operation.

6.5. Scalability

The WSN scalability is the ability of the network to assimilate more number of nodes that might not be foreshadowed during the initial network deployment stage. Scalability is a major design issue in the wireless sensor network domain because it specifies the systems capability to accommodate additional nodes up to certain threshold without restructuring the entire system. Therefore, the routing protocols used for WSN should support network scalability where such protocols should continue to do well, as the network grows larger [21].

From Fig. 11a, it can be inferred that while increasing the number of nodes from 100 to 500 (with the step size of 100 nodes) in the terrain of size 500 $\times$ 500 m^2, FND_Statistics of HEEMP considerably increased. It can be attributed to the fact that with increase in number of nodes (keeping the terrain fixed), the inter node distance decreased. In HEEMP, inside each cluster, multi-hop forwarding takes place within same cluster members. Thus, each node communicates along small communication links. But, in GSTEB all nodes transmit their sensed data to a single parent (in a multi-hop manner) without any cluster formations, resulting in communication along long links for some unfortunate nodes. Also, this single parent node communicates with sink directly, resulting in high transmission energy. Thus, FND_Statistics for GSTEB are always seen to be lesser than HEEMP. All other routing protocols have very low FND_Staistics because of aforementioned problems of frequent direct communication along long links.

6.6. Impact of network area variation

In order to test the suitability of HEEMP for large network areas, we evaluate the performance of HEEMP by increasing the ROI. For this simulation, we fixed the number of nodes to be 300 and increased the terrain size from 100 m^2 to 1000 m^2. Fig. 11b shows the alive node statistics of the five protocols, at the end of 100 rounds of activity, as a function of network area. Clearly, HEEMP outperforms all other four protocols. This is mainly because increasing the terrain size results in large inter-node distance, which causes nodes to expend more energy. In HEEMP, each node communicates with its neighboring nodes only adopting a multi-hop routing scheme to transmit data to its cluster head. Different cluster heads transmit their aggregated data to the sink along the pre-computed path, whereas in all other routing protocols, cluster head (or in GSTEB, parent node) directly communicates with sink, leading to high energy expenditure. It can be easily seen that more than 90% of the HEEMP nodes still remain alive for a network with area 1000 m^2, while the other four protocols encounter significant

[12] distance greater than d_o (long links).

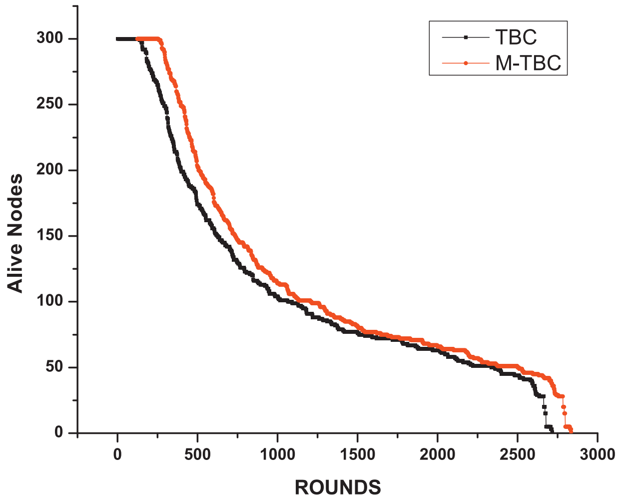


Fig. 12. Alive node statistics of M-TBC over TBC.

sensor node deaths. From these analyses, it is clear that HEEMP offers significant performance gain for networks with large coverage areas.

6.7. HEEMP operational modes

As already mentioned in Section 4, HEEMP has two neighbor selection modes *viz.*, Philanthropist mode (where neighbor selection is based on the maximum residual energy of neighboring node) and Selfish mode (where the nearest neighboring node is selected as next hop).

Our simulations confirm that LND_Statistics of selfish mode has an improvement of $\approx$15% over philanthropist mode ,[13] because in selfish mode, each node transmits its sensed data to its nearest neighbor but in philanthropist mode, a node may transmit its data to its farthest neighbor (considering maximum residual energy criteria). Initially, all sensor nodes have same battery capacity (any neighbor can be selected as next hop) thus impact of philanthropist or selfish mode cannot be captured for FND_Statistics (we observed nearly same FND_Statistics for both modes of operation). But after many rounds of activity, when all nodes have depleted their battery capacity, HEEMP under selfish mode achieves longer network lifetime compared to philanthropist mode. It has been well captured by LND_Statistics; for selfish mode, it is 4641 rounds and for philanthropist mode, it is 4061 rounds. If HEEMP is operated in selfish mode, it will not only prolong the network lifetime but will also result in limited storage requirement and smaller packet size. In this mode, the node selects its nearest neighbor, so it need not store its neighbors' remaining energy value and also no node needs to reveal its remaining energy in the packets being sent (as it is not required by the neighboring nodes).

7. Improvement in tree based clustering (TBC)

In Section 5 we proposed Modified-TBC (M-TBC), an improvement in TBC protocol [10]. M-TBC considerably improves the overall network lifetime compared to TBC. Fig. 12 clearly depicts that M-TBC performs better than TBC keeping the network operational for a longer period of time. For any round under consideration, M-TBC always has more number of alive nodes as compared to TBC. Thus, throughout our research, we opted M-TBC (as a better alternative to TBC) for comparison purposes with HEEMP. It has been shown in preceding sections that HEEMP has outperformed M-TBC for various performance metrics, indicating its supremacy over TBC also.

8. Limitations and future work

HEEMP relies on sink to compute the topology, which may result in longer delays. But, such delays are acceptable where the primary goal is to prolong the network lifetime. In future, we plan to extend our research to show the trade-off between increased delay and energy minimization. We also plan to use cognitive radio wireless networks (CRNs) [22–24] with HEEMP. Cognitive radio has the capability to sense the spectrum and determine the vacant bands. By dynamically changing its operating parameters, cognitive radio can make use of these available bands in an opportunistic manner surpassing the traditional fixed spectrum assignment approach in terms of overall spectrum utilization [25]. In order to test how well HEEMP works with CRNs, we aim to simulate it on NS2 in future.

[13] 300 nodes are deployed in the terrain size of 500 $\times$ 500 m^2 and sink is placed at center.

9. Conclusion

Conservation of energy is the main challenge in the development of wireless sensor networks. Thus, we have introduced a novel energy balanced routing protocol, which can adapt itself, under the centralized control of the sink. Our *hybrid energy-efficient multi-path routing scheme* consists of clustering and routing phases. We have developed an efficient strategy with which *sink* partitions the network into various clusters. We have also devised simple but elegant scheme to ensure that inside each cluster, nodes adopt multi-hop forwarding to communicate with cluster head. Finally, our proposed protocol ensures *that no node transmits greater than* d_0 *distance* as opposed to existing hierarchical approaches, which compel sensors to latch to at least one cluster head, which results in tremendous energy exhaustion of the nodes due to the large distance between cluster nodes and the cluster head.

In order to test the robustness of our approach, we simulated it with different parameters and settings *viz.*, varying the terrain size, number of nodes and location of the sink. Simulation results confirm that proposed protocol yields improved *network lifetime*, reduced *energy consumption* and provides high *scalability* compared to its existing counterparts.

Supplementary material

Supplementary material associated with this article can be found, in the online version, at 10.1016/j.compeleceng.2018.03.018.

References

[1] Ammari HM, Das SK. Fault tolerance measures for large-scale wireless sensor networks. ACM Trans Auton Adapt Syst (TAAS) 2009;4(1):2.
[2] Snigdh I, Gosain D. Analysis of scalability for routing protocols in wireless sensor networks. Optik-Int J Light Electron Opt 2016;127(5):2535–8.
[3] Ok C-S, Lee S, Mitra P, Kumara S. Distributed energy balanced routing for wireless sensor networks. Comput Indus Eng 2009;57(1):125–35.
[4] Gosain D, Snigdh I, Sajwan M. Dserr: Delay sensitive energy efficient reliable routing algorithm. Wireless Pers Commun 2017:1–20.
[5] Heinzelman WR, Chandrakasan A, Balakrishnan H. Energy-efficient communication protocol for wireless microsensor networks. In: System sciences, 2000. Proceedings of the 33rd annual Hawaii international conference on. IEEE; 2000. p. 10.
[6] Younis O, Fahmy S. Heed: a hybrid, energy-efficient, distributed clustering approach for ad hoc sensor networks. IEEE Trans Mob Comput 2004;3(4):366–79.
[7] Meng X, Shi X, Wang Z, Wu S, Li C. A grid-based reliable routing protocol for wireless sensor networks with randomly distributed clusters. Ad Hoc Netw 2016.
[8] Lindsey S, Raghavendra CS. Pegasis: Power-efficient gathering in sensor information systems. In: Aerospace conference proceedings, 2002. IEEE, 3. IEEE; 2002. p. 3–1125.
[9] Han Z, Wu J, Zhang J, Liu L, Tian K. A general self-organized tree-based energy-balance routing protocol for wireless sensor network. IEEE Trans Nucl Sci 2014;61(2):732–40.
[10] Kim KT, Lyu CH, Moon SS, Youn HY. Tree-based clustering (tbc) for energy efficient wireless sensor networks. In: Advanced information networking and applications workshops (WAINA), 2010 IEEE 24th international conference on. IEEE; 2010. p. 680–5.
[11] Gawade RD, Nalbalwar S. A centralized energy efficient distance based routing protocol for wireless sensor networks. J Sens 2016;2016.
[12] Huynh T-T, Dinh-Duc A-V, Tran C-H. Delay-constrained energy-efficient cluster-based multi-hop routing in wireless sensor networks. J Commun Netw 2016;18(4):580–8.
[13] Anand Chatterjee R, Kumar V. Energy-efficient routing protocol via chain formation in gaussian distributed wireless sensor networks. Int J Electron Lett 2017:1–14.
[14] Sivaraj C, Alphonse P, Janakiraman T. Independent neighbour set based clustering algorithm for routing in wireless sensor networks. Wireless Pers Commun 2017:1–23.
[15] Niculescu D, Nath B. Ad hoc positioning system (aps). In: Global telecommunications conference, 2001. GLOBECOM'01. IEEE, 5. IEEE; 2001. p. 2926–31.
[16] Rajagopalan R., Varshney P.K.. Data aggregation techniques in sensor networks: A survey2006;.
[17] Sohraby K, Minoli D, Znati T. Wireless sensor networks: technology, protocols, and applications. John Wiley & Sons; 2007.
[18] Amgoth T, Jana PK. Energy-aware routing algorithm for wireless sensor networks. Comput Electr Eng 2015;41:357–67.
[19] Chang J-H, Tassiulas L. Energy conserving routing in wireless ad-hoc networks. In: INFOCOM 2000. Nineteenth annual joint conference of the IEEE computer and communications societies. Proceedings. IEEE, 1. IEEE; 2000. p. 22–31.
[20] SOLAIMAN B. Energy optimization in wireless sensor networks using a hybrid k-means pso clustering algorithm. Turk J Electr Eng Comput Sci 2016;24(4):2679–95.
[21] Alazzawi L, Elkateeb A, et al. Performance evaluation of the wsn routing protocols scalability. J Comput Syst Netw Commun 2009;2008.
[22] Bukhari SHR, Siraj S, Rehmani MH. Ns-2 based simulation framework for cognitive radio sensor networks. Wireless Netw 2016:1–17.
[23] Bukhari SHR, Rehmani MH, Siraj S. A survey of channel bonding for wireless networks and guidelines of channel bonding for futuristic cognitive radio sensor networks. IEEE Commun Surv Tutor 2016;18(2):924–48.
[24] Ozger M, Akan OB. On the utilization of spectrum opportunity in cognitive radio networks. IEEE Commun Lett 2016;20(1):157–60.
[25] Akan OB, Karli OB, Ergul O. Cognitive radio sensor networks. IEEE Netw 2009;23(4).

Mohit Sajwan received his B.Tech. degree in Computer Science Engineering from Amrapali Institute of Technology, Uttrakhand Technical University, Dehradun and M.E. (software engineering) from Birla Institute of Technology, Mesra, Ranchi. Presently, he is pursuing his Ph.D. in wireless sensor networks from National Institute of Technology, Delhi.

Devashish Gosain received his B.Tech. degree in Computer Science Engineering from Guru Gobind Singh Indraprastha University, Delhi and M.Tech. (Computer Science) degree from Birla Institute of Technology, Mesra, Ranchi. Presently he is pursuing his Ph.D. in network anonymity privacy and anti-censorship from Indraprastha Institute of Information Technology, Delhi

Ajay K. Sharma is working as Director National Institute of Technology, Delhi. He received his B.E. (Electronics and Electrical Engineering) from Punjab University, M.S. (Electronics and Control) from Birla Institute of Technology, Pilani and Ph.D. in Electronics communication and Computer Engineering. His areas of interest are broadband optical wireless communication systems, optical transmission, WDM systems, Radio-over-Fiber and wireless sensor networks.