Balanced and Energy Efficient Multi-Hop Techniques for Routing in Wireless Sensor Networks

Article · October 2017

DOI: 10.1049/iet-net.2017.0063

3 authors:

Abdelwahab Fawzy
Minoufiya University

Mona Shokair
Minoufiya University

W. Saad
Minoufiya University

IET Networks

Research Article

ISSN 2047-4954
Received on 10th April 2017
Revised 31st August 2017
Accepted on 22nd September 2017
doi: 10.1049/iet-net.2017.0063
www.ietdl.org

Balanced and energy-efficient multi-hop techniques for routing in wireless sensor networks

Abd Elwahab Fawzy[1] ✉, *Mona Shokair*[1], *Waleed Saad*[1]

[1]*Electronics and Electrical Communications Engineering Department, Faculty of Electronic Engineering, Menouf, Egypt*

✉ *E-mail: abdelwabfawzy@el-eng.menofia.edu.eg*

Abstract: In Wireless Sensor Networks, the power resources of the nodes are significantly restricted. Hence, a special treatment for their available energy is deeply required. In long distance transmission, Multi-Hop (MH) techniques are preferred. Although MH minimizes the amount of energy cost consumed by each node along the path but finding the optimal routing path between nodes is still very interesting issues. This paper proposes a Balanced and Energy Efficient MH (BEEMH) algorithm that is developed based on Dijkstra algorithm. It gives great interest to the residual energy of nodes; hence higher energy nodes are exclusively elected to work as relays. Moreover, the total energy consumption at both TX and RX has been merged to model the weight of links between nodes. Finally, Dijkstra algorithm is employed to efficiently search for the minimum cost path. Furthermore, two proposed MH protocols are introduced. Both are mainly based on the BEEMH algorithm. MATLAB simulator has been used to evaluate BEEMH in comparison with other conventional algorithms such as; minimum transmission energy (MTE), energy saving oriented least-hop routing algorithm (ESLHA), and energy saving-oriented routing algorithm based on Dijkstra (ESRAD) under various scenarios of network models. Then the performance of our proposed protocols is compared with the related MH protocols.

Nomenclature

List of notations

k	length of data packet (bits)
k_{CP}	length of overhead control packet (bits)
d	transmission distance
E_{TX}	dissipated energy by a sensing node in transmission process
E_{Rx}	dissipated energy by a sensing node in receiving process
E_{relay}	dissipated energy by a sensing node in relaying data packet k
E_{elec}	dissipated energy in electronic circuit of the radio model
$\varepsilon_{\mathrm{amp}}$	the amplification cost factor of the radio model
$\varepsilon_{\mathrm{fs}}$	free space dissipated energy in the amplifier of the radio model
$\varepsilon_{\mathrm{mp}}$	multipath dissipated energy in the amplifier of the radio model
E_{DA}	dissipated energy in data aggregation process
E_o	node's initial energy
E_t	summation of nodes residual energy
E_{RS}	residual energy of the sensor node
E_{Avg}	average residual energy of all alive nodes in the sensing field
$G(V,\ E)$	weighted directed graph for the network
V	the set of vertices or nodes
E	weight of edges between nodes
S	source node
D	target node
Q'	the set of visited nodes
Q	the set of unvisited nodes
v	normal neighbour node
v_{cost}	dissipated energy cost to reach node v
v_{previous}	previous step node to reach node v in the optimal path
u	current node
r	total number of rounds over the simulation time
N	total number of nodes in the sensing field
N_{live}	number of alive nodes in the sensing field
N_1	number of nodes in the near field of BEERAD protocol
$N_{\mathrm{live}-2}$	number of alive nodes in the far field in BEERAD protocol
$N_{\mathrm{live}-j}$	number of alive nodes in group (j) of BEERAD-E protocol
n	number of groups in BEERAD-E
d_{Max}	maximum distance of a node from BS
d_{char}	characteristic distance
d_i	distance between node and BS

1 Introduction

The great advance in micro-electro-mechanical systems technology allows the feasibility of small and low-cost sensing nodes. These nodes are equipped with capabilities of sensing the surrounding environment, processing the sensed data, and wireless communication [1]. Wireless sensor network (WSN) is composed of numerous battery-powered nodes and base station (BS). Normal nodes work cooperatively to sense the physical phenomena and send the sensed data to BS. BS is a master node which has an unlimited power source and equipped with high capabilities. It acts as a gateway between the normal nodes and the end user. WSNs can be typically used to achieve continuous monitoring (CM) or event-detection inside the supervised area. In CM applications, each sensor node transmits its sensed dataperiodically while in event-detection driven (EDD) applications, once an event occurs, it is reported to BS by the sensors within the event area. There are many applications which use both CM and event-driven reporting such as WSN for temperature control, air pollution monitoring, forest fire detection, landslide detection, and natural disaster prevention [2]. Often, nodes are randomly deployed in a hostile environment by helicopter; hence, the maintenance process is an almost impossible task. Efficient utilisation of nodes' available power is one of the main trends in WSN design [3].

Routing process plays a pivotal role in saving nodes' energy. Hence, reliable and energy-efficient routing protocol become the leading target that is sought by all researchers [4]. Conventional routing techniques can be classified into two main categories; direct transmission (DT) and multi-hop (MH) transmission. In DT, each node sends its data directly to BS. Since the transmission cost increases exponentially with distance, hence a large amount of power is exhausted by far nodes. This leads to the excessive drain of their power. To overcome this problem, MH transmission is employed. In MH, nodes work cooperatively to relay data to BS. Hence, the total distance is portioned into small parts. Although MH techniques uniformly distribute the cost between the nodes, the minimum energy consumption is not always guaranteed. Therefore, the total cost might be greater than that in DT [5, 6]. Moreover, through the network operation, the nodes closest to BS will be overloaded with numerous data messages. This causes the hot spot problem which results in a faster drain of their energy [5, 7].

MH paths can be evaluated through several metrics such as the hop count, energy cost, and residual energy along the path. In terms of hop count, MH transmissions are categorised into two approaches: short-hop routing, where data is routed over many short hops and long-hop routing at which data is routed through fewer long hops. Haenggi and Puccinelli in [8] had mentioned several reasons which endorse the second approach such as the total energy consumption, the overhead cost [9], reliability, sleep modes, and delay. In terms of total energy consumption, it is often assumed that a reduction of the transmit (or radiated) energy yields a proportional reduction of total energy consumption. Even without taking into account receive energy, this is not true for any practical power amplifier. In particular, in low-power transceivers, the local oscillators and bias circuitry will dominate the energy consumption, so that short-hop routing does not yield any substantial energy benefit if a more distant relay node can be reached with sufficient reliability. Therefore, long hops routing is confirmed as more effective, secure, and competitive than the short ones.

From the point of view of the transmission range, nodes seek to control the variable range transmit power [10, 11]. Otherwise, the nodes may broadcast using the same transmission power level without any power control [12]. Certainly, undue high-power transmission causes the fast drain of nodes' energy and introduces superfluous interference. In [13], the lowest possible transmit power while preserving network connectivity was investigated. Turning to the residual energy of nodes through the path, a power aware routing protocol was investigated in [14]. Not only does it minimise the energy consumption per packet, but it also maximises the average residual energy of the nodes. This improves the network lifetime and keeps the variance of nodes' energy at the minimum. The previous metrics of MH paths can be considered separately or jointly. In [15], a novel link metric was designed to attain a trade-off between power consumption by the node and its hop distance to BS. Hence, a power aware routing protocol based on a modified Dijkstra algorithm was suggested. Chiang *et al.* in [16] introduced energy saving-oriented least-hop routing algorithm (ESLHA). It searches for the shortest path (SP) of energy consumption with the least cost. Undoubtedly, energy cost is a pivotal metric for nodes. However, in heterogeneous WSN, where nodes have different abilities of energy, computing power, and sensing range, balancing energy consumption is more important. Otherwise, unpredictable failures can be taken place.

In this paper, balanced and energy-efficient MH (BEEMH) algorithm based on Dijkstra is introduced. While most of the previously discussed works give much care to the basic metrics separately, BEEMH tries to a trade-off between all of them. Hence, small long hops, least cost, and higher residual energy paths can be obtained. BEEMH extends the nodes' lifetime by minimising and balancing the energy consumed over the selected path. Furthermore, in terms of BEEMH, two centralised protocols, namely balanced and energy-efficient routing approach based on modified Dijkstra protocol (BEERAD) and BEERAD-enhanced (BEERAD-E) are proposed. These protocols outperform the previous related ones in terms of network lifetime and throughput.

The rest of this paper is organised as follows: Section 2 presents some basics and assumptions such as the network and the radio models, before turning into a literature review about the related works in Section 3. Our proposed techniques are introduced in Section 4. At first, BEEMH algorithm is proposed. Then, in the light of BEEMH, two proposed centralised routing protocols: BEERAD and BEERAD-E are illustrated. Energy consumption of our proposals is analysed in Section 5. Thereafter, the simulation results and discussion are investigated in Section 6. Finally, conclusions are drawn in Section 7.

2 Network and radio models

WSN consists of numerous nodes that are deployed in a certain geographic area. To develop our proposals, the following assumptions about the network model are considered:

- All sensor nodes are stationary after the random deployment process.
- Each node knows its location using global positioning system module or certain localisation techniques [17].
- The network is heterogeneous; hence, all sensor nodes have different initial energies. Except BS has unlimited power resource.
- The transmission links between nodes are bi-directional.
- All nodes are able to control the transmission power according to their distance from the receiving nodes.

Moreover, we model the network as a weighted directed graph $G\,(V,\ E)$. If there are two nodes v_i and v_j, which are able to communicate with each other, a weight of the edge $e_{i,\,j} \epsilon\ E$ exists.

All sensing nodes can sense, process, transmit, and receive data. Each of these tasks consumes a specific amount of energy [18]. To specify these quantities of energy, the first radio model is used [19]. To transmit a k-bit data packet from source to destination through a distance d, the consumed energy by the source's transmitter (TX) $E_{\rm TX}$ is given in (1). It is divided into two terms: the component cost which is concerned with running the electrical circuits, and the amplification cost which makes the transmitted signal has the ability to overcome noise in its path. On the other side, the receiver (RX) of the destination consumes the component cost to run the electrical circuits. If the RX tries to aggregate or fuse the received data, additional aggregation cost is needed as in (2)

$$E_{\rm TX}\,(k,\ d) = \underbrace{E_{\rm TX-elec}\,(k)}_{\text{component cost}} + \underbrace{E_{\rm TX-amp}\,(k,\ d)}_{\text{amplification cost}} \tag{1}$$

$$= \begin{cases} k\cdot E_{\rm elec} + k\cdot \varepsilon_{\rm fs}\cdot d^2 & \text{if } d \le do \\ k\cdot E_{\rm elec} + k\cdot \varepsilon_{\rm mp}\cdot d^4 & \text{if } d > do \end{cases}$$

$$E_{\rm Rx}\,(k) = \underbrace{k\cdot E_{\rm elec}}_{\text{component cost}} + \underbrace{k\cdot E_{\rm DA}}_{\text{aggregation cost}} \tag{2}$$

where $E_{\rm elec}$, $\varepsilon_{\rm fs}$, and $\varepsilon_{\rm mp}$ are the expended energy per bit in the circuitry of either TX or RX, the free space loss coefficient, and multipath loss coefficient, respectively. d_o is the threshold distance and equals $\sqrt{\varepsilon_{\rm fs}/\varepsilon_{\rm mp}}$. $E_{\rm DA}$ is the expended energy per bit per signal. If a given node works as intermediate node to relay data between a source and a destination, its radio expends

$$E_{\rm Relay}(k,\,d) = E_{\rm Rx}(k) + E_{\rm TX}(k,\,d) \tag{3}$$

$$= \begin{cases} 2\cdot k\cdot E_{\rm elec} + k\cdot E_{\rm DA} + k\cdot \varepsilon_{\rm fs}\cdot d^2 & \text{if } d \le do \\ 2\cdot k\cdot E_{\rm elec} + k\cdot E_{\rm DA} + k\cdot \varepsilon_{\rm mp}\cdot d^4 & \text{if } d > do \end{cases}$$

3 Previous work

Dijkstra algorithm is a search graph algorithm that is used for finding the SP between two vertices in a graph [20]. It is the most classical and mature algorithm. As a result, this SP algorithm has been widely used in networks routing, most notably intermediate system (IS) to IS and open SP first [21]. Let $G\,(V,\ E)$ be a directed weighted graph having a set of vertices or nodes V. There is a source node $S\ \epsilon\ V$. V nodes are divided into two groups: the visited

Table 1 Different models of edges' weight

Algorithm	MTE	ESLHA	ESRAD
cost	$k * \varepsilon_{\rm amp} * d^n$	$k * (E_{\rm elec} + \varepsilon_{\rm amp} * d^n)$	$k * (E_{\rm elec} + \varepsilon_{\rm amp} * d^n + E_{\rm elec})$
hops number	large	moderate	small
disadvantages	• higher cost • bad in a dense network • energy independent	• does not consider the component cost at RX • energy independent	• does not consider the aggregation cost at RX • energy independent

nodes Q' that have already found the SP between them and S, and the rest of the nodes (unvisited nodes) Q. Dijkstra algorithm repeatedly selects from Q, the SP's nodes before adding them to Q'. It is terminated as soon as the target node S is reached. Dijkstra algorithm can be concluded in the following steps:

Step 1: Assigning a tentative cost for nodes (0 for our initial source node S and ∞ for the rest of nodes).
Step 2: Keeping a group of visited nodes Q'. It starts with S only.
Step 3: From the current node u, consider all of its unvisited neighbours $v \epsilon Q$ and calculate the sum of energy cost of u plus the cost from u to reach v. If summation is less than the current tentative cost of v, replace it with the new value.
Step 4: After considering all neighbours v, select the minimum cost node from v as current node u, remove it from the unvisited set Q and mark it as visited node $u \epsilon Q'$.
Step 5: If the destination node D has been marked as visited, the algorithm is terminated.
Step 6: From Q, return to step 3.

It is confirmed that the least distance between any two nodes is the straight line which connected them together, hence it is not acceptable to consider the physical distance as the weight of links between nodes. Therefore, we are concerned with estimating the weight of edges. We have to take the energy cost, not the physical distance as the weight of edges. A variety of approaches has been investigated for this purpose as follows:

(1) *Amplification cost:* In minimum transmission energy (MTE) [5], the component cost at nodes' transceiver was neglected compared with the amplification cost. So, the amplification cost which is exponentially dependent on the transmission distance dominates the energy consumption and was used as the weight of edges. As result, a greedy algorithm had been obtained where each node sends messages to the closest node on its way to the destination. Thus, many short hops paths were produced. This case suits low-density networks, where there are no constraints on the number of hops. However, in high dense networks, the situation becomes worst.
(2) *Component cost:* In specific networks, the amplification cost is neglected compared with the component cost. Hence, the energy consumption of the node's circuitry was introduced as the evaluation index. Therefore, the path which uses the least number of hops consumes minimum energy and is considered as the optimal path [22].
(3) *Transmission cost:* In [16], ESLHA was discussed where the total energy cost at the TX that includes the amplification cost plus the component cost was used as the weight factor of edges. Thence, the intermediate nodes were chosen such that the total transmission cost is the least. This phenomenon was strongly apparent in small area networks. Undoubtedly as the weight factor increases, the path's hops become more limited in comparison with the previous approaches.
(4) *Total radio cost:* In WSN, the component cost is identical for all nodes. Hence, it is not accepted to neglect the component cost of the RX's circuitry at the objective node during weight matrix formation. In [23], ES-oriented routing algorithm based on Dijkstra (ESRAD) was introduced, where the weight of two-connected nodes was modified by considering the total radio cost at both nodes. Table 1 summarises the major differences between previously mentioned works.

No doubt that, energy cost is a very important factor for nodes, but in heterogeneous WSNs, balancing this cost is a more critical issue. In CM applications, nodes continue to consume energy to perform their related tasks. As soon as the sensor node lost its energy, it is dead. With the death of first sensor node, the network becomes unstable. Hence, unpredictable failures can be taken place. Moreover, in EDD applications, the exhausted energy differs from one node to another depending on its specification, operation mode, and the relative position from BS. For example, the nodes that are near to the target node will be used to route larger number of data packets. As result, the variance of nodes' residual energy increases causing energy hole problem. One of the main problems of the previous MH algorithms that they do not consider the residual energy of nodes during the SP detection. Hence, low-energy nodes are exposed to be overloaded with numerous data messages. This causes the hot spot problem which results in a faster drain of their energy. Moreover, the processing cost at the relay nodes is neglected in comparison with the transmit and receive costs. This, of course, affects the optimal routing path selection process using Dijkstra. Hence, the small longer hops, least cost, and higher residual paths are not guaranteed to be selected as recommended in [8].

4 Proposed techniques

4.1 Proposed BEEMH algorithm

Either minimising or balancing the energy consumption are very critical issues in MH communication. They play a major role in prolonging the node's life by avoiding either the hot spot problem or the energy hole problem. In the light of the critical issues of previous works, Dijkstra-based algorithm BEEMH will be introduced. It differs from Dijkstra in the following points:

- The average residual energy of nodes is used as a threshold level for nodes' classification into high and low-energy nodes. Thus, higher nodes can be exclusively selected as relays, where lower ones are not overloaded with any relaying tasks. Therefore, the energy consumption is balanced.
- It modifies the evaluation index of nodes in the network by increasing the weight of the nodes to include not only the transmit and receiving cost, but also the processing cost. As the weight of edges increases, the number of hops decreases. Hence, the long-hop transmission can be achieved as recommended in [8] and the total energy consumption can be minimised. The pseudocode of BEEMH is given in Fig. 1.

4.2 BEERAD protocol

In this section, we are interested in testing the validity of implementing our proposed algorithm BEEMH in WSNs' routing. The main target of our proposal is to increase the stability period of the network, i.e. the period before the death of the first node. Therefore, a centralised routing protocol BEERAD will be constructed. Fig. 2*a* shows BEERAD timetable. Its operation begins with initiation phase, and then it is divided into rounds. Each round is accomplished through two phases: the setup phase and the steady-state phase.

(1) *Initiation phase:* After nodes' deployment process, BS broadcasts a discovery request toward all nodes asking them to identify themselves. For the first time, each node responds to this request with a control packet to authenticate its presence in the

%% Average Energy Estimation	Calculation of the average energy of nodes
$E_t = 0$	E_t = summation of nodes residual energy
For all nodes $v \epsilon V$	v = normal node, V = set of all nodes
$E_t = E_t + E_{RS}$	E_{RS} = node's residual energy
End	
$E_{Avg} = E_t / N$	N = total number of nodes, E_{Avg} = average residual energy
%% Initialization	Estimating the source node (S) and the set of un-visited nodes (Q)
For all nodes $v \epsilon V$	
If ($v = S$)	S = source node
$v_{Cost} = 0$	v_{Cost} = tentative cost of the source node
End	
If ($E_{RS} \geq E_{Avg}$)	
$v_{cost} = \infty$	Unknown tentative energy cost to reach node v from S
$v_{Previous} = []$	Previous step node of node v in the optimal path is not defined
$v \epsilon Q$	Adding all nodes except S to the set of unvisited nodes (Q)
End	
End	
%% Main Loop	
While (Q != [])	Q set is not empty
$u = \min(Q, Cost)$	a node $u \epsilon Q$, & has minimum energy cost to reach it from S
remove u from Q	Node u became visited node
If ($u = D$)	Condition, D = Destination node
Break	Break out from the loop
End	
for all $v \epsilon Q$ and nearby u	
If ($u_{Cost} + Cost(u, v) < v_{Cost}$)	New minimum cost path to v has been found
$v_{cost} = u_{Cost} + Cost(u, v)$	Replacing the tentative cost by the new minimum one
$v_{Previous} = u$	Set u as the previous step node of node v
End	Energy cost equation $k * (E_{elec} + \varepsilon_{amp} * d^n + E_{elec} + E_{DA})$
End	
End	
Return previous []	we get the optimal path in reverse order (D --› S)

Fig. 1 *Function BEEMH (G (V, E), source node [S], destination node [D])*

network. This packet contains the node's identification number, location dimensions, and the initial energy level. BS utilises the location information to classify these nodes according to their distance from BS (d_i), into two groups: near nodes and far nodes. The network structure of BEERAD protocol is shown in Fig. 2*b*. If the distance d_i is smaller than d_{char}, nodes are marked as near nodes, else it is far nodes. d_{char} is the characteristic distance which is used as a judging metric to specify whether to use DT or MH? [6, 24]. If the distance is greater than d_{char} MH is used, else DT is preferred. It is totally dependent on the design parameters [25] as follows:

$$d_{\text{char}} = \sqrt{\frac{-B + \sqrt{B^2 - 4AC}}{2A}} \tag{4}$$

where $A = \varepsilon_{\text{mp}}$, $B = -\varepsilon_{\text{fs}}/2$, $C = -2E_{\text{elec}}$.

(2) *Setup phase:* This phase is assigned for optimal paths' estimation. BS exploits the nodes' information about locations and energy levels to set up the weight matrix of the network as explained previously in BEEMH. Then, it uses Dijkstra steps to estimate the optimal data route for each node. As soon as the routes for all nodes are specified, BS will announce the results to the related nodes. Each node constructs its routing table and updates it with its next hop node. For further upcoming rounds, the routing table for far nodes is updated. On the other hand, the near nodes will use DT for rest of the network operation [25]; hence, their routing table would not need an update. This, of course, minimises the setup cost of the near nodes.

(3) *Steady-state phase:* This is the final stage where the energy levels of nodes are attached to the data packet. Then, they begin to transmit their data packet to the next hop nodes. The intermediate nodes relay these packets till they reach to BS finally. Once BS receives these packets from all nodes, it extracts the energy information before updating the weight matrix. As soon as this stage is finished, another setup phase begins and so on. The steps of BEERAD protocol is summarised in its flowchart in Fig. 2*c*. It is clear that the initiation phase is performed only once at the start of the network while both the setup phase and the steady-state phase are performed periodically at each round.

4.3 BEERAD-E protocol

Often, WSNs are deployed in a hostile environment. Thus, hundreds or thousands of nodes are randomly deployed. BS in BEERAD protocol has to deal with all these nodes simultaneously at the setup phase to find their optimal paths and report them with their next hops. As a result, BS is heavily overloaded. This may cause some failures. Moreover, stability is one of the major concerns in the advancement of WSNs. A number of applications of WSN require guaranteed sensing, coverage, and connectivity throughout its operational period. The death of the first node might cause instability in the network. Therefore, all the sensor nodes must be alive to achieve the goal during that period. For all aforementioned reasons, BEERAD needs more additional improvement. BEERAD can be enhanced by dividing the network into layers and the nodes into groups. Thereafter, the setup phase is performed for each group separately. In BEERAD-E, the nodes are

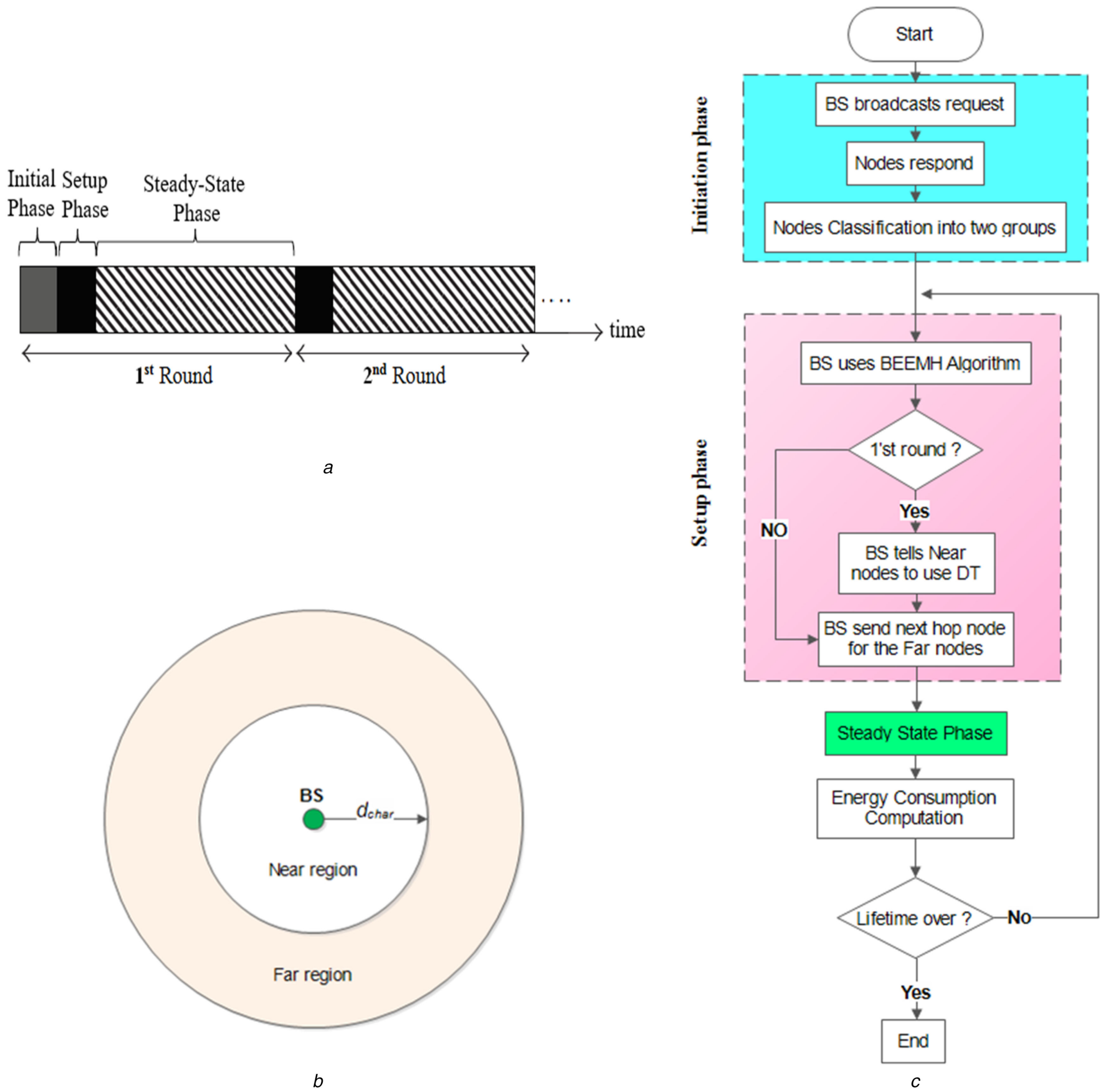

Fig. 2 *BEERAD protocol*
(a) BEERAD timetable, ***(b)*** BEERAD network structure, ***(c)*** BEERAD protocol flowchart

divided not only into two groups such as BEERAD, but also into larger number G_i, where $i = 1, 2, \ldots, n$. The network structure for BEERAD-E is shown in Fig. 3*a*. The number of groups can be calculated in terms of the maximum distance of nodes from BS $d_{\max}$ and the characteristic distance d_{char} as follows:

$$n = \frac{d_{\max}}{d_{\text{char}}} \tag{5}$$

In BEERAD-E, the route setup phase will not be performed only at the start of each round as in BEERAD, but it is performed for each group in a sequential manner within the same round. The setup phase of each group is followed by data transmission of its members. BEERAD-E operation begins with the nearest group to BS till the farthest ones as shown in Fig. 3*b*. Hence, BS is updated by the energy level of the inner nodes more than once within the same round. Therefore, BEERAD-E is more aware of nodes energy than BEERAD. This helps in balancing the consumed energy by nodes by a larger degree. The flowchart of BEERAD-E is given in Fig. 3*c*.

5 Energy consumption analysis

Now, we are interested in illustrating the energy cost consumed by nodes in each phase of our proposed protocols in comparison with other related protocols: MTE, ESLHA, and ESRAD.

(1) *Initiation phase:* This is the first step in our protocols where BS broadcasts the discovery request. The amount of energy consumed by each node to receive and respond this request is given as follows:

$$E_{\text{Initial}} = \underbrace{\underbrace{k_{\text{CP}}\, E_{\text{elec}}}_{\substack{\text{recieve request}\\ \text{from BS}}} + \underbrace{k_{\text{CP}}\, E_{\text{elec}} + k_{\text{CP}}\varepsilon_{\text{amp}}\, d_{\text{to BS}}^{n}}_{\substack{\text{transmit ID, position,}\\ \text{energy level to BS}}}}_{\text{only once at the start of the network}} \tag{6}$$

where k_{CP} and ε_{amp} are the length of the overhead control packet and the amplification cost factor, respectively.
(2) *Setup phase:* To estimate the optimal paths for nodes in the setup phase, a big effort is exerted by BS. As BS is assumed to have an unlimited power source, the whole attention has been given to the energy cost of the nodes. In this phase, the exhausted energy by each node is restricted to receive the control packet from BS to update its routing table as follows:

$$E_{\text{Setup}} = \underbrace{k_{\text{CP}}\, E_{\text{elec}}}_{\text{recieving next hop node from BS}} \tag{7}$$

MTE, ESLHA, and ESRAD are fully location-dependent algorithms. As the nodes in our network are assumed to be stationary, their optimal routing paths became relatively fixed. Therefore, nodes have the ability to dispense with updating their routing tables. This saves the setup cost of these protocols. This outcome is true and accepted until the beginning of nodes' death.

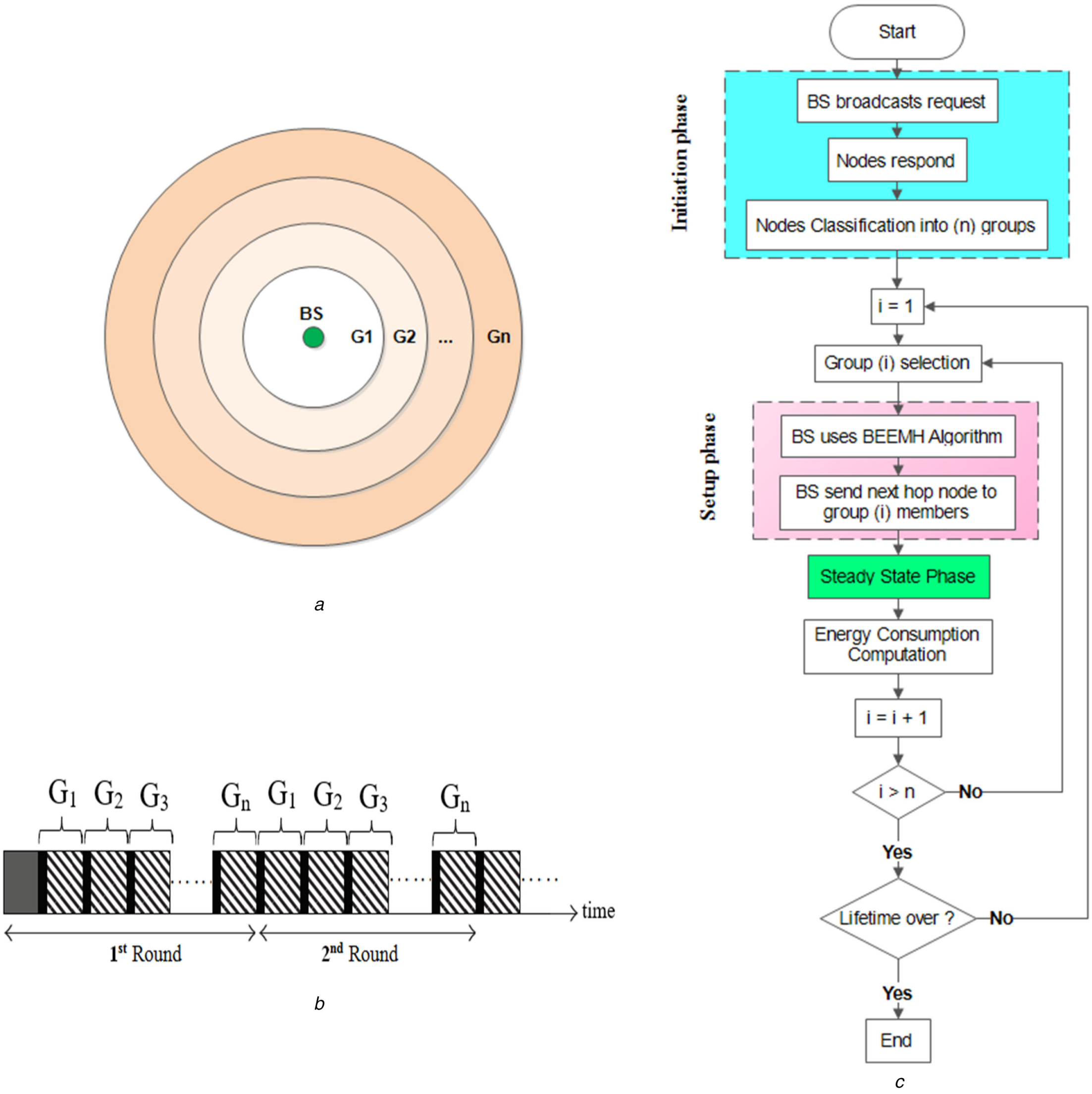

Fig. 3 *BEERAD-E protocol*
(a) BEERAD-E network structure, ***(b)*** BEERAD-E timetable, ***(c)*** BEERAD-E protocol flowchart

Table 2 Total setup cost of MH protocols

Protocols	Total setup cost over network lifetime
MTE, ESLHA, and ESRAD	$\sum_{i=1}^{N_{\text{live}}} k_{\text{CP}}\, E_{\text{elec}}$
BEERAD	$\sum_{i=1}^{N_1} k_{\text{CP}}\, E_{\text{elec}} + r \sum_{i=1}^{N_{\text{live}}-2} k_{CP}\, E_{\text{elec}}$
BEERAD-E	$\sum_{i=1}^{N_1} k_{\text{CP}}\, E_{\text{elec}} + r \sum_{j=2}^{n} (n-j+1) \sum_{i=1}^{N_{\text{live}}-j} k_{\text{CP}}\, E_{\text{elec}}$

At this time, the tables of some nodes would need updating to avoid passing through these dead nodes as intermediate nodes. On the other hand, both BEERAD and BEERAD-E are location and energy-based routing protocols. Owing to the continuous change of nodes' energy, the optimal paths and next hop nodes for most of the nodes are changed accordingly. As a result, nodes' tables need an update at each round. This increases the setup cost energy. The nodes of near region are an exception to update because they always use DT.

In BEERAD-E, the setup phase is performed intermittently depending on the number of groups. The routing tables of some inner nodes may be updated more times. For example, during the setup phase of G_i, routing tables of some nodes in G_{i-1}, G_{i-2}, and G_2 may need an update to play a more effective role in relaying data. Although this causes additional setup cost, it provides more balance in the energy consumption during the steady-state phase. The total setup cost consumed for all MH protocols are given in Table 2.

(3) *Steady-state phase:* The energy consumption by nodes in the steady-state phase is governed by (1)–(3).

6 Performance evaluation

6.1 Simulation parameters

Now, we are interested in evaluating the performance of BEEMH algorithm. MATLAB simulator is used to compare it with MTE, ESLHA, and ESRAD [26]. The evaluation process has been taken place under different degrees of network complexity. Hence, four diverse scenarios of networks are used. All networks are heterogeneous; therefore, nodes have different levels of initial energies. The simulation parameters are given in Table 3 [25].

For each network, the different algorithms have been employed to find the optimum route between two nodes. Thereafter, the selected paths are evaluated in terms of the hop number, length, the energy cost, and the average residual energy of each path.

6.2 BEEMH algorithm performance evaluation

(1) *The selected paths:* The selected paths by the different algorithms in WSN-I, WSN-II, WSN-III, and WSN-IV are shown in Figs. 4–7, respectively. In all figures, after nodes classification, the higher-energy nodes are shown in green, whereas the lower ones are shown in red. MTE is a greedy algorithm where each node sends data to its nearest neighbours. Hence, it always selects the

Table 3 Simulation parameters [25]

Parameter	WSN-I	WSN-II	WSN-III	WSN-VI
number of nodes	50	75	100	150
field dimensions	100 × 100 m^2	200 × 200 m^2	400 × 400 m^2	600 × 600 m^2
nodes density × 10^3	40/8 m^{-2}	15/8 m^{-2}	5/8 m^{-2}	3.33/8 m^{-2}
Initial Energy E_o			($0.5 < E_o < 1$) j	
E_{elec}			50 nJ/bit	
$\varepsilon_{\mathrm{fs}}$			10 pJ/bit/m^2	
$\varepsilon_{\mathrm{mp}}$			0.0013 pJ/bit/m^4	
E_{DA}			5 nJ/bit/signal	
data packet size k			4000 bits	

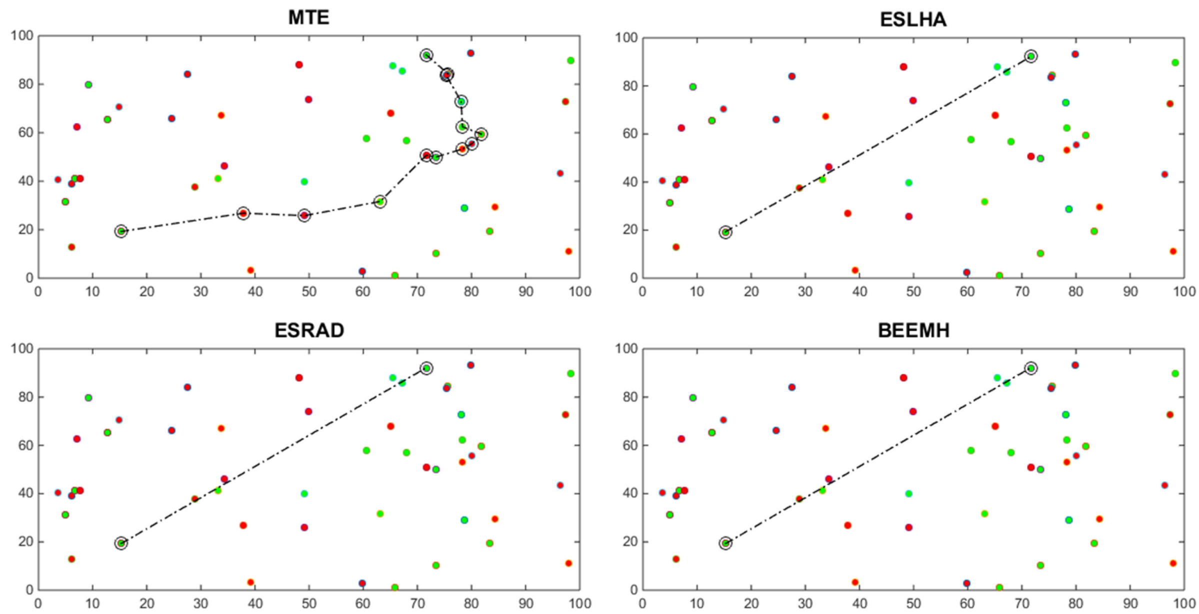

Fig. 4 *Selected paths in WSN-I*

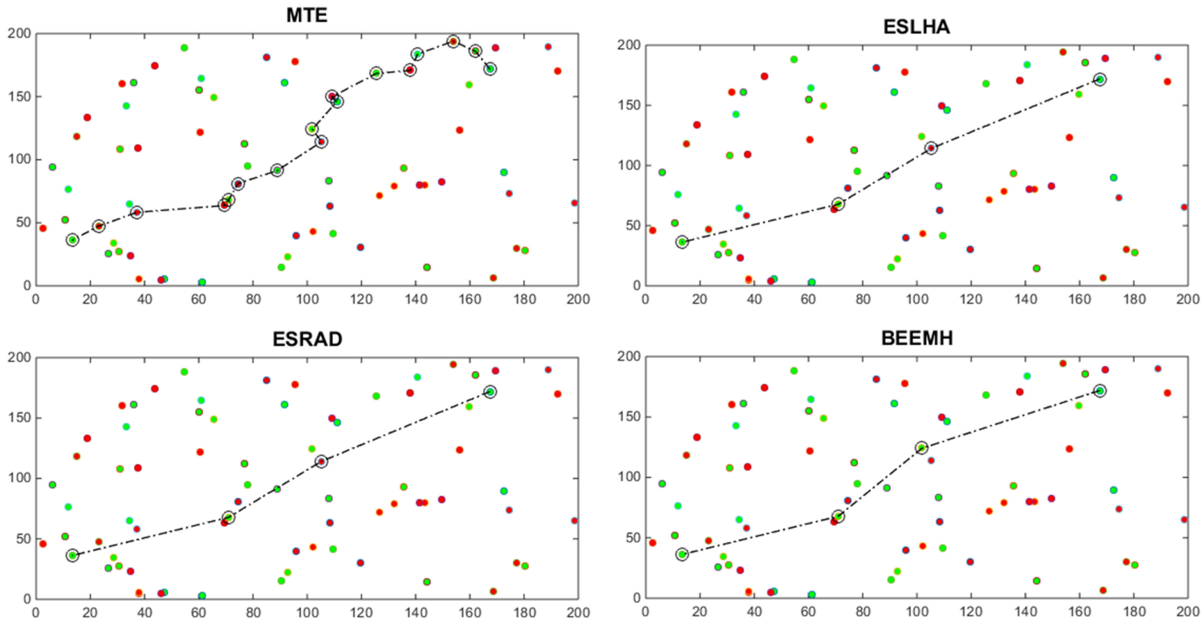

Fig. 5 *Selected paths in WSN-II*

longer paths. Both ESLHA and ESRAD try to take the least-cost path regardless the energy level of nodes. On the other hand, BEEMH algorithm uses the high-energy nodes that exist on the way to the target nodes. In small area network such as WSN-I, the distance between the source and the destination is smaller than d_{char} (equals 104.4 [25]), hence all algorithms except MTE use DT as shown in Fig. 4.

(2) *Evaluation metrics:* The evaluation metrics of the selected paths such as the hop counts, path length, path cost ratio, and the average residual energy are shown in Fig. 8. MTE is a greedy algorithm where each node sends data to its nearest neighbours

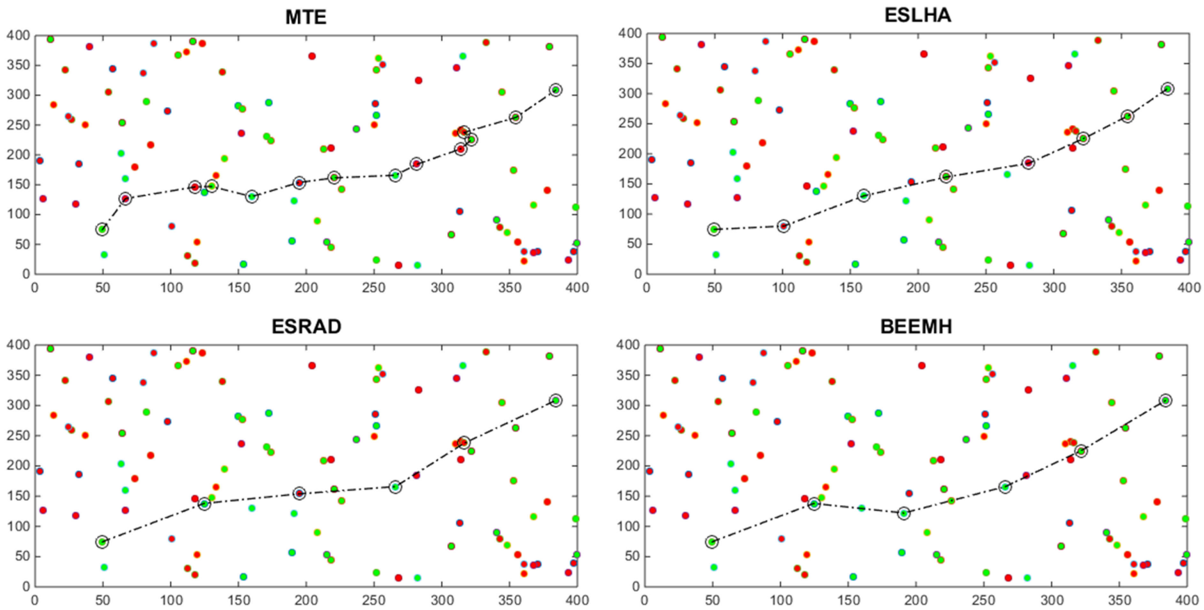

Fig. 6 *Selected paths in WSN-III*

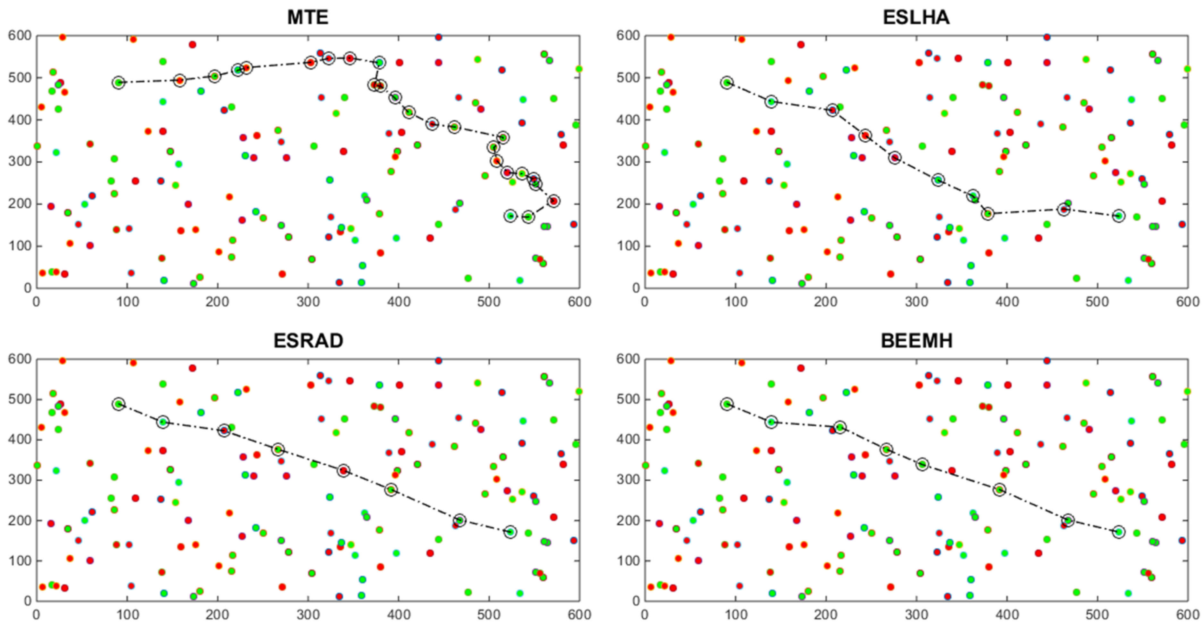

Fig. 7 *Selected paths in WSN-IV*

regardless the overall large cost or the residual energy of nodes through the selected path. Hence, for all network scenarios, MTE always selects the larger number of hops, longer, higher cost and lower residual energy path in comparison with others. As the weight of edges between nodes increases to include the component cost at the TX as in ESLHA or the component cost at both TX and RX as in ESRAD, the paths' hops between nodes become more restricted. Hops number in ESRAD is smaller than these of ESLHA. This is clearly shown in large networks scenarios such as WSN-III and WSN-VI as in Fig. 8*a*. As BEEMH takes all the energy consumption between nodes into consideration, it provides the least-hop path such as ESRAD as shown in all scenarios of Fig. 8*a*.

In low-power transceivers such as sensing nodes the local oscillators and bias circuitry dominate the energy consumption. Hence, long-hop routing yield substantial energy benefits as more distant relay nodes can be reached with sufficient reliability. From Fig. 8*b*, ESRAD always provides the minimum cost paths in all scenarios. Moreover, if nearby neighbours are not used as relays, they can be put into very low-power sleep modes, whereas short-hop routes require many nodes to be awake frequently. Sleep modes provide substantial ES. In a first-order approximation, the control traffic for routing and route maintenance is proportional to the number of nodes in the route. Also, the probability of a route break due to energy depletion and node failure clearly increases with the number of nodes involved. As BEEMH provides the least-hop path than ESLHA as in WSN-III and WSN-IV in Fig. 8*a*, its path cost ratio is lower than that of ESLHA as shown in Fig. 8*b*.

Both ESLHA and ESRAD try to take the least-cost path regardless the energy level of nodes. Otherwise, BEEMH algorithm uses the high-energy nodes that exist on the way to the target node as relay nodes. Hence, its residual energy is greatly larger than that of MTE, ESLHA, and ESRAD as shown in Fig. 8*c*. This, of course, increases the reliability of the selected paths. In small networks or low-density networks, BEEMH may have to select a little bit longer path than ESRAD or ESLHA as in WSN-II and WSN-III in Fig. 8*d* to avoid passing through the low-energy nodes. As the network size or density increases, this problem is reduced as there will be many available options of nodes to be selected. This is clearly shown in WSN-IV in Fig. 8*d*.

From the previous discussion, it is concluded that BEEMH not only selected the minimum cost route, but also it gave great interest to the residual energy of nodes. This minimised the total energy cost and increased the residual energy of the selected path. Hence,

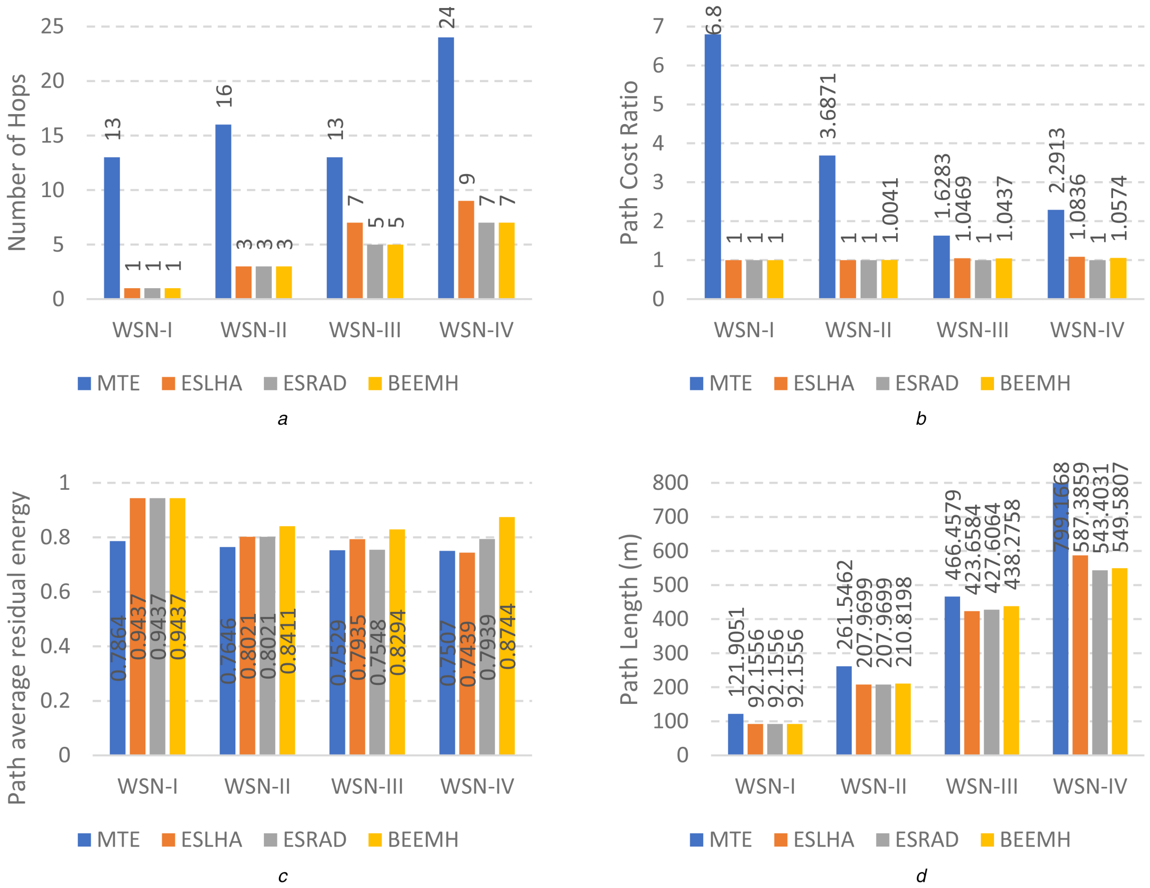

Fig. 8 *Evaluation metrics of the selected paths*
(a) Number of hops, ***(b)*** Cost ratio, ***(c)*** Average residual energy, ***(d)*** Length of paths in the different networks

the reliability of the path has been enhanced. Moreover, the energy consumption has been balanced between nodes. Therefore, the variance of the nodes' residual energy overall the network has been minimised.

Bounds of the running time of BEEMH's algorithm on a graph with edges E and vertices V can be expressed as a function of the number of vertices, denoted $|V|$, using big-O notation as follows:

$$\underbrace{O(|V|+C_o)}_{\text{average energy estimation loop}} + \underbrace{O(C_1\,|V|)}_{\text{initialisation loop}} + \underbrace{O\left(|V|^2\right)}_{\text{main loop}} = O\left(|V|^2\right) \tag{8}$$

where C_o and C_1 are time constants. Since the number of edges of any vertex in a simple, undirected graph will always be less than $|V|$, the rest of the algorithm runs in less than $O\left(|V|^2\right)$ time. So, the complexity of BEEMH is $O\left(|V|^2\right)$ assuming that the first step takes $O(|V|)$ to find the next current vertex. It turns out that selecting the next current can be done in $O(\log|V|)$ time if we use a priority queue for our unvisited set. Using a priority queue, BEEMH's algorithm will run in $O(|V|\log|V|)$ time. This is the same complexity of Dijkstra's algorithm [27].

(3) *BEERAD and BEERAD-E protocols evaluation:* In this section, we are interested in evaluating our proposed protocols; BEERAD and BEERAD-E. Let us assume a heterogeneous network which consists of 100 sensor nodes which are randomly deployed in $400 \times 400\,\mathrm{m}^2$ sensing field. BS is located at (200, 0). Each node is able to generate one packet per round. The rest of simulation parameters are given previously in Table 3. Several metrics can be used to evaluate the routing protocols such as:

Network lifetime: The time interval from the start of the network operation until the death of the last sensor node.

First node die (FND): The number of rounds after which the first sensor node dies.

Half nodes die (HND): The number of rounds after which half of sensor nodes die.

Last nodes die (LND): The number of rounds after which all sensor nodes die.

Stability period: The time interval from the start of network operation until the death of the first sensor node.

Instability period: The time interval from the death of the first sensor node until the death of the last node.

Total residual energy: The total residual energy of all nodes over the network lifetime.

Throughput: The total number of sent packets by nodes to BS.

In several applications, after the death of 50% of nodes, the network becomes useless. Hence, LND parameter could be neglected in comparison with HND.

(1) *Network lifetime:* Fig. 9*a* shows the FND and HND for all protocols. Unlike MTE, ESLHA, and ESRAD protocols, BEERAD and BEERAD-E take the residual energy of nodes into account. Then, the lower-energy nodes are excluded from working as a relay. This balances the energy consumption between nodes. As result, the stability period is improved and the FND of BEERAD is extended by 145, 127, and 121 rounds in comparison with MTE, ESLHA, and ESRAD, respectively. Moreover, The HND of BEERAD exceeds that of MTE, ESLHA, and ESRAD by 123, 39, and 22 rounds, separately. BEERAD-E is more energy aware than BEERAD. Hence, its stability period exceeds that of BEERAD by 19 rounds. However, owing to the large setup cost for updating nodes' tables in BEERAD-E, its HND is reduced by 43 rounds.

Fig. 9*b* shows the number of live nodes over the simulation time. The instability period of MTE, ESLHA, and ESRAD are 180, 244, and 255 rounds, respectively. These periods are higher than that in BEERAD and BEERAD-E which are 144 and 80 rounds, in order. This underscores that some nodes have higher remaining energy than others and last for a longer time than lower ones. This reveals the unbalance energy consumption of the conventional protocols in comparison with our proposed ones. As they did not take the energy of nodes into consideration during path selection.

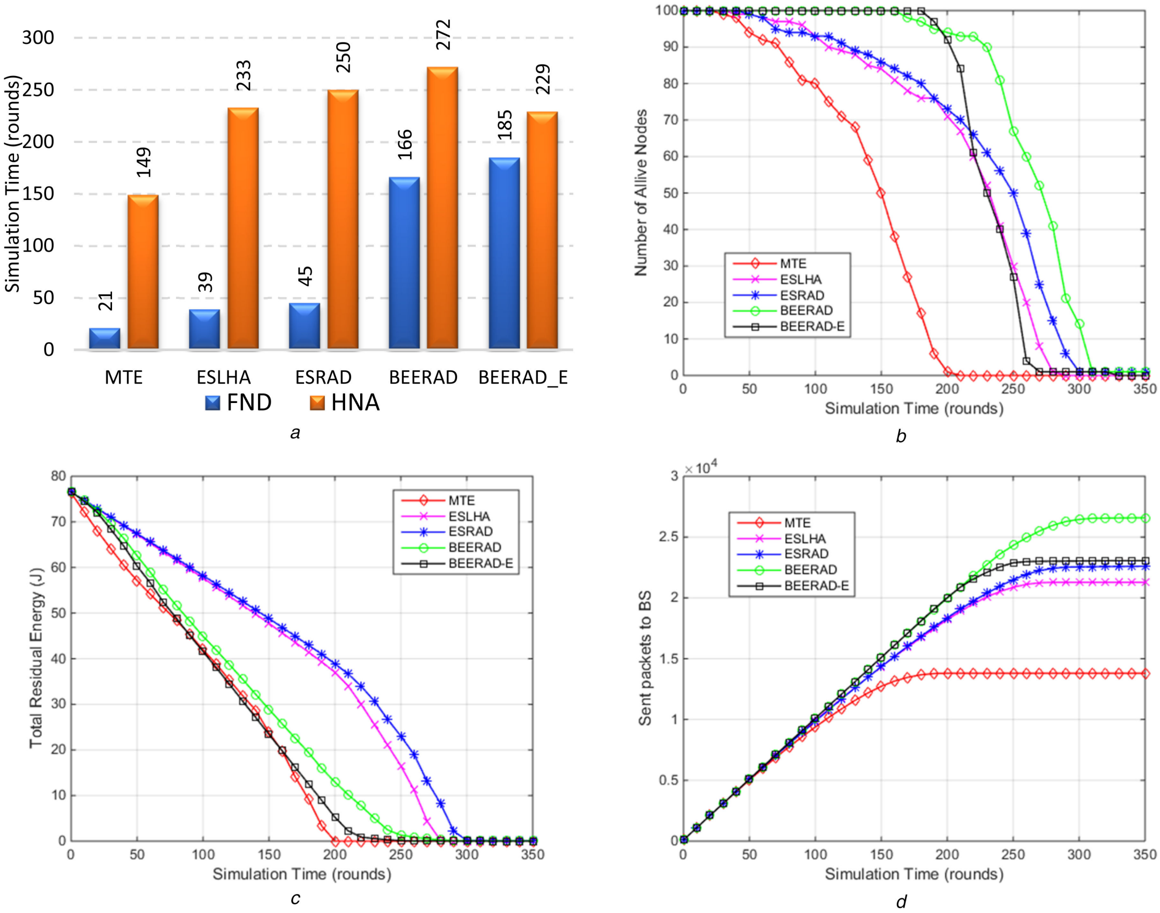

Fig. 9 *Evaluation metrics of the network*
(a) Number of rounds at FND and HND, ***(b)*** Distribution of live nodes over the network lifetime, ***(c)*** Total residual energy of nodes over time, ***(d)*** Number of sent packets to BS

(2) *Residual energy:* Fig. 9*c* shows the total residual energy of nodes over the simulation time. MTE is a greedy protocol where each node in the path to BS searches for its nearest neighbour, regardless the total cost of the path. Hence, the total energy consumption during the steady-state phase increases. This makes the slope of MTE curve is significantly below. ESLHA and ESRAD always look for the least-cost path, regardless the residual energy of nodes. Hence, their curves are higher than MTE, BEERAD, and BEERAD-E. Unlike conventional protocols, our BEERAD and BEERAD-E look for higher-energy nodes to work as a relay. So, they achieve a trade-off between the energy cost and the residual energy of the path. Hence, the slope of residual energy curve is uniform over the network lifetime. Owing to the large cost of BEERAD-E than BEERAD in the setup phase, its curve is lower than that of BEERAD.

(3) *Throughput:* The number of sent packets to BS over the simulation time is shown in Fig. 9*d*. The lifetime of our proposed protocols is larger than others. Hence, a larger number of nodes still able to send their sensed data to BS properly. As result, the curves of BEERAD and BEERAD-E are higher than that in MTE, ESLHA, and ESRAD. Although BEERAD-E has larger stability period than BEERAD, the lifetime of BEERAD in terms of HND and LND is still the largest. Hence, a larger number of nodes live longer in which a higher number of packets are sent to BS.

7 Conclusions

Stability is one of the major concerns in the advancement of WSN. A number of applications of WSN require guaranteed sensing, coverage and connectivity throughout network operational period. The death of the first node might cause instability in the network. One of the major obstacles to ensure network stability is the unbalanced energy consumption rate. This paper proposed BEEMH routing techniques in WSN. At first, BEEMH algorithm for routing in WSN was proposed. BEEMH is a Dijkstra-based algorithm used to estimate long-hop and least-cost paths between nodes. This was done by introducing the total energy consumption as the weight of edges between nodes and use only higher-energy nodes as relays. BEEMH increased the residual energy of the selected path. In the light of BEEMH, two proposed centralised routing protocols: BEERAD and BEERAD-E are introduced. Both of them balanced the energy consumption between nodes and eliminated the energy hole problem. From the simulation results, they improved the stability period of nodes and enhanced the throughput of the network by larger degree. This paper only employs simulation evaluations to discuss the energy efficiency of the BEEMH. However, the application of the improved BEEMH is connected to data collection, data combination, data analysis, and hardware and software design etc. With the support of technology, the practicality and the application boundaries of the BEEMH should be investigated in the test field area. As future extensions to this work, it is interesting to apply BEEMH algorithm in inter-cluster head communications in clustering routing protocols.

8 References

[1] Yick, J., Mukherjee, B., Ghosal, D.: 'Wireless sensor network survey', *Comput. Netw.*, 2008, **52**, (12), pp. 2292–2330

[2] Li, Y., Thai, M.T.: '*Wireless sensor networks and applications*' (Springer Science & Business Media, 2008)

[3] Rault, T., Bouabdallah, A., Challal, Y.: 'Energy efficiency in wireless sensor networks: a top-down survey', *Comput. Netw.*, 2014, **67**, (4), pp. 104–122

[4] Al-Karaki, J.N., Kamal, A.E.: 'Routing techniques in wireless sensor networks: a survey', *IEEE Wirel. Commun.*, 2004, **11**, (6), pp. 6–28

[5] Heinzelman, W.R., Chandrakasan, A., Balakrishnan, H.: 'Energy-efficient communication protocol for wireless microsensor networks'. Proc. 33rd Annual Hawaii Int. Conf. System Sciences, 2000, p. 10

[6] Wang, J., Cho, J., Lee, S., *et al.*: 'Hop-based energy aware routing algorithm for wireless sensor networks', *IEICE Trans. Commun.*, 2010, **93**, (2), pp. 305–316

[7] Ergen, S.C., Varaiya, P.: 'On multi-hop routing for energy efficiency', *IEEE Commun. Lett.*, 2005, **9**, (10), pp. 880–881

[8] Haenggi, M., Puccinelli, D.: 'Routing in *ad hoc* networks: a case for long hops', *IEEE Commun. Mag.*, 2005, **43**, (10), pp. 93–101

[9] Ephremides, A.: ‘Energy concerns in wireless networks’, *IEEE Wirel. Commun.*, 2002, **9**, (4), pp. 48–59
[10] Gomez, J., Campbell, A.T.: ‘A case for variable-range transmission power control in wireless multihop networks’. IEEE INFOCOM 2004, 2004, pp. 1425–1436
[11] Mahmood, D., Javaid, N., Mahmood, S., *et al.*: ‘MODLEACH: a variant of LEACH for WSNs’. 2013 Eighth Int. Conf. Broadband and Wireless Computing, Communication and Applications, Compiegne, 2013, pp. 158–163
[12] Heinzelman, W.B., Chandrakasan, A.P., Balakrishnan, H.: ‘An application-specific protocol architecture for wireless microsensor networks’, *IEEE Trans. Wirel. Commun.*, 2002, **1**, (4), pp. 660–670
[13] Panichpapiboon, S., Ferrari, G., Tonguz, O.K.: ‘Optimal transmit power in wireless sensor networks’, *IEEE Trans. Mob. Comput.*, 2006, **5**, (10), pp. 1432–1447
[14] Murali, P., Rakesh, K., Hota, C., *et al.*: ‘Energy-aware routing in mobile ad-hoc networks’. 2008 First IFIP Wireless Days, Dubai, 2008, pp. 1–5
[15] Sanchez, E.R., Murillo, L.M., Montrucchio, B., *et al.*: ‘An adaptive power-aware multi-hop routing algorithm for wireless sensor networks’. 2011 Eighth Int. Conf. Information Technology: New Generations, Las Vegas, NV, 2011, pp. 112–116
[16] Chiang, S.-S., Huang, C.-H., Chang, K.-C.: ‘A minimum hop routing protocol for home security systems using wireless sensor networks’, *IEEE Trans. Consum. Electron.*, 2007, **53**, (4), pp. 1483–1489
[17] Chuang, P.-J., Jiang, Y.-J.: ‘Effective neural network-based node localization scheme for wireless sensor networks’, *IET Wirel. Sens. Syst.*, 2014, **4**, (2), pp. 97–103
[18] Bachir, A., Dohler, M., Watteyne, T., *et al.*: ‘MAC essentials for wireless sensor networks’, *IEEE Commun. Surv. Tutor.*, 2010, **12**, (2), pp. 222–248
[19] Fawzy, A.E., Amer, A., Shokair, M., *et al.*: ‘Proposed intermittent cluster head selection scheme for efficient energy consumption in WSNs’. 2017 34th National Radio Science Conf. (NRSC), Alexandria, 2017, pp. 275–283
[20] Cormen, T.H.: ‘*Introduction to algorithms*’ (MIT Press, 2009)
[21] Umale, M., Markande, S.: ‘Energy efficient routing algorithm on the target tracking in wireless sensor network’. 2015 Int. Conf. Information Processing (ICIP), Pune, 2015, pp. 176–181
[22] Nelson, D.J., Sayood, K., Chang, H.: ‘An extended least-hop distributed routing algorithm’, *IEEE Trans. Commun.*, 1990, **38**, (4), pp. 520–528
[23] Zhang, L., Yang, W., Rao, Q., *et al.*: ‘An energy saving routing algorithm based on Dijkstra in wireless sensor networks’, *J. Inf. Comput. Sci.*, 2013, **10**, (7), pp. 2087–2096
[24] Azad, A., Kamruzzaman, J.: ‘Energy-balanced transmission policies for wireless sensor networks’, *IEEE Trans. Mob. Comput.*, 2011, **10**, (7), pp. 927–940
[25] Qasem, A.A., Fawzy, A.E., Shokair, M., *et al.*: ‘Energy efficient intra cluster transmission in grid clustering protocol for wireless sensor networks’. Wireless Personal Communications, 2017, pp. 1–18
[26] MATLAB and Statistics Toolbox Release 2014b, The MathWorks, Inc., Natick, MAs, USA
[27] Barbehenn, M.: ‘A note on the complexity of Dijkstra's algorithm for graphs with weighted vertices’, *IEEE Trans. Comput.*, 1998, **47**, (2), p. 263