J Supercomput (2017) 73:1168–1188
DOI 10.1007/s11227-016-1801-0

Rendezvous based routing protocol for wireless sensor networks with mobile sink

Suraj Sharma[1] · **Deepak Puthal**[2] · **Sanjay Kumar Jena**[3] · **Albert Y. Zomaya**[4] · **Rajiv Ranjan**[5,6]

Abstract In wireless sensor networks, the sensor nodes find the route towards the sink to transmit data. Data transmission happens either directly to the sink node or through the intermediate nodes. As the sensor node has limited energy, it is very important to develop efficient routing technique to prolong network life time. In this paper we proposed rendezvous-based routing protocol, which creates a rendezvous region in the middle of the network and constructs a tree within that region. There are two different modes of data transmission in the proposed protocol. In Method 1, the tree is directed towards the sink and the source node transmits the data to the sink via this tree, whereas in Method 2, the sink transmits its location to the tree, and the source node gets the sink's location from the tree and transmits the data directly to the sink. The proposed protocol is validated through experiment and compared with the existing protocols using some metrics such as packet delivery ratio, energy consumption, end-to-end latency, network life time.

Keywords Mobile sink · Mobility management · Energy efficiency · Wireless sensor networks · Routing

---

✉ Suraj Sharma
suraj@iiit-bh.ac.in

[1] Department of Computer Science and Engineering, IIIT Bhubaneswar, Khurda, India

[2] CSIRO Data61, Sydney, Australia

[3] Department of CSE, National Institute of Technology Rourkela, Rourkela, India

[4] School of Information Technologies, The University of Sydney, Sydney, Australia

[5] School of Computing Science, Newcastle University, Newcastle upon Tyne, United Kingdom

[6] School of Computer Science, China University of Geosciences, Beijing, China

1 Introduction

Routing technique plays a vital role in the wireless sensor network. It is extremely difficult to assign the global *ids* for a large number of deployed sensor nodes. Thus, traditional protocols may not be applicable for WSN. Unlike conventional wireless communication networks (MANET, cellular network, etc.), WSN has inherent characteristics. It is highly dynamic network and specific to the application, and additionally it has limited energy, storage, and processing capability. These characteristics make it a very challenging task to develop a routing protocol [1–3]. In most of the scenarios, multiple sources are required to send their data to a particular base station. The nodes near to the sink deplete more energy and hence eventually die. This causes partitioning of the network; consequently, the lifetime of the network gets to reduce. This phenomenon is known as hotspots [5] or energy hole problem [6]. A mobile sink is used in the network to overcome this problem [4,23]. The network with mobile sink implicitly balances the load among the sensor nodes and reduces the chance of hotspots [5]. It can help to achieve the uniform energy consumption and prolong the lifetime. On the other hand, some problems are associated with the mobile sink. The mobile sink is frequently required to send its current position information across the network. This process causes a significant energy consumption overhead. In addition to that, the mobile sink makes the sensor network dynamic in nature. Hence, it is not feasible to find the routing path prior to its requirement. Generally, in the reactive routing, the end-to-end latency is high, which can compromise the requirement of fresh data. In the event-based application, the validity of the sensor data depends on its freshness. The delayed data are of no use. So the primary requirement of the event-based application is to reduce the end-to-end latency. Latency may be affected by many factors like availability of routing path, known mobile sink location, the existence of non-interference paths, etc.

It has been observed that the rendezvous-based approaches are suitable for the time-sensitive applications. They are capable of reducing the latency. Some examples of these time-sensitive applications would be intruder detection systems for building security, target tracking, telemonitoring of human health status, and smoke/CO detection system. In the mobile sink environment, the source node has to wait until it gets the routing path to transmit the data. In rendezvous-based routing some predefined area is specified, where the source node can communicate. In some approaches like line-based data dissemination (LBDD) [7] and grid-based energy efficient routing [8] source node can transmit the data to the rendezvous region, and the rendezvous nodes can further forward the data to the sink, whereas, in the approaches like railroad [9] and ring routing [10], source node can retrieve the current position of the sink from the rendezvous region and transmit the data directly to the sink through intermediate nodes using geographical based approaches [11,12]. In the first type of approaches, the end-to-end latency is very less, but it compromises the energy-efficiency, whereas the second type of approaches are energy-efficient, but compromise the latency, which thus motivates proposing rendezvous-based routing protocol, which can be energy-efficient and takes less time to deliver the sensed data.

In this paper, a rendezvous-based routing protocol (RRP) is proposed, which addresses the requirement of energy-efficiency and less end-to-end latency. In RRP, a virtual cross area is created in the middle of the network. It is called rendezvous region, and the nodes belonging to this region are called backbone nodes. A tree is formed within the rendezvous region, and each sensor node can communicate with the rendezvous region. In RRP, two methods are proposed for the data transmission. In the first method, the source node transmits the data to the sink through the rendezvous region. In the second method, source node retrieves the position of the sink and transmits the data to the sink using geographical based approach [11].

The rest of the paper is organized as follows: related work is discussed in Sect. 2. The system model is defined in Sect. 3, In Sect. 4 the description of the proposed protocol is presented. The simulation result and analysis are discussed in Sect. 5 and finally, the proposed protocol is summarized in Sect. 6.

2 Related works

Energy-efficient data routing is a major research challenge in wireless sensor networks. Researchers are working in this area from long time, still it has research gap when it comes to different kinds of applications. Here in this section we describe the background study of data transmission in sensor networks for different applications, followed by existing solutions for mobile sink sensor network.

Khalid et al. in [17] give the thorough review of the wireless sensor network applications and associate security issues. Where authors highlighted the requirements of sensor networks applications. In [18] Chen et al. proposed new method by considering the natural disasters as the source of applications by highlighting their merits in (1) low cost, (2) quick response, and (3) scalability and flexibility. Then they design an early warning system for geohazards in reservoir region with focuses on issues of (1) supporting reliable data transmission, (2) handling huge data of heterogeneous sources and types, and (3) minimizing energy consumption. Our two proposed protocols follow the same way by focusing different applications. Jan et al. [22] improved the lifetime of sensor network using LEACH-based protocols and efficiently utilizing the limited energy available in these sensor nodes. In [19] Jan et al. proposed a priority-based application-specific congestion control clustering (PASCCC) protocol, and their focus is on integration of the mobility and heterogeneity of the nodes in the network. The authors proposed a scheme whereby a small number of high-energy nodes gather location information and residual energy status of the sensing nodes and transmit to the Base Station [20]. They also proposed a method to avoid the most common Sybil attack over the exiting technique [21,32].

The recent trend of sensor data touching a new trend is called as big data. Puthal et al. [25,26] proposed the solution to handle the big data stream when source of the data is sensor networks and also proposed the solutions to avoid the possible security attacks. Here authors proposed security solutions to deal with big data stream by reducing secret key length. Again in [27,28], authors improved the performance of sensor-generated big data stream by making the secret key length as dynamic one.

The concept of virtual infrastructure acts as a rendezvous area for storing and retrieving the collected data. The sensor nodes belonging to the rendezvous area are designated to store the generated measurements during the absence of the sink. Once, the mobile sink crosses the network, the selected nodes are queried to report the sensory input. A number of routing protocols [7–10,13] have been invented for the mobile sink.

Hamida et al. [29] proposed data dissemination protocols towards mobile sinks in wireless sensor network. The authors analysed sink mobility, energy consumption, and its impact in network lifetime. They presented a line-based data dissemination (LBDD) protocol for mobile sink network end evaluated through empirical studied [7]. Xing et al. [24] proposed a rendezvous-based approach in which a subset of nodes serves as the rendezvous points (RPs) that buffer data originated from sources and transfer to mobile elements when they arrive. Their design is for rendezvous-based data collection protocol that facilitates reliable data transfers from RPs to MEs in the presence of significant unexpected delays in ME movement and network communication. Ekici et al. [30] summarised the existing proposals that use mobility in WSNs and later proposed a new approach to compute mobile platform trajectories. We have highlighted the three more relevant work close to our proposed protocols with more details as follows and compare our proposed method performance with them with simulation results.

Hamida et al. [7] have proposed a line-based data dissemination protocol (LBDD). It defines a virtual horizontally centred line, which divides the sensor field into two equal parts. This line is also divided into groups. This line acts as a rendezvous region for data storage and looks up. This virtual line is placed in the centre of the field to make it accessible by each node. The nodes within the virtual line are called inline-nodes, and the rest of the nodes are called ordinary nodes. When an ordinary node generates a new data, it transmits the data towards the virtual line. The inline-node stores the data and waits for the sink query. The sink transmits a query towards the virtual line in the horizontal direction. The inline-node that receives the query disseminates it in both the directions in the virtual line. When the storing inline-node receives the query, it directly sends the data to the sink.

Shin et al. [9] have proposed railroad protocol, which constructs a virtual structure, called the rail, that is placed in the middle area of the network. It is a closed loop of a strip of nodes, shaped to reflect the outline of the network. The nodes inside the rail are called rail-nodes. At the centre of the rail, the stations are construed by rail-nodes. When a source node generated the data, it sends information about the data, called metadata, to the nearest rail node. This message travels within the rail until it reaches the rail-nodes that store the relevant source node information. The metadata is shared among the nodes on the station. The sink queries the rail for metadata, and when the query reaches a station node, it informs the source about the sinks position, and data are forwarded directly to the sink. In railroad, the sink's queries travel on the rail by unicast rather than broadcasts.

An energy-efficient routing protocol, called ring routing, has been proposed by Tunca et al. [10]. It establishes a ring structure that aims to combine the easy accessibility of the grid structures and the easy changeability of the back- bone structure. Since it incorporates a minimal number of nodes in the ring structure, the redundancy of data packets is significantly reduced for sharing sink position advertisement packets among the ring nodes. It devises a straightforward and efficient mechanism. The ring can be constructed with low overhead unlike the structures utilized in the area-based approaches as in LBDD and Railroad. On the other hand, ring routing relies on the mini- mum amount of inefficient broadcasts which are extensively used in area-based protocols.

The drawback of the above protocols is the mobility management cost and the end-to-end delay. To solve this problem, in this paper a routing protocol have been proposed, where a rendezvous area is defined in the network for data communication.

3 System model

3.1 Energy model

The total energy consumption by the sensor node in the network is derived and used in the implementation of the proposed protocol. The transmitting and receiving energy cost for k bits over the distance of d metres are $E_{\mathrm{TX}}(k, d)$ and $E_{\mathrm{RX}}(k)$, respectively. The derivations of $E_{\mathrm{TX}}(k, d)$ and $E_{\mathrm{RX}}(k)$ are illustrated in Eqs. (1) and (2).

$$E_{\mathrm{TX}}(k, d) = E_{\mathrm{elec}} \times k + E_{\mathrm{amp}} \times k \times d^{\gamma} \tag{1}$$

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

where E_{elec} is the energy cost of the embedded circuit to transmit or receive a signal of one bit, and E_{amp} denotes the energy consumption of the amplifier to maintain the radio for reliable transmission. By using the free space propagation model [14] the energy cost on amplifier E_{amp} is referred as:

$$E_{\mathrm{amp}} = \varepsilon_{\mathrm{fs}}, \tag{3}$$

where $\varepsilon_{\mathrm{fs}}$ is the energy cost of the amplifier to transmit one bit at an open space (one-hop), and γ is the path-loss-exponent and the value of $\gamma \in \{2, 4\}$ [15].

If the distance between the transmitter and recipient is d metres and threshold value of the distance is d_0, then

$$\gamma = \begin{cases} 2 & \text{if}, d \le d_0 \\ 4 & \text{if}, d > d_0 \end{cases} \tag{4}$$

d_0 can be denoted as

$$d_0 = \sqrt{\frac{\varepsilon_{\mathrm{fs}}}{\varepsilon_{\mathrm{mp}}}} \tag{5}$$

Here $\varepsilon_{\mathrm{mp}}$ is the energy cost of the amplifier to transmit one bit at multi-hop model. Using Eq. (3) to (5), Eq. (1) can be rewritten as

$$E_{\mathrm{TX}}\,(k, d) = \begin{cases} E_{\mathrm{elec}} \times k + E_{\mathrm{amp}} \times k \times d^2 & if, d \le d_0 \\ E_{\mathrm{elec}} \times k + E_{\mathrm{amp}} \times k \times d^4 & if, d > d_0 \end{cases} \tag{6}$$

The energy spent by the sensor node in the sleep mode is

$$E_{\text{sleep}}(t) = E_{\text{low}} \times t, \tag{7}$$

where E_{low} is the energy consumption of any node in sleep mode for one second. The total time spent in the sleep mode is t seconds. So the total energy consumption by a sensor node in the network is

$$E_{\text{Total}} = E_{\text{TX}}(k, d) + E_{\text{RX}}(k) + E_{\text{sleep}}(t) \tag{8}$$

3.2 Assumptions

The following assumptions are considered for the proposed protocol:

- Sensor nodes are all stationary after deployment.
- The sink is moving within the network.
- The sensors are randomly deployed in the network field with uniform distribution.
- The base station (sink) possesses unlimited memory, computation and battery power.
- Each node possesses its *id* and can calculate the residual energy.
- Sensor nodes are homogeneous and have the same capabilities.
- Sensor nodes have limited energy.
- Links are symmetric, i.e., the data speed or quantity is the same in both directions, averaged over time.

3.3 Network model

The network consists of n number of sensor nodes and a sink. The sensor nodes are static, and a sink is moving within the network with the speed varying from 5 to 30 m/s. A pause time (δ) for the sink is considered to collect the data. A virtual horizontal and vertical region of width w is considered. It resides in the middle portion of the network having a centre position (u, v). This region has four parts: (i) horizontal left h_l, (ii) horizontal right h_r, (iii) vertical up v_u and (iv) vertical bottom v_b as shown in Fig. 2a. If the sensor node is detected any event, then it should report to the sink. The Random Waypoint mobility model [16] has been considered for the sink mobility. The sensor node can find their location information, and the node can vary their transmission range up to the maximum range R. The threshold energy is the minimum residual energy of a sensor node, beyond which it cannot perform any additional functions except sensing and relaying the data.

3.4 Performance metrics

The efficacy of the proposed protocol has been demonstrated by using the standard performance metrics like control packet overhead, energy consumption, end-to-end latency, packet delivery ratio and network lifetime.

- *Control packet overhead* It is the energy consumption at each sensor node due to the transmission and reception of control packets. These packets are not data. The control packets are used in neighbour discovery, route construction, cluster formation, maintenance process, and so on. This metric is called an overhead because the packet transmission and reception, other than data, is a burden to the network.
- *Energy consumption* It is the total energy consumption at each sensor node due to transmitting, receiving, listening, processing and sleeping. The routing protocol computes the energy consumption based on the energy model. This metric indicates as to how efficiently a protocol works in the network.
- *End-to-End Latency* The end-to-end latency is measured as the time taken for a data packet to transmit over a network from source to sink. It considers all types of delay such as queuing delay, route discovery delay, processing delay and so on. This metric indicates the robustness of the routing protocol.
- *Packet delivery ratio* It is measured as the ratio of the data packet received at the sink to the data packet sent by the sensor nodes. It defines the successful delivery of the data. The protocol with the better delivery ratio is considered to be consistent. This metric also signifies the reliability of the routing protocol.
- *Network lifetime* This metric indicates the duration for which the sensor network is fully functional. It depends on different applications. The lifetime of the network can be a time span when the first sensor dies, a percentage of sensors die, the network partitions, or the loss of coverage occurs. In this paper, the network lifetime is the time span when the sensor network is partitioned into two or more networks and some of the nodes cannot send their sensed data to the sink. From the perspective of the network layer, the control packets are exchanged for route discovery, establishment, and maintenance reflected the routing overhead,which directly affects the network lifetime.

4 The proposed protocol

The proposed protocol is a rendezvous-based routing protocol. In this, a virtual cross area is created of width w, in the middle region of the network. These cross area acts as a rendezvous region for sensor node communication. The nodes in rendezvous area are called backbone node. A tree has been created in the cross area. This tree involves only a few backbone nodes, and it is created such a way that the boundaries can be covered. The tree nodes are responsible to forward the information from the source to the sink or from the sink to the source. The proposed protocol consists of various phases such as neighbour discovery, cross area formation, tree construction, sensor node region discovery and data transmission.

4.1 Neighbour discovery

In this phase, each sensor node finds the neighbours information as discussed in the Algorithm 1. The initiator node broadcasts a control packet NBR_DET, which contains the node *id*, residual energy and the location information. The neighbour node that receives the NBR_DET packet will maintain a table, called *NbrTable*. The *NbrTable* consists of node *id* of the sender, its residual energy and location. If the sender node *id* is already in the *NbrTable*, then the packet is dropped by the receiver node. The receiver node creates and broadcasts the NBR_DET control packet if it did not broadcast before. At the end of the neighbour discovery phase, each node has the one-hop neighbour list and corresponding information.

```
Algorithm 1 Neighbour Discovery Data structure for any sensor node $x$ :
$Nbr(x)$: neighbor set of node $x$, initialized to $\phi$.
$NbrTable(x)$: neighbor table of node $x$, initialized to $\phi$.
$Er_x$: residual energy of any node $x$.
$NbrDETSent_x$ : set to true when the sensor node $x$ sends NBR_DET packet, initialized to false.
node $x$ receives following packet from node $y$:
  NBR_DET :$< NBR\_DET, id_y, Er_y, Loc_y >$
  if $(y \notin Nbr(x))$ then
    $Nbr(x) \leftarrow Nbr(x) \cup \{y\}$;
    Update $NbrTable(x)$ with $< id_y, Er_y, Loc_y >$;
    if $(NbrDETSent_x == false)$ then
      $NbrDETSent_x \leftarrow true$;
      $l\ rb(NBR\_DET, id_x, Er_x, Loc_x)$;                    ◁ Broadcast NBR_DET packet
    else
      Drop the packet;
    end if else
    Drop the packet;
end if
```

4.2 Cross area formation

The proposed protocol divides the sensor field into equal parts of a vertical, and a horizontal stripe, called cross area, as shown in Fig. 1a. The cross area is independent with respect to network area. In the Fig. 1a the network area is square but it is applicable to any type of sensor network area.

The nodes belonging to the cross area are called the backbone nodes. Let us consider w which is the width of the strip and maximum network area is $(x_{\max}, y_{\max})$. So, w_x and w_y, the horizontal and vertical ranges of the backbone, are defined as shown in Eq. (9):

$$w_x = \left(\frac{x_{\max} - w}{2}\right) to \left(\frac{x_{\max} + w}{2}\right); w_y = \left(\frac{y_{\max} - w}{2}\right) to \left(\frac{y_{\max} + w}{2}\right); \quad (9)$$

If any sensor node belongs to the range of w_x and w_y, it can be labelled as a backbone node. In the protocol, the cross area is used as a rendezvous region. This region works as a communication point for the sensor nodes. The rendezvous region and backbone node in the network are shown in Fig. 1.

4.3 Tree construction

The tree construction is performed inside the rendezvous region. The protocol allows only some of the backbone nodes to take part in the tree construction. The boundary nodes of the four sections of rendezvous region h_r, h_l, v_u, v_b, as shown in Fig. 2a, start the process of tree construction. Each node has the neighbour information that includes *id*, residual energy and the location. The boundary node selects one of its neighbour using the following criteria:

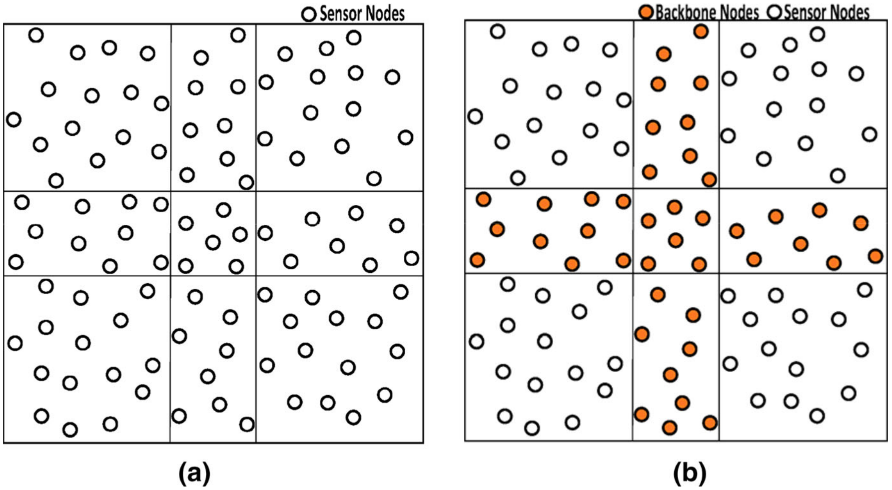

Fig. 1 Rendezvous region and backbone nodes. **a** Initial view of Rendezvous region. **b** Initial view of backbone nodes

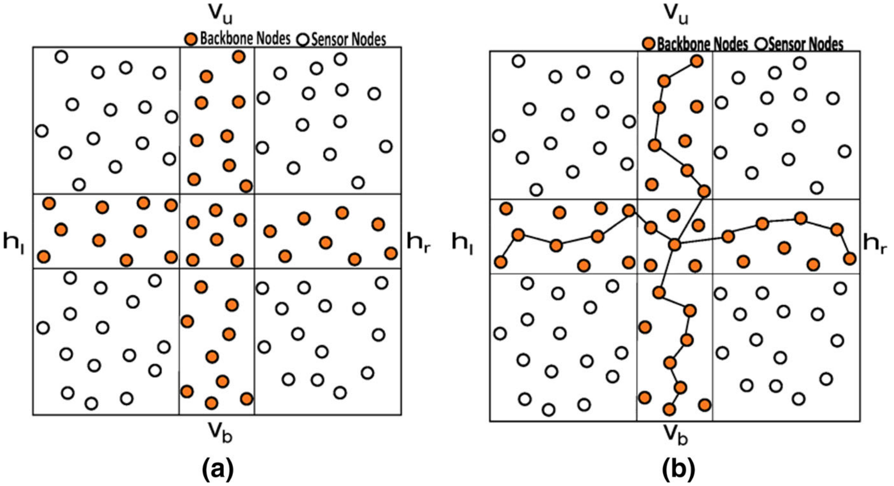

Fig. 2 Rendezvous region and tree within the rendezvous region. **a** Rendezvous region with the boundary *hr, hl*, *vu* and *vl*. **b** Final view of tree construction

```
1.
    The node should be a backbone node,
    Let $BB_x$ is true if any node $x$ labelled as backbone node,
    $Nbr(x)$ is a set of neighbour node of $x$ and $z$ is a sensor node;
    if ($z \in Nbr(x)$ && $BB_z$ == true) then
            $z$ can be chosen by $x$ in tree construction    ◁ First criteria for node selection.
    end if
```

2. the residual energy of the backbone node should be greater than the threshold value,
   Er(z) is the residual energy of any node $z \in Nbr(x)$, then
   Er$(z) \geq E_{\text{threshold}}$ $\triangleleft$ Second criteria for node selection.
3. and the sensor node should be closer to the centroid of the network.

Let $z_1, z_2, z_3, ...z_j$ be the nodes belonging to the backbone and Nbr(x) and $|D(z_i)|$ be the distance of any node z_i from the centroid of the network So, $z = \min_{1 \leq i \leq j}(|D(z_i)|)$ $\triangleleft$ third criterion for node selection.

After selecting one of the neighbour nodes, the boundary node transmits the control packet to the selected node for tree construction. The receiver node makes the sender node as the parent and selects the next neighbour node closest to the centroid. This process repeats until the packet initiated by the boundary node reaches the centroid of the network as shown in Fig. 2b.

```
Algorithm 2 Node Region Discovery
  variables : $\theta = 0$; $\alpha = 0$;
  $(u, v)$: center location of the network.
$(x, y)$: any sensor node location in the network.
Let $\pi = 180^0$;
    $C \leftarrow (u, v)$;                                   ◁ $C$ is the center of the network.
    for any node $z$ in the network with location(x, y)
    Let new coordinates $(A, B) \leftarrow (x - u, y - v)$;   ◁ evaluate $(A, B)$ corresponding to the center $C$.
    Calculate $\theta = tan^{-1}$ |B/A|
    if ($A>0 \&\& B>0$) then
        $\alpha \leftarrow \theta$                             ◁ node with location $(x, y)$ is in $1^{st}$ quadrant.
        if ($\alpha$ lies between $0$ to $\pi/4$) then
            Node with location$(x, y)$ belongs to $1^{st}$ octant and node can communicate from $h_r$ with estination location $(x, v)$.
        else if ($\alpha$ lies between $\pi/4$ to $\pi/2$) then
            Node with location $(x, y)$ belongs to $2^{nd}$ octant and node can communicate from $v_u$ with destination location (u,
            y).
        end if
    end if
    if ($A<0 \&\& B>0$) then
        $\alpha \leftarrow \pi - \theta$                       ◁ node with location $(x, y)$ is in $2^{nd}$ quadrant.
        if ($\alpha$ lies between $\pi/2$ to $3\pi/4$) then
            Node with location$(x, y)$ belongs to $3^{rd}$ octant and node can communicate from $v_u$ with destination location $(u, y)$.
        else if ($\alpha$ lies between $3\pi/4$ to $\pi$) then
            Node with location $(x, y)$ belongs to $4^{th}$ octant and node can communicate from $h_l$ with destination location $(x, v)$.
        end if
    end if
    if ($A<0 \&\& B<0$) then
        $\alpha \leftarrow \pi + \theta$                       ◁ node with location $(x, y)$ is in $2^{nd}$ quadrant.
        if ($\alpha$ lies between $\pi$ to $5\pi/4$) then
            Node with location$(x, y)$ belongs to $5^{th}$ octant and node can communicate from $h_l$ with destination location $(x, v)$.
        else if ($\alpha$ lies between $5\pi/4$ to $3\pi/2$) then
            Node with location $(x, y)$ belongs to $6^{th}$ octant and node can communicate from $v_b$ with destination location $(u, y)$.
        end if
    end if
    if ($A>0 \&\& B<0$) then
        $\alpha \leftarrow 2\pi - \theta$                      ◁ node with location $(x, y)$ is in $2^{nd}$ quadrant.
        if ($\alpha$ lies between $3\pi/2$ to $7\pi/4$) then
            Node with location$(x, y)$ belongs to $7^{th}$ octant and node can communicate from $v_b$ with destination location $(u, y)$.
        else if ($\alpha$ lies between $7\pi/4$ to $2\pi$) then
Node with location $(x, y)$ belongs to $8^{th}$ octant and node can communicate from $h_r$ with destination location $(x, v)$.
end if
end if
```

4.4 Sensor node region discovery

After the tree construction, the sensor node can communicate with the backbone- tree nodes. In this process, the sensor node is required to find out the region in which it belongs so the sensor node can find the shortest destination to communicate with the rendezvous region. The network is virtually divided into octants as illustrated in Fig. 3a.

The sensor node follows Algorithm 2 with the location information of itself and location of the centroid of the network to get the shortest destination.

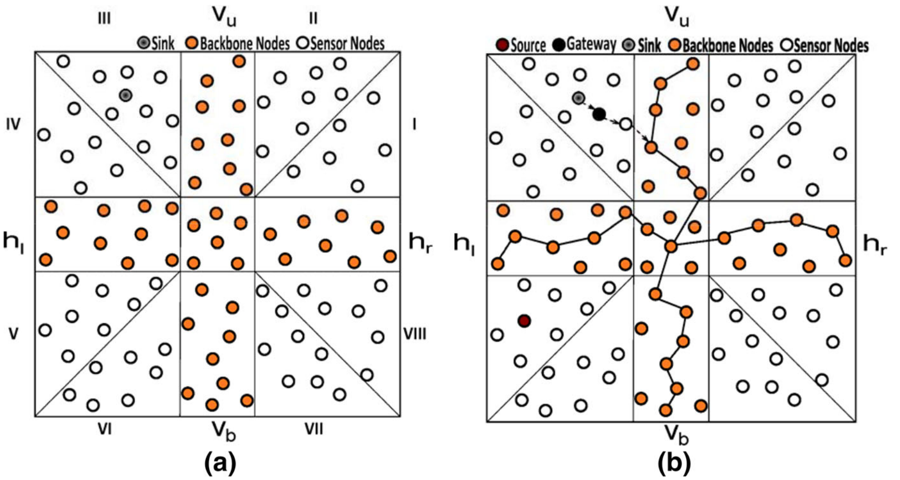

Fig. 3 Sensor node region discovery and gateway node selection. **a** Virtually divided octants region. **b** Gateway node selection

The sensor nodes can calculate the octant in which they belong using their location information (x, y). For example, if the nodes belong to 1st and 8th octant, they will communicate from h_r with destination location (x, v). Similarly, 2nd and 3rd octant sensor nodes can communicate from v_u with destination location (u, y) and so on, where (u, v) are the centre location of the network.

4.5 Data transmission

The sensor node monitors the environment and accordingly generates the data. In the proposed protocol, the source nodes can send the data to the sink whenever required. Two different methods are considered to send the data to the mobile sink. In the first method, the source node transmits the data to the closest backbone-tree node. The backbone-tree node forwards the data to the sink. In the second method, the source node retrieves the sink location from the nearest backbone-tree node and transmits the data directly to the sink by using the sink location. Both the methods are described in the following sections.

4.6 Proposed method 1

4.6.1 Mobile sink management

The sink is moving within the network using the random waypoint mobility model. The mobile sink always moves into the network and pause for a certain time (δ) to collect the data.

When the sink reaches a new position, it selects a gateway node for data collection. The gateway node forwards the ACK packet towards the backbone node through intermediate nodes. Every node that receives the ACK packet first time selects their *next_node* as the preceding node *id* as described in Algorithm-3. This process is shown in Fig. 3b. When the ACK packet reaches the backbone-tree node, it forwards the ACK packet to the rest of the tree. All tree nodes set their *next_node* as preceding node *id* to transmit the data as described in Algorithm 3. This process is depicted in Fig. 4a. The detailed packet communication for sink management is discussed in Algorithm 3. The objective of this phase is to make the reverse link towards the sink for transmitting the data.

```
Algorithm 3 Mobile Sink Management (Proposed Method 1)
Data Structure for any sensor node $x$ and sink:
$Gateway_{sink}$: gateway node selected by the sink.
$Gateway\_selected$: set to true once the sink chooses the gateway node, initialized to false.
$next\_node_x$: the cluster head $x$ selects the next cluster head for data transmission$x$.
$SendData_x$ : set to true once the cluster head chooses the $nex\_ node_x$ for data transmission,
initialized to false.
$BB_x$ : is true if any node $x$ labeled as the backbone node;
$Gateway_x$ : gateway node selected by the node $x$.
---------------------------------------------------------------------------------------------
  Node $x$ receives following packets from the $sink$:
  Beacon: $< \text{Beacon}, id_{sink} >$
  $l\_rf(\text{BeaconReply}, id_x, id_{sink})$;        ◁ Unicast the BeaconReply packet to the sink.

  the sink receives following packets from $y$:

  BeaconReply: $< \text{BeaconReply}, id_y, id_{sink} >$
  if ($Gateway\_selected == false$) then
      $Gateway_{sink} \leftarrow id_y$;
      $Gateway\ selected \leftarrow true$;
      $l\_rf(\text{Gateway}, id_{sink}, Gateway_{sink})$;   ◁ the sink unicasts the Gateway packet to the selected gateway node.
  else
      Drop the packet;
  end if

  node $x$ receives following packets from the $sink$:

  Gateway: $< \text{Gateway}, id_{sink}, Gateway_{sink} >$
  if ($id_x == Gateway_{sink}$) then
      $next\_node_x \leftarrow id_{sink}$;
      As described in the Algorithm 2 the gateway node chooses the backbone and destination location.
      The node forwards the ACK packet to the node $z$ closest to the destination using the Equation
  (11).
      $l\_rf(\text{ACK}, id_x, id_z, Gateway_{sink})$;        ◁ Forwards the ACK packet.
  else
      Drop the packet;
  end if

  node $x$ receives following packets from the node $y \in Nbr(x)$:

  ACK: $< \text{ACK}, id_y, id_z, Gateway_{sink} >$
  if ($id_x == id_z$) then
      if ($Gateway_x \neq Gateway_{sink}$) then
          $Gateway_x \leftarrow Gateway_{sink}$;
          $SendData_x \leftarrow true$;
          $next\ node_x \leftarrow id_y$;
          if ($BB_x == true \&\& Parent_x == true$) then
              Choose the node $z$ as parent and child $id$;
          else
              Choose the node $z$ closest to the destination using the Equation (11).
          end if
          $l\_rf(\text{ACK}, id_x, id_z, Gateway_{sink})$;        ◁ Forwards the ACK packet in the tree.
      else
          Drop the packet;
      end if
  else
      Drop the packet;
end if
```

4.6.2 Data transmission

The sensor node can send their data to the sink through the backbone-tree nodes. The sensor node finds the destination for data transmission using the Algorithm 2. Each sensor node has the neighbour information, which contains the neighbours' location and residual energy. It can easily send the generated data to the backbone-tree node through the neighbour nodes using the location factor (LF) as derived in the Eq. (11). The source node can select the node that has the sufficient residual energy and minimal

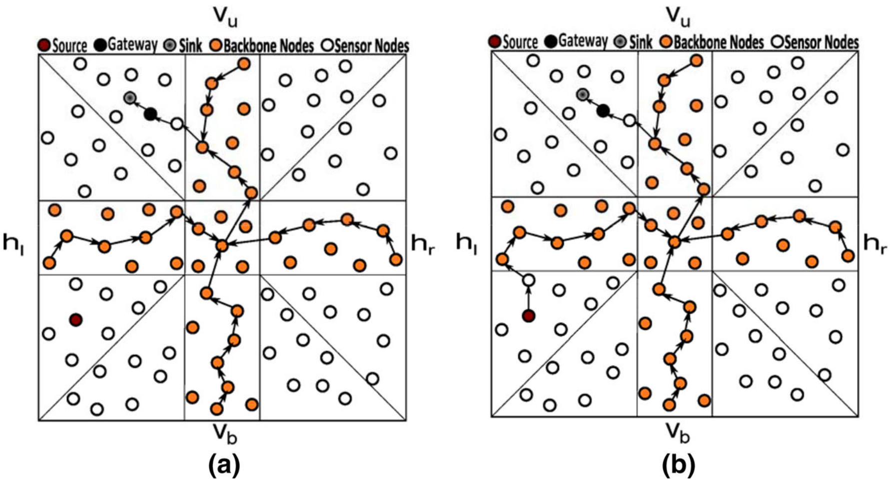

Fig. 4 Data transmission using proposed method 1. **a** Backbone-tree node link directed towards the sink. **b** Data transmission through the backbone-tree nodes

distance from the destination for data transmission. This process is shown in Fig. 4b. In a regular interval, each node broadcasts their residual energy to update the neighbour information.

Let node i be required to select the nodes from its neighbours. Nbr(i) is the set of neighbours of node i, LF (i) is the set of location factors of each member of Nbr(i), *Erk* is the residual energy of node $k \in Nbr(i)$, *(xk, yk)* is the location information of node $k \in Nbr(i)$ and *Dk* is the Euclidean distance from the destination.

$$Let, \mathrm{Er}_{\max} = \max_{k \in Nbr(i)} E_r;$$

then for kth neighbour LF_k can be computed as

$$\mathrm{LF}_k = \hat{\mathrm{E}}r_k \times \frac{1}{D_k} = \frac{\hat{\mathrm{E}}r_k}{D_k} \quad \forall\, k : k \in Nbr\,(i)\,, \tag{10}$$

where

$$\hat{\mathrm{E}}r_k = \frac{\mathrm{Er}_k}{\mathrm{Er}_{\max}}$$

$$D_k = \sqrt{(x_{\mathrm{dest}} - x_k)^2 + (y_{\mathrm{dest}} - y_k)^2}$$

And,

$$nextnode_i = \max(\mathrm{LF}(i)), \tag{11}$$

where next node$i$ is the sensor node selected by the node i.

4.7 Proposed Method 2

4.7.1 Mobile sink management

In the second method of rendezvous-based routing protocol, the sink node informs its position to the backbone-tree nodes. They have the latest location information of the sink as shown in Fig. 5a.

When a sink node moves to a new position, it broadcasts a Beacon packet to get the neighbour information. The sink selects one of its neighbour nodes to forward the location information. Sink refers Algorithm 2 and Eq. (11) to select the forwarding node. The forwarding node again relays the sink's location to its neighbour using the same technique. When location information reaches the backbone-tree node, it disseminates the location information into the tree. The communication details on the sink management are discussed in Algorithm 4.

```
Algorithm 4 Mobile Sink Management (Proposed Method 2)
Data Structure for any sensor node $x$ and sink:
  $Sink\_Loc_x$ : any node $x$ stores the sink location information.
  $BB_x$ : is true if any node $x$ labeled as the backbone node, initialized as false.
$Loc_{sink}$ : the location of the sink.
Beacon: $< Beacon, id_{sink} >$
    $l\_rf(BeaconReply, id_x, Er_x, id_{sink})$;            ◁ Unicast the BeaconReply packet to the sink.
    As described in the Algorithm 2 the sink node chooses the backbone to send its location. The sink node forwards the
    Location packet to the node $z$ using the Equation (11).
    $l\_rf(Location, id_{sink}, Loc_{sink}, next\_node_z)$;     ◁ Unicast the Location packet to the selected node $z$.
    Node $x$ receives following packets from the sink or any node $y$:

    Location : $< Location, id_y, Loc_{sink}, next\_node_y >$
    if $(id_x == next\_node_y)$ then
        if $(Sink\,Loc_x \neq Loc_{sink})$ then
            $Sink\,Loc_x \leftarrow Loc_{sink}$;
            if $(BB_x == true \&\& Parent(x) == true)$ then
                Choose the node $z$ as parent and child $id$;
            else
                Choose the node $z$ closest to the destination using the Equation (11).
            end if
            $l\_rf(Location, id_x, Loc_{sink}, next\_node_z)$;  ◁ Unicast the Location packet to the selected
    node $z$.
        else
            Drop the packet;
        end if
    else
        Drop the packet;
end if
```

4.7.2 Sink location recovery and data transmission

To transmit the data source node needs to find the sink location. It can get the sink location from the backbone-tree nodes. For finding the sink location the source, the node makes a request to the backbone-tree node by sending a Loc_Req packet. When the backbone-tree node receives the request, it replies with the sink location as shown in Fig. 5a. The sink location recovery process is discussed in the Algorithm 5.

After getting the sink location, the source node transmits the data to sink through the neighbour nodes. It selects one of the neighbour nodes having sufficient residual energy and minimum distance from the sink as mentioned in Eq. (11). When the neighbour node receives the data, it selects another node from its neighbour list using the technique as mentioned above. Figure 5b illustrated the data transmission from the source to the sink through intermediate nodes.

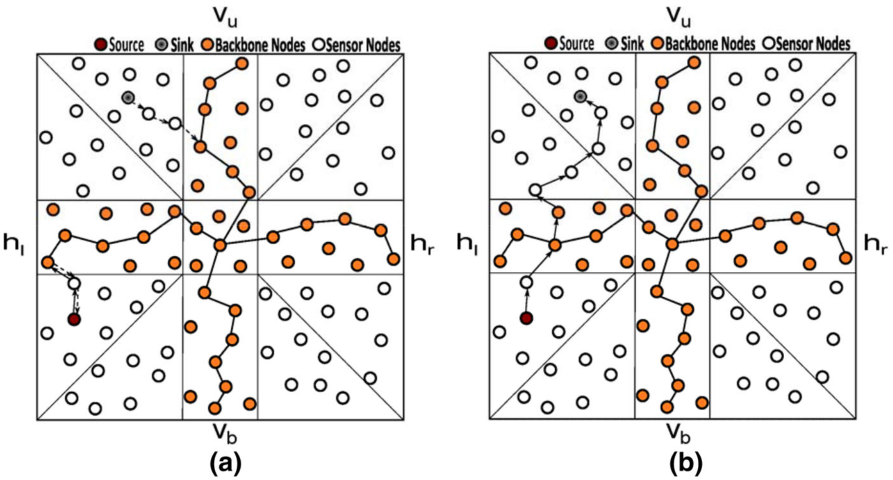

Fig. 5 Data transmission using proposed Method 2. **a** Sink location recovery. **b** Data transmission

```
Algorithm 5 Sink Location Recovery (Proposed Method 2)
Data Structure for any sensor node $x$:
  $Sink\_Loc_x$: any node $x$ stores the sink location information.
  $BB_x$ : is true if any node $x$ labelled as the backbone node;
  $Loc_{sink}$ : the location of the sink.
  $next\_node_x$ : any sensor node $x$ selects the next node for packet transmission .
$reverse\_link_x$ : any sensor node $x$ select the sender for sending sink location
    As described in the Algorithm 2 the source node chooses the destination to send the location request.
    The source node forwards the Loc_Req packet to next node using the Equation (11).
    $l\_rf$(Loc_Req, $id_x$, $next\_node_x$);          ◁ Unicast the Location packet to the selected next node.

    Node $x$ receives following packets from any node $y \in Nbr(x)$:
    Loc_Req : < Loc_Req, $id_y$, $next\ node_y$ >
    if ($id_x == next\_node_y$) then
        $revers\_link_x \leftarrow id_y$;
        if ($BB_x == true \&\& Parent(x) == true$) then
            $l\_rf$(Loc_Reply, $id_x$, $Loc_{sink}$, $reverse\_link_x$);  ◁ Reply the sink location to the requested node.
        else
            The node selects the $next\_node$ using the Equation (11).
            $l\_rf$(Loc_Req, $id_x$, $next\_node_x$);          ◁ Unicast the Loc_Req packet to the next node.
        end if
    else
        Drop the packet;
    end if

    Node $x$ receives following packets from any node $y \in Nbr(x)$:
    Loc_Reply : < Loc_Reply, $id_y$, $Loc_{sink}$, $reverse\_link_y$ >
    if ($id_x == reverse\_link_y$) then
        if ($id_x == id_{source}$) then
            $Sink\_Loc_x \leftarrow Loc_{sink}$;
        else
            $l\_rf$(Loc_Reply, $id_x$, $Loc_{sink}$, $reverse\_link_x$);          ◁ Unicast the Location packet to
    the requested node.
        end if
    else
        Drop the packet;
end if
```

5 Simulation results

Through the simulation, the proposed protocol's performance has been analysed and compared with the existing protocols such as line-based data dissemination (LBDD) [7], railroad [9] and ring routing [10]. Each experiment has been performed with the varying sink speed from 5 to 30 m/s. The impact of the sink speed in energy consumption, end- to-end latency and data delivery ratio has been observed. An extensive

Table 1 Simulation parameters

Parameter name	Value
Network area	$500 \times 500\,\text{m}^2$
Number of sensor nodes	200
Data packet size	512 bytes
Control packet size	32 bytes
Initial energy	1 J
δ	5 s
Sink speed	(5, 10, 15, 20, 25, 30) m/s
Mobility model	Random waypoint
E_{elec}	50 nJ/bit
ε_{fs}	$10\,\text{pJ/bit/m}^2$
ε_{mp}	$0.0013\,\text{pJ/bit/m}^4$
$d0$	87 m
E_{low}	0.2 nJ/s
Simulation time	600 s
MAC protocol	TMAC

set of simulation is performed based on the parameter illustrated in Table 1 using the Castalia (v3.2) simulator.

5.1 Average control packet overhead

The sensor node transmits the control packets to construct the rendezvous region and manage the sink mobility. The average energy consumption of control packet with varying sink speed for various protocols is illustrated in Fig. 6. As shown in the graph, the control packet overhead is very less in the proposed method 2 as compared to the other protocols.

In LBDD, an inline-node stores the data from the source node. When that inline-node receives the query, it sends the data to the sink. The sink's query is flooded into the rendezvous region, which causes an increased control packet overhead. In the railroad protocol, the rail construction and station formation is the one-time process. However, the process of metadata storage at station and retrieval of the sink location from the station requires the control packet exchange. In ring routing, all the ring nodes store the location of the sink. So the retrieval of the sink location is easier. However, as the network operation progresses, it requires the exchange of control packets to repair the ring. So the ring length increases, and as a result, the distance from the source or the sink causes more energy consumption. The proposed method 1 only needs to maintain the tree within the rendezvous region to transmit the data. The control packets are required to set the link according to the sink position. However, the proposed method 2 consumes less control packet overhead. It is because the average distance between rendezvous region and the source or the sink is less than the other protocols.

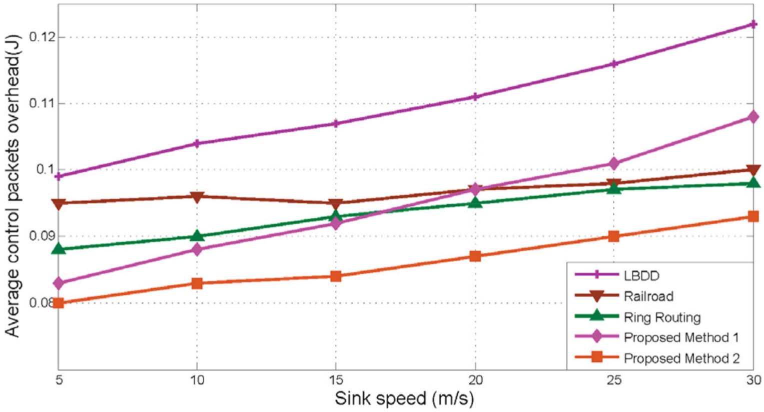

Fig. 6 Control packet overhead

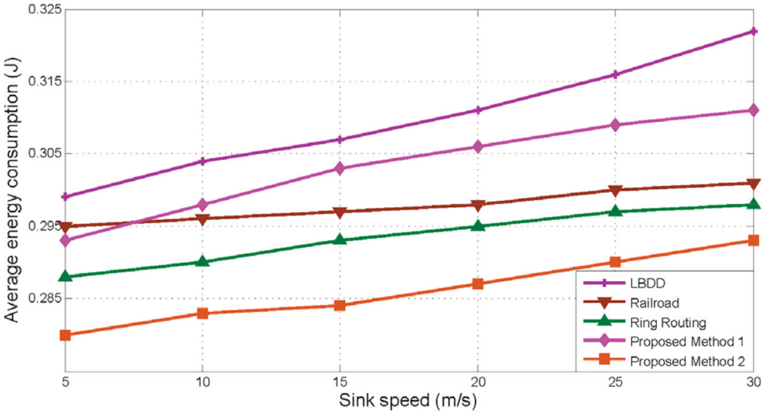

Fig. 7 Average energy consumption

5.2 Average energy consumption

The total energy consumption at each node for various protocols is shown in Fig. 7. It has been observed that the energy consumption of LBDD is highest due to greater control packet overhead. It stores the data from the source node and floods the sink's query in the rendezvous region. The energy consumption of the LBDD grows monotonically as the sink speed increases. The proposed method 1 does not require sink location, but the average path length is higher than railroad, ring routing and proposed method 2. So the overall energy consumption is more and increases according to the sink speed. In the proposed method 2, the average distance between source and the sink is almost the same as the railroad and ring routing. However, due to the less control packet overhead, the proposed method 2 outperforms the existing protocols.

5.3 Average end-to-end latency

Figure 8 presents the average end-to-end latency of different protocols with various sink speeds. It depends on the time duration to find the sink's location and propagate the data to the sink. The proposed method 1 instantly transmits the data to the backbone tree. The tree forwards the data to the sink, as it is always connected with the sink. As a result, the end-to-end delay is very less. However, in the LBDD the inline-node transmits the data as soon as it gets the sink location. The proposed method 2 takes less time to deliver the data as compared to railroad and ring routing. It is due to the shorter distance between the rendezvous region and the source node.

5.4 Packet delivery ratio

Figure 9 illustrates the data delivery ratio of various protocols. It shows the success rate of the data reception at the sink. The proposed method 1 maintains the connection between the tree and the sink. Hence, the delivery ratio is higher than other protocols. In LBDD, the data are stored by the inline-node and transmitted to the sink as soon as it gets the location. So the possibility of data loss is less than the other protocols. In railroad and ring routing the time duration to get the sink's location is higher than that in the proposed method 2. It increases the delay to the data transmission. In that duration, the sink may move to the new location that causes data loss.

5.5 Network lifetime

The energy consumption at each node and imbalance load among the sensor nodes affects the network lifetime. It is clearly shown in Fig. 10 that the network lifetime of the proposed method 2 is greater than that of the other protocols. The reason behind this is that it consumes fewer control packets, balances the load among the sensor nodes and follows an optimal route for data transmission.

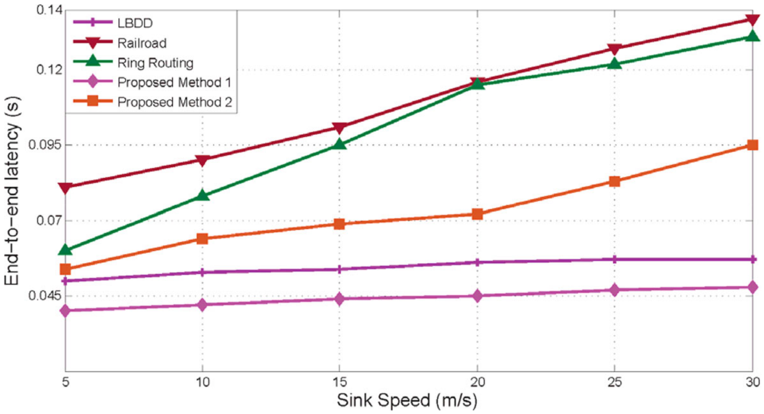

Fig. 8 Average end-to-end latency

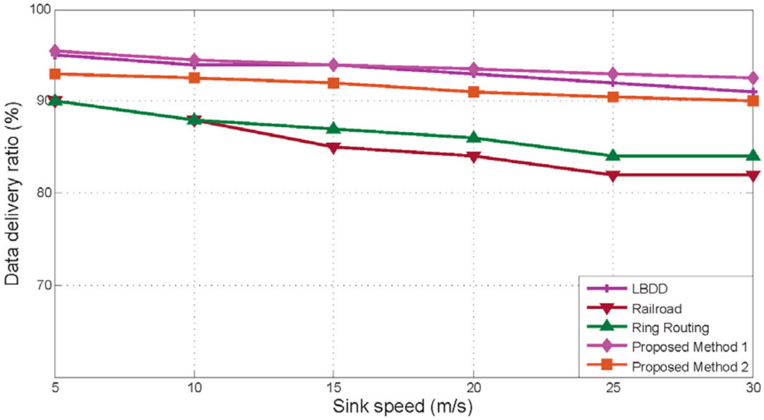

Fig. 9 Packet delivery ratio

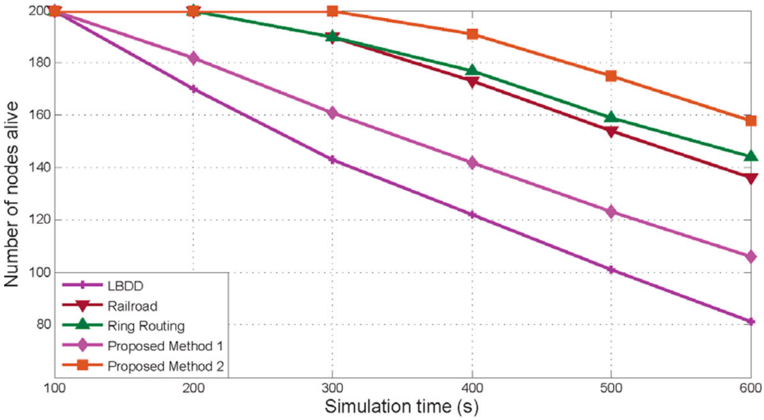

Fig. 10 Network lifetime

6 Conclusion

In this paper, we proposed rendezvous-based routing protocols. It creates a rendezvous region in the middle of the network and constructs a tree within that region. In the proposed protocol, two different methods are used for data transmission. In proposed method 1, the tree is directed towards the sink and source node transmits the data to the sink via this tree, whereas in proposed method 2, the sink transmits its location to the tree, and the source node gets the sink's location from the tree and transmits the data directly to the sink. Both the methods are compared with the existing protocols such as LBDD, railroad and ring routing. From the simulation results, it has been observed that the proposed method 1 outperformed the existing protocols in terms of end-to-end latency and delivery ratio. The energy consumption of the proposed method 2 is very less than the existing protocols.

Acknowledgements This work is the part of PhD thesis work of Dr. Suraj Sharma, which is available online at NIT Rourkela ethesis [31]. Deepak Puthal contributed to this work during his Master,s study at NIT Rourkela.

References

1. Zhang D, Li G, Zheng K, Ming X, Pan Z (2014) An energy-balanced routing method based on forward-aware factor for wireless sensor networks. IEEE Trans. Ind. Inf. 10(1):766–773
2. Bangash J, Abdullah A, Anisi M, Khan A (2014) A survey of routing protocols in wireless body sensor networks. Sensors 14(1):1322–1357
3. Rault T, Bouabdallah A, Challal Y (2014) Energy efficiency in wire-less sensor networks: a top-down survey. Comput. Netw. 67(1):104–122
4. Puthal D, Sahoo B, Sharma S (2012) Dynamic model for efficient data collection in wireless sensor networks with mobile sink. Int J Comput Sci Technol 3(1):623–628
5. Perillo M, Cheng Z (2005) An analysis of strategies for mitigating the sensor network hot spot problem. In: 2nd Annual International Conference on Mobile and Ubiquitous Systems: Networking and Services (MobiQuitous 2005), pp 474–478
6. Li J, Mohapatra P (2007) Analytical modeling and mitigation techniques for the energy hole problem in sensor networks. Pervasive Mob Comput 3(3):233–254
7. Hamida E, Cheliusi G (2008) A line-based data dissemination protocol for wireless sensor networks with mobile sink. In: IEEE International Conference on Communications (ICC 2008), pp 2201–2205
8. Kweon K, Ghim H, Hong J, Yooni H (2009) Grid-based energy-efficient routing from multiple sources to multiple mobile sinks in wireless sensor net-works. In: 4th International Symposium on Wireless Pervasive Computing (ISWPC 2009), pp 1–5
9. Shin J, Kim J, Park K, Park D (2005) Railroad: virtua infrastructure for data dissemination in wireless sensor networks. In: 2nd ACM International Workshop on Performance Evaluation of Wireless Ad Hoc, Sensor, and Ubiquitous Net-works (PE-WASUN 2005), pp 168–174
10. Tunca C, Isik S, Donmez M, Ersoy C (2014) Ring routing: an energy- efficient routing protocol for wireless sensor networks with a mobile sink. IEEE Trans Mob Comput 13(11):1–15
11. Karp B, Kung H (2000) GPSR: greedy perimeter stateless routing for wireless networks. In: 6th Annual International Conference on Mobile Computing and Networking, pp 243–254
12. Xu Y, Heidemann J, Estrin D (2001) Geography informed energy conservation for ad-hoc routing. In: 7th Annual ACM/IEEE International Conference on Mobile Computing and Networking, pp 70–84
13. Luo H, Ye F, Cheng J, Lu S, Zhang L (2005) TTDD: two-tier data dissemination in large-scale wireless sensor networks. Wirel Netw 11:161–175
14. Wendi B, Heinzelman A, Chandrakasan P, Balakrishnan H (2002) An application-specific protocol architecture for wireless microsensor networks. IEEE Trans Wirel Commun 1(4):660–670
15. Wang Q, Hempstead M, Yang W (2006) A realistic power consumption model for wireless sensor network devices. In: IEEE International Conference on Sensing, Communication, and Networking (SECON 2006), pp 286–295
16. Broch J, Maltz D, Johnson D, Hu Y, Jetcheva J (1998) A performance comparison of multi-hop wireless ad hoc network routing protocols. In: 4th Annual ACM/IEEE International Conference on Mobile Computing and Networking, pp 85–97
17. Khalid O, Khan S, Madani S, Hayat K, Khan M, Min-Allah N, Kolodziej J, Wang L, Zeadally S, Chen D (2013) Comparative study of trust and reputation systems for wireless sensor networks. Secur Commun Netw 6(6):669–688
18. Chen D, Liu Z, Wang L, Dou M, Chen J, Li H (2013) Natural disaster monitoring with wireless sensor networks: a case study of data-intensive applications upon low-cost scalable systems. Mob Netw Appl 18(5):651–663
19. Jan M, Nanda P, He X, Liu R (2014) PASCCC: priority-based application-specific congestion control clustering protocol. Comput Netw 74:92–102
20. Jan M, Nanda P, He X (2013) Energy evaluation model for an improved centralized clustering hierarchical algorithm in WSN. In: WWIC, pp 154–167
21. Jan M, Nanda P, He X, Liu R (2015) A sybil at-tack detection scheme for a centralized clustering-based hierarchical network. Trust-com/BigDataSE/ISPA, IEEE, vol 1, pp 318–325

22. Jan M, Nanda P, He X, Liu R (2013) Enhancing lifetime and quality of data in cluster-based hierarchical routing protocol for wireless sensor network. In: 10th International Conference on High Performance Computing and Communications, pp 1400–1407
23. Puthal D (2012) Secure data collection and critical data transmission technique in mobile sink wireless sensor networks. M.Tech Thesis, National Institute of Technology, Rourkela
24. Xing G, Wang T, Xie Z, Jia W (2008) Rendezvous planning in wireless sensor networks with mobile elements. IEEE Trans Mob Comput 7(12):1430–1443
25. Puthal D, Nepal S, Ranjan R, Chen J (2015) DPBSV-An efficient and secure scheme for big sensing data stream. Trustcom/BigDataSE/ISPA, IEEE, vol 1, pp 246–253
26. Puthal D, Nepal S, Ranjan R, Chen J (2016) A dynamic prime number based efficient security mechanism for big sensing data streams. J Comput Syst Sci http://dx.doi.org/10.1016/j.jcss.2016.02.005
27. Puthal D, Nepal S, Ranjan R, Chen J (2015) A dynamic key length based approach for real-time security verification of big sensing data stream. In: Web information systems engineering-WISE 2015. Springer, Berlin, pp 93–108
28. Puthal D, Nepal S, Ranjan R, Chen J (2016) DLSeF: a dynamic key length based efficient real-time security verification model for big data stream. ACM Trans Embed Comput Syst
29. Hamida E, Chelius G (2008) Strategies for data dissemination to mobile sinks in wireless sensor networks. IEEE Wirel Commun 15(6):31–37
30. Ekici E, Gu Y, Bozdag D (2006) Mobility-based communication in wireless sensor networks. IEEE Commun Mag 44(7):56–62
31. Sharma S (2016) On energy efficient routing protocols for wireless sensor networks. PhD dissertation, National Institute of Technology, Rourkela
32. Jan M, Nanda P, He X, Liu R (2016) A sybil attack detection scheme for a forest wildfire monitoring application. Future Gener Comput Syst