Research Article

Data Dissemination Protocol for Mobile Sink in Wireless Sensor Networks

Suraj Sharma[1] and Sanjay Kumar Jena[2]

[1] *International Institute of Information Technology, Bhubaneswar 751003, India*
[2] *National Institute of Technology, Rourkela 769008, India*

Correspondence should be addressed to Suraj Sharma; suraj.atnitrkl@gmail.com

In wireless sensor networks, the sensor nodes find the route towards the sink to transmit the data. The sensor node transmits the data directly to the sink, or it relays the data through neighbor nodes. The nodes near to the sink transmit more data than other nodes. It results in the small lifetime of the network. To prolong the lifetime of the network, we use the mobile sink approach. The mobile sink makes the network dynamic. It is a challenging task to find the route in the dynamic network. In this paper, we have proposed a distributed tree-based data dissemination (TEDD) protocol with mobile sink. The protocol is validated through simulation and compared with the existing protocols using some metrics such as energy consumption, average end-to-end delay, and throughput. The experiment results show that the proposed protocol outperforms the existing protocols.

1. Introduction

Sensor network is a multihop network which consists of hundreds of sensor nodes. The main resource constraint of the sensor node is the energy. Generally, sensor networks are deployed in the unattended and hostile environment such as wildlife detection, continuous environment monitoring, and military. So it is impossible to replace or recharge the battery. The main goal of the proposed paper is to develop the energy-efficient protocol to prolong the lifetime of the network.

In the sensor network, the work of the sensor node is not only to sense environmental data, but also to relay those data to the sink. Sink is a resource-rich node, whose responsibility is to collect the sensed data from the sensor nodes and send it to the user via the Internet. Sensor node has constraints of limited communication range, which does not allow direct communication between source and sink. It relays the data to the sink in the multihop manner.

The sensor nodes close to the sink transmit more data than the other nodes in the network. That is why they depleted their energy and died. This may result in the partition of the network. This situation is called "crowded center effect" [1] or "energy hole problem" [2]. The energy hole problem can be overcome by using the mobile sink in the network. The mobile sink moves across the network and collects the data from the sensor nodes. The movement of the sink may be random, controlled, or predefined. The mobile sink makes the network dynamic. So the data dissemination protocols for network with static sink are unsuitable for the network with the mobile sink. It is a challenge to develop energy-efficient data dissemination protocols for the mobile sink.

In this paper, we have proposed a tree-based energy-efficient data dissemination protocol. In this protocol, any sensor node can disseminate the data to the sink via tree.

The rest of the paper is organized as follows. Related work is discussed in Section 2. We describe the working principle and algorithm of the proposed model in Section 3. In Section 4, we discussed the experimental setup, energy model for sensor nodes, mobility model for sink, and performance metrics to evaluate the protocol. The simulation result and analysis are described in Section 5.

2. Related Work

A number of data dissemination protocols have been invented for the mobile sink. These data dissemination protocols are broadly classified as single-hop or multihop.

In the single-hop data dissemination protocol, distance between the source and the sink is one-hop. Khan et al. [3] and Sudarmani and Kumar [4] have proposed cluster-based approach, where a cluster head sends aggregated data to the mobile sink. In flat structure schemes [5–7], mobile sink broadcasts the small beacon packets periodically while it moves to the new location. The sensor nodes in its range receive the beacon and transmit data to the sink. In all single-hop schemes, data deliveries are more reliable but increase the latency (delay). These types of schemes are suitable for the delay tolerant applications [8–10].

The multihop data dissemination protocols have been proposed by many researches [11–16]. Virtual grid-based, geographical-based, cluster-based, flat, proactive routing-based, and tree-based protocols are the examples of multihop data dissemination protocols. In virtual grid-based protocols [11, 17], network is partitioned among the number of grids, which consists of a limited number of sensor nodes and a head node. The head node is responsible for relaying the data to the mobile sink. Lee et al. [18] have proposed a grid-based two-tier data dissemination model, which discovers the routing path from each source to the sink in mobile sink environment. In geographical routing, the sink informs its current position to the network, so that source node can disseminate the data to the sink [19, 20]. In cluster-based data dissemination strategy [3, 4], the network is divided into a number of clusters and each cluster is associated with a cluster head. Sink informs its present position to the nearest cluster head. Each cluster head aggregates the data and sends it to the sink. Intanagonwiwat et al. [21] have proposed a flat multihop routing protocol called directed diffusion. The sink node broadcasts its interest to the network. The node which satisfies the interest will send the data to the sink. The routing path is constructed by setting gradients towards the sink. In tree-based data dissemination approach, a tree is constructed to disseminate the data to the sink. The tree structure frequently changes according to the new position of the sink. Kim at al. [12] have proposed a tree-based routing protocol for mobile sink. In this protocol, a tree is constructed to disseminate the data to mobile sink, via an access node. The access node is the node which can send the data directly to the sink and the location of the sink known by the access node only. The dissemination tree is reconstructed when the network needs to elect the new access node. Zhang et al. [22] provide the solution to reconfigure the tree when the source and the sink change their positions. However, it uses more control packets as the tree size increases. Hwang and Eom [14] have proposed an adaptive reversal tree (ART) protocol. It uses an adaptive reversal algorithm to make the reverse link to the sink node. It uses a dynamic method to manage the mobile sink. Hwang and Eom [13] have proposed another tree-based protocol called Distributed Dynamic Shared Tree. The shape of the tree dynamically changes according to the sink location. For making a sink-oriented tree, the sink selects a root node. It maintains a robust connectivity with the sink. Another tree-based protocol, which supports multiple mobile sink, has been proposed by Carneiro Viana et al., known as SUPPLE [15]. This protocol allows each source to send its generated data to the target set. The target sets are the

nodes called storing nodes. This scheme efficiently selects those well distributed storing nodes. They store the incoming data until the sink comes in its trajectory. Faheem and Boudjit have proposed a multipoint relay-based data dissemination protocol called SN-MPR [16]. In this scheme, the tree is constructed with the sink as a root.

In this paper, we have discussed tree-based protocols in detail as they increase energy efficiency and decrease the latency due to their connectivity capability. Although there is the drawback of the above tree-based protocols, which is the high mobility management cost, the mobility management cost depends on the affected area due to sink mobility. To solve this problem, we proposed a tree-based protocol. The structure of the tree depends on the nodes instead of the mobile sink. Source nodes can send their data to the sink through relay nodes.

3. Proposed Model

We proposed an energy-efficient data dissemination protocol which generates a tree T from the sensor network. It can be represented as a graph $G(V, E)$ where V are the sensor nodes and E are the links between them. The tree construction is independent of the sink position. This method reduces the traffic and prolongs the lifetime of the network.

3.1. TEDD: Tree-Based Efficient Data Dissemination Protocol. TEDD is an energy-efficient data dissemination protocol with mobile sink. Initially, it creates the tree in the network with a root node. There are two categories of the nodes in the network; one is relay node (*RN*) and the other is nonrelay node (*non-RN*). The relay node is responsible to relay the data from the nodes to its next relay node. The nonrelay node can only communicate its data to a relay node. So it is a unidirectional communication between nonrelay and relay nodes. However, the communication is bidirectional between two relay nodes. The tree topology changes when the role of the node changes from relay to nonrelay or from nonrelay to relay node. To rotate the responsibility of the relay node, each node's residual energy is considered.

The sink is mobile and collects the data from the source nodes through the gateway node. The gateway node may be the relay node or the nonrelay node. The gateway node is selected by the sink based on the criteria mentioned in Section 3.2. The sink periodically transmits a small beacon to make the connection alive with the gateway node. If the sink moves out from the range of the current gateway node, then it elects another node as the gateway node. The rotation of the gateway node can overcome the problem of the energy hole [2].

3.2. TEDD Working Principle. Let n be the number of sensor nodes which are randomly deployed in the network. All nodes are homogeneous and static in nature. Each node possesses its id and knows the residual energy. In the beginning, the initiator node triggers the neighbor discovery phase by broadcasting the NBR_DET control packet. At the
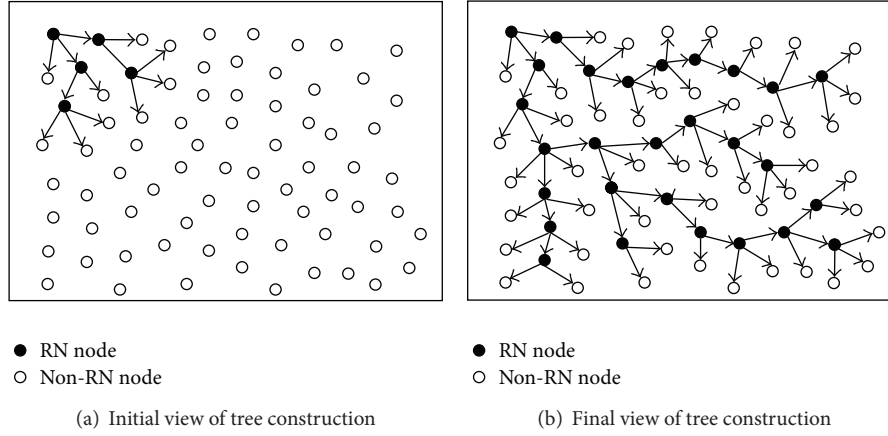
- ● RN node
- ○ Non-RN node

(a) Initial view of tree construction

- ● RN node
- ○ Non-RN node

(b) Final view of tree construction

FIGURE 1: Tree construction steps shown in (a) and (b).

- ● RN node
- ○ Non-RN node

(a) Link reversal process

- ● RN node   ◉ Gateway RN
- ○ Non-RN node   ⬧ Gateway non-RN
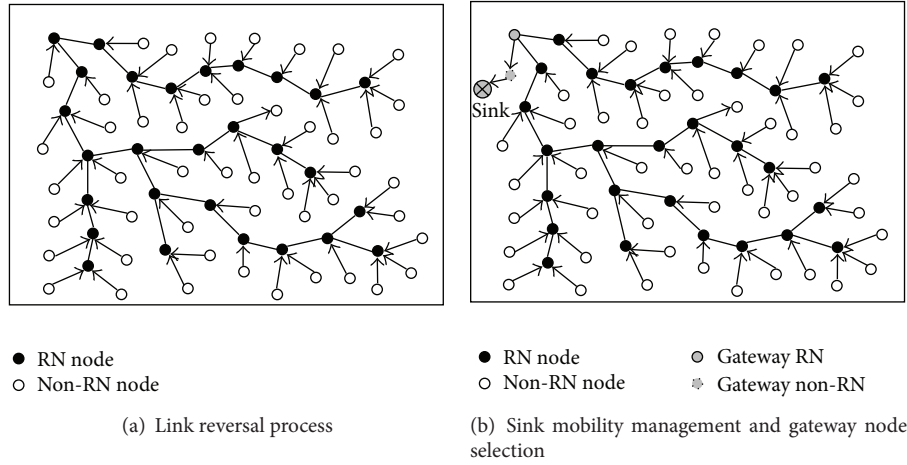
(b) Sink mobility management and gateway node selection

FIGURE 2: Link reversal and sink mobility management shown in (a) and (b).

end, each node acquires neighbor list $NBR(x)$ and candidate relay node list $RN(x)$.

The initiator node triggers the tree construction by broadcasting T_MSG. Figure 1 illustrated the construction of the tree in the network. There are two types of nodes in the network, as shown in Figure 1: relay node and nonrelay node.

After the tree construction, it is required to reverse the communication link. The nonrelay node only communicates to its parent relay node, and the relay node communicates to its neighbor relay node. The link between nonrelay and relay nodes is unidirectional and between relay node and relay node is bidirectional as shown in Figure 2(a).

The source nodes can send their data to the sink by managing the mobility of the sink. The interface between network and sink is the gateway node. The gateway node is selected by the sink if that node is in the range. Mobile sink periodically broadcasts the small signal called beacon to notify the neighbor sensor nodes. The nodes that receive the beacon send their response to become the gateway node based on their residual energy. The sink selects one of them and declares it as the gateway node. Among the responses received by the sink, it prefers relay node as the gateway. If the gateway is a nonrelay node, then its parent relay node will be

the gateway node and set the link as shown in Figure 2(b). The gateway node sends the RREQ packet to relay nodes to make the path towards the gateway node for data transmission. The data dissemination starts as soon as routing path construction is over as shown in Figure 3. The proposed protocol consists of various phases like neighbor discovery, tree construction, relay node selection, and data dissemination.

3.2.1. Neighbor Discovery. It is the initial phase of the proposed protocol in which each node finds its neighbor nodes. As illustrated in Algorithm 1, the initiator node broadcasts the NBR_DET packet. It includes the node id of the sender id_x and the willingness to be the relay node, ⟨NBR_DET, id_x, $WILL_x$⟩. The willingness is decided by the sender node itself based on its residual energy E_r. If $E_r \geq E_{\text{Threshold}}$, $WILL_x$ will be true, otherwise false. In the protocol, we assumed that the threshold energy is the half of the node's initial energy. Any node x that receives the NBR_DET packet does the following operations.

(i) It checks for existence of the sender node id; if it is not found, it includes the sender node id in the neighbor list $NBR(x)$.

Data structure for any sensor node x:

$NBR(x)$: neighbor set of node x, initialized to ϕ.

$RN(x)$: the set of neighbors of node x, which are willing to be the relay node, initialized to ϕ.

$WILL_x$: either **true** or **false** depends on the willingness of node x to become a relay node.

$NBR\_DET\_SENT_x$: 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: $\langle$NBR_DET$, id_y, WILL_y\rangle$

if ($y \notin NBR(x)$) **then**

   $NBR(x) \leftarrow NBR(x) \cup \{y\}$;

   if ($WILL_y == true$) **then**

      $RN(x) \leftarrow RN(x) \cup \{y\}$;

   end if

   if ($NBR\_DET\_SENT_x == false$) **then**

      $NBR\_DET\_SENT_x \leftarrow true$;

      l_rb(NBR_DET$, id_x, WILL_x$); $\triangleright$ Broadcast NBR_DET packet

   else

      Drop the packet;

   end if

else

   Drop the packet;

end if

ALGORITHM 1: Neighbor discovery.

(ii) It checks for the willingness to be a relay node; if this is found to be true, it includes sender node id in the relay node list $RN(x)$.

(iii) It checks if the NBR_DET packet is broadcasted by the receiver node; if it is not broadcasted, then it broadcasts the packet with format $\langle$NBR_DET$, id_x, WILL_x\rangle$ and makes $NBR\_DET\_SENT_x$ as true.

Neighbor discovery phase is over as soon as each node broadcasts its NBR_DET packet. At the end, each node gets the partial view of the network in the form of neighbor information.

3.2.2. Tree Construction and Relay Node Selection. After getting the neighbor list, each node has the neighbor information such as id and the willingness to become the relay node. The tree construction and relay node selection phase are initiated by using the neighbor information. As depicted in Algorithm 2, the initiator node starts the tree construction by broadcasting the T_MSG control packet. The node receives the following packets during the tree construction and relay node selection phase.

T_MSG. In the process of tree construction, T_MSG control packet is used. The format of the packet is $\langle$T_MSG, $id_y, Parent(y)\rangle$. Here, id_y is the sender node id and $Parent(y)$ is its parent node id. Any node x that receives the T_MSG packet performs the following operations.

(i) If the sender's parent node id is the same as the receiver node id, then, it includes the sender id in the children list $Children(x)$ and includes the receiver id in the relay node list RN_{nodes}.

(ii) If it has not selected any parent and sender belongs to the list of relay node RN_{nodes}, then, it selects sender node as its parent.

(iii) If T_MSG_SENT is false, then, it broadcasts T_MSG packet with modified parameter to the network.

T_ERR. Timeout occurs to the node when the time duration expires for the tree construction phase. Any node y checks for its parent node; if it does not exist, then node y broadcasts an error message T_ERR to its neighbor nodes. The receiver node performs the following operation.

(i) It initiates tree construction by broadcasting T_MSG if it belongs to the tree; otherwise, it drops the packet.

In this way, the rest of the nodes that do not belong to the tree will get the opportunity to connect with the tree.

3.2.3. Data Dissemination. Data can be generated by the nonrelay nodes or relay nodes. The responsibility of relay node is to forward the data to the next relay node. Any node can sense the data from the environment and send it to the next relay node. Node x receives the following packet during the data dissemination phase from node y as shown in Algorithm 3.

DATA. Each node in the network senses the environment, generates the data, and sends it towards the next relay node with the format $\langle$DATA$, id_y, sec\_no_y\rangle$. Here, id_y is the id of sender node y and sec_no_y is the data sequence number of the node y. Any node that receives the DATA packet performs the following actions.

(i) If receiver node is a relay node and it receives the duplicate data, then it drops the data packet.

Data structure for any sensor node x:

$Children(x)$: children set of node x, initialized to ϕ.

$Parent(x)$: parent of node x, initialized to ϕ.

RN_{nodes}: set of relay nodes in the network.

$Parent\_Selected_x$: set to **true** once the sensor node x selects its parent, initialized to **false**.

$T\_MSG\_SENT_x$: set to **true** once the sensor node x sends T_MSG packet, initialized to **false**.

$RN(x)$: the set of neighbors of node x, which are willing to be the relay node, initialized to ϕ.

Node x **receives following packets from node** $y \in NBR(x)$:

T_MSG: $\langle$T_MSG, id_y, $Parent(y)\rangle$

If ($id_x \in Parent(y)$) **then**
    $Children(x) \leftarrow Children(x) \cup \{id_y\}$;
    $RN_{nodes} \leftarrow RN_{nodes} \cup \{x\}$;   ▷ node x declares itself as a relay node
    Drop the packet;
else if ($Parent\_Selected_x == false$ and $y \in RN(x)$) **then**
    $Parent(x) \leftarrow y$;
    $Parent\_Selected_x \leftarrow true$;
    if (($T\_MSG\_SENT_x == false$)) **then**
        $T\_MSG\_SENT_x \leftarrow true$;
        $l\_rb(\text{T\_MSG}, id_x, Parent(x))$;   ▷ Broadcast T_MSG packet
    else
        Drop the packet;
    end if
else
    Drop the packet;
end if

        ▷ Timeout occurs to the node y when the time duration expires for the tree construction
phase and $TIMEOUT_y$ become $true$.

if ($TIMEOUT_y == true$) **then**
    if ($Parent\_Selected_y == false$) **then**
        $l\_rb(\text{T\_ERR}, id_y)$;   ▷ Broadcast T_ERR packet
    end if
end if

T_ERR: $\langle$T_ERR, $id_y\rangle$

If ($Parent\_Selected_x == true$) **then**
    $T\_MSG\_SENT_x \leftarrow true$;
    $l\_rb(\text{T\_MSG}, id_x, Parent(x))$;   ▷ Broadcast T_MSG packet
else
    Drop the packet;
end if

ALGORITHM 2: Tree construction and relay node selection.

- ● RN node
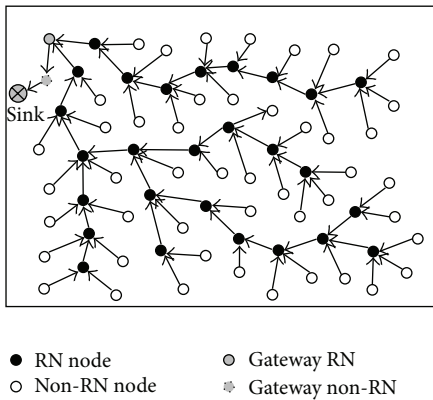- ○ Non-RN node
- ◍ Gateway RN
- ✦ Gateway non-RN

FIGURE 3: Path construction for gateway node and data transmission.

(ii) If receiver node is a gateway node, then it forwards the data packet to the sink; otherwise, it forwards the DATA packet to its next relay node.

(iii) It adds the sender id and data sequence number to the list $Send\_Data(x)$.

4. Simulation Model

4.1. Experimental Setup and Simulator. The simulation is performed using the network simulator NS-2 version 2.34. In NS-2, we concentrated in the network layer more specifically on routing protocol. Our aim is to simulate the proposed protocol (TEDD) and the existing protocols such as SUPPLE [15], SN-MPR [16], and ART [14] to examine the energy consumption, end-to-end delay, and throughput of the network.

```
Data structure for any sensor node x:
Send_Data(x): node x adds the pair of id and sec_no after receiving the DATA packet, initialized to φ.

Node x will receive following packet from node y ∈ NBR(x):
DATA: ⟨DATA, id_y, sec_no_y⟩
if (x ∈ RNnode) then
    if (⟨id_y, seq_no_y⟩ ∉ Send_Date(x)) then
        if (x == Gateway) then
            Send_Data(x) ← Send_Data(x) ∪ {y, sec_no_y};
            Forward DATA packet towards the sink;
        else
            Send_Data(x) ← Send_Data(x) ∪ {y, sec_no_y};
            Forward DATA packet to its neighbor relay node towards gateway
        end if
    else
        Drop the packet;
    end if
else
    Drop the packet;
end if
```

ALGORITHM 3: Data dissemination.

In the simulation, we use the specification of MICAz [23], a popular sensor mote, to make the simulation support to the real hardware parameters of the sensor networks. The MICAz mote transceiver power range is from −24 dbm to 0 dbm and outdoor communication range is from 75 m to 100 m. Our simulation follows the power consumption model of the MICAz motes that require about 1.0 mJ for transmitting, about 0.5 mJ for receiving, and about 0.04 mJ in idle mode. The transceiver in the simulation has an 80 m radio range at 2.4 GHz frequency, which is the case with the radio transceiver of a MICAz mote. The initial energy of each sensor node is 1.0 J at the time of deployment. For fair comparison between the proposed protocol and the existing protocol, we set simulation parameters equivalent to SUPPLE [15], SN-MPR [16], and ART [14]. The simulation runs with up to 200 sensor nodes with energy constraint and a mobile sink with no constraint. The nodes are randomly deployed in the 1000×1000 meter^2 area. The simulation also includes IEEE 802.11 as the underlying MAC protocol. The sensor node's energy model and sink mobility model are discussed in Sections 4.2 and 4.3. In NS-2, we use omnidirectional antenna and two-ray ground model for radio propagation. Each sensor node senses the environment and generates data of 64 bytes at each θ interval (here, $\theta = 1$ sec), and the size of the control packet is 32 bytes. We performed extensive simulations up to the duration of 200 seconds.

4.2. Energy Model.

Each sensor node constantly calculates its residual energy based on the energy model. The energy consumption in the sensor nodes depends on the various radio interface mode and processing cost. The energy modeling in the sensor network is based on the theoretical energy consumption. In the energy model, we consider the energy consumption due to transmission of the packet (1), reception of the packet (2), and energy spent by nodes in the idle mode (3). The total energy consumption (4) of a sensor node is the sum of transmission, receiving, and idle mode energy consumption. Consider the following:

$$E_{\text{Transmission}} = E_{XT} * t \, (\text{bits}) + E_{XP} \left(\text{d}^2 \right), \qquad (1)$$

$$E_{\text{Receiving}} = E_{XR} * t \, (\text{bits}) + E_{XA} * t \, (\text{bits}), \qquad (2)$$

$$E_{\text{Sleep}} = E_{XI} * t \, (\text{sec}), \qquad (3)$$

$$E_{\text{Total}} = E_{\text{Transmission}} + E_{\text{Receiving}} + E_{\text{Sleep}}. \qquad (4)$$

In (1), (2), and (3), E_{XT} refers to energy consumption per bit for transmission, $E_{XP}(\text{d}^2)$ is the energy consumed for finding the next hop forward node, E_{XR} is the energy consumption per bit for receiving, and E_{XA} refers to the energy consumption per bit for aggregating the received data packet. E_{XI} is the energy consumption per second in ideal mode. In the proposed protocol, the sensor nodes are of two types: relay node or nonrelay node. A nonrelay node will not consume energy in aggregation (E_{XA}) since it only receives the control packets. The energy consumption is calculated in the joule per node to find the total energy consumption. The consumption of energy is measured in each phase such as the neighbor detection, tree construction, relay node selection, sink mobility management, and data dissemination phase.

4.3. Mobility Model.

In the simulation, to show the impact of the sink mobility, we considered two mobility models: Gaussian-Markov model [24] and random waypoint model [25].

4.3.1. Gaussian-Markov Model.

The Gaussian-Markov model has been initially proposed for PCS [24] and also used in

the ad hoc networks. It is a mobility model which generates the next position depending on the previous position and considering the parameters like speed and direction.

If at time t_1 the initial position of the sink is $P(X_1, Y_1)$, then, the next position is determined with the following equations:

$$X_n = X_{n-1} + S_{n-1} \cos(D_{n-1}),$$
$$Y_n = Y_{n-1} + S_{n-1} \sin(D_{n-1}). \tag{5}$$

Here, S_{n-1} and D_{n-1} are speed and direction; (X_{n-1}, Y_{n-1}) and (X_n, Y_n) are the old and new positions of the sink, respectively. The Gaussian-Markov model is used to calculate the (n)th position, direction, and speed from the $(n-1)$th position, direction, and speed. The equations for speed (S_n) and direction (D_n) are as follows:

$$S_n = \alpha S_{n-1} + (1 - \alpha) S' \sqrt{(1 - \alpha^2)} Sx_{n-1},$$
$$D_n = \alpha D_{n-1} + (1 - \alpha) D' \sqrt{(1 - \alpha^2)} Dx_{n-1}, \tag{6}$$

where S' and D' are the values representing the mean of the speed and direction as $n \rightarrow \infty$; Sx_{n-1} and Dx_{n-1} are random variables from a Gaussian distribution. The level of randomness is obtained by varying the value of α from 0 to 1; that is, $0 \leq \alpha \leq 1$.

To restrict the sink within the bounded area, we consider the boundary value $P_{\max}$, that is, $[X_{\max}, Y_{\max}]$. The calculation of the next position takes place from the previous nonboundary position. Sink keeps the earlier position in the memory as long as it does not get the valid subsequent position, so this model generates the relative motion of the sink. For the experiment, we consider the sink *Pause* time (δ) as 5 sec. The extensive simulations are performed for the protocol with the speed $S = (5, 10, 15, 20, 25, 30)$ meter/sec.

4.3.2. Random Waypoint Model. Random waypoint model is a "benchmark" mobility model for ad hoc networks to evaluate the performance of the routing protocol. We consider the random waypoint model for the sink mobility in wireless sensor networks. In the network simulator (NS-2), *setdest* tool from the CMU monarch group widely used random waypoint model. It randomly generates the next position in between $P_{\min}$ and $P_{\max}$. It then travels towards its next position with constant speed or random speed. The simulation is performed with the speed of $S = (5, 10, 15, 20, 25,$ and $30)$ meter/sec. When the sink node reaches the next position, it pauses for a duration called the *Pause* time (δ); here, we consider (δ) = 5 sec.

Unlike the Gaussian-Markov model, the random waypoint model does not consider the previous position to calculate the next position. Hence, it does not generate the relative motion. In the simulation, we have analyzed the impact of relative motion and random motion of the sink in various data dissemination protocols with the pause time (δ) and the speed (S).

4.4. Performance Metrics

4.4.1. Energy Consumption. Energy consumption at each node is considered. The total communication energy includes neighbor discovery, tree construction, mobile sink management, and data transmission. In the experiment, we consider the control packet and the control plus data packet communication. The goal is to minimize the control packet overhead to manage the mobile sink. Due to the less control overhead, total communication energy also decreases, which prolongs the lifetime of the network. This metric indicates how efficiently a protocol works in the network.

4.4.2. Average End-to-End Delay. Average end-to-end delay is measured as the average time between sending and successfully receiving a packet. Here, the sender is the sensor node, and the receiver is the sink. We can say, the average time a packet takes to reach the sink. It considers all types of delays such as queuing delay, route discovery delay, and interface delay.

4.4.3. Throughput (Packet Delivery Ratio). Throughput is measured as the ratio of packet received at the sink to the packet sent by the sensor node. Throughput defines the successful delivery of the data packet. Protocol with better throughput is considered as the consistent protocol. This metric also indicates the degree of reliability and robustness of the routing path.

5. Simulation Result

The performance of the proposed protocol TEDD is evaluated, and the result is compared with the tree-based models such as SUPPLE [15], SN-MPR [16], and ART [14]. In the simulation, we have used a mobile sink and 200 randomly deployed sensor nodes. Each experiment has been performed with the varying sink speed from 5 meter/sec to 30 meter/sec. We observed the impact of sink speed in energy consumption, end-to-end delay, and throughput. In addition, we observed the impact of mobility models like Gaussian-Markov and random waypoint model in energy consumption.

5.1. Energy Consumption

5.1.1. Average Energy Consumption of Control Packet. Figure 4 illustrated the average energy consumption of control packet in the network with varying sink speed. To construct a tree and manage the sink mobility, the sensor node transmits the control packets. The tree reconstruction and sink management cost much less in the proposed protocol (TEDD) as compared to the other protocols.

In ART [14], the whole network should know the current position of the sink. The tree is rebuilt with the nearest node as root. The tree reconstruction cost of ART depends on the affected area.

In SN-MPR [16], the root of the tree is the sink. Like ART, SN-MPR also rebuilt the tree when the sink moves. However,
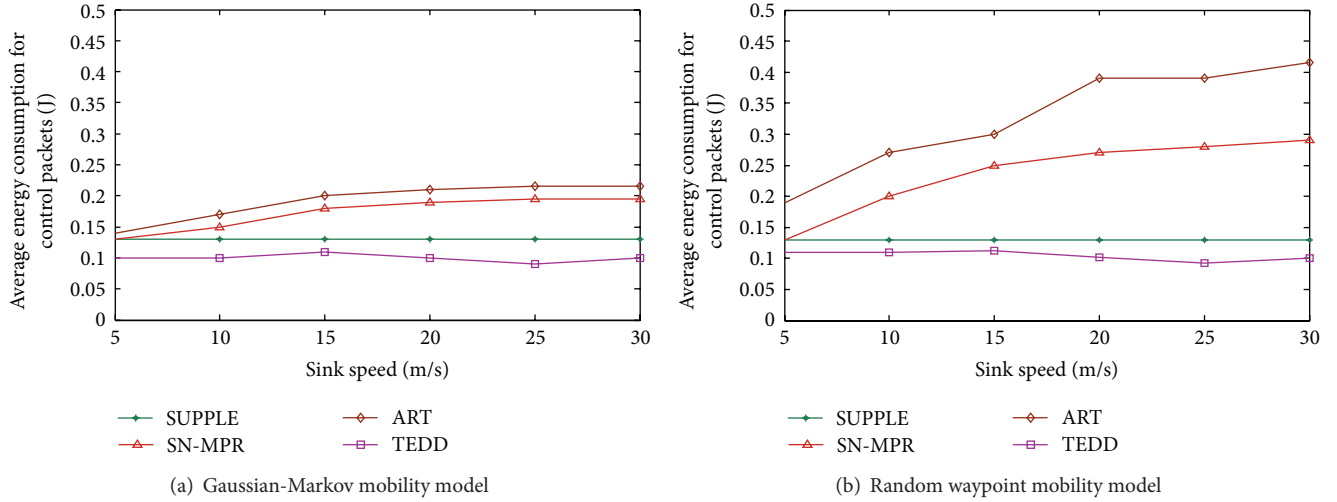
(a) Gaussian-Markov mobility model

(b) Random waypoint mobility model

FIGURE 4: Average energy consumption for control packet with changing sink speed. Result with different mobility model is shown in (a) and (b).

(a) Gaussian-Markov mobility model

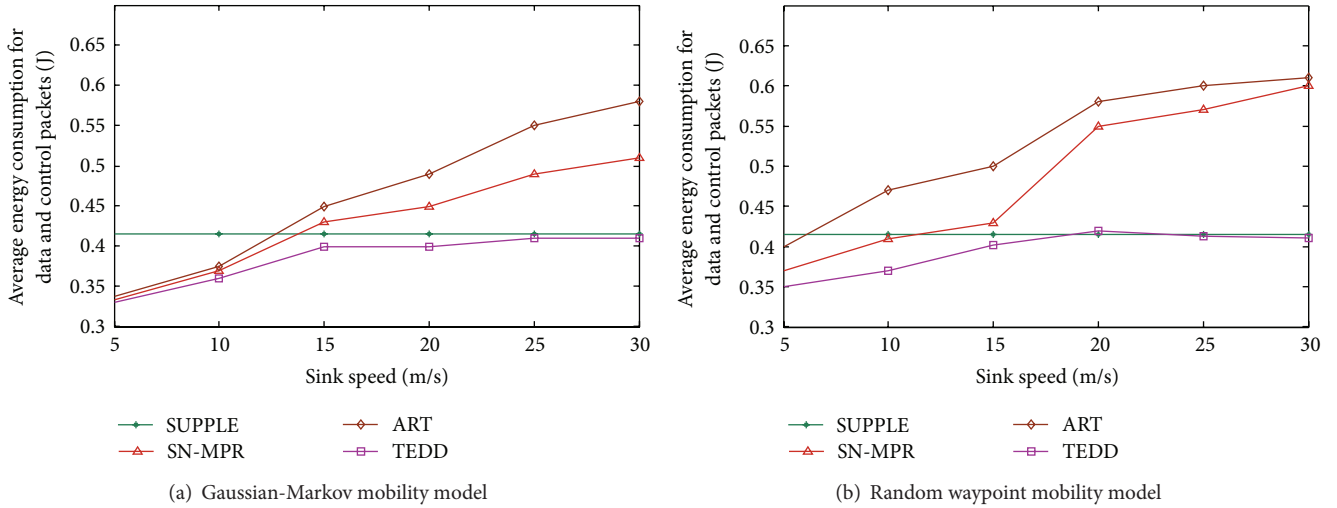(b) Random waypoint mobility model

FIGURE 5: Average energy consumption for data and control packet with changing sink speed. Result with different mobility model is shown in (a) and (b).

the new position of the sink is only known to the selected nodes. So the control overhead of the SN-MPR is less than that of the ART.

In SUPPLE [15], the tree is constructed and storing nodes are selected. The storing nodes temporarily store the data from the nodes. When the sink comes in the range, the storing node transmits the data. Unlike the above protocols, the SUPPLE does not depend on the movement of the sink. So control packet overhead is only due to tree formation and storing node selection.

In TEDD, the new position of the sink should be known only to the one-hop neighbors, which leads to the less control packet overhead.

5.1.2. Average Energy Consumption of Data and Control Packet. The total energy consumption at each node for data

and control packet is shown in Figure 5. Although in the proposed protocol the average distance between source and sink is the same as ART and SN-MPR, due to the less control packet overhead, the proposed protocol (TEDD) outperforms the existing protocols.

In SUPPLE, the average distance between the source and the storing nodes is $n/2$, where n is the number of sensor nodes. The distance between the storing nodes to the sink is one-hop. Although the average distance is less, it consumes more energy than the proposed protocol. Since each storing node stores the data of all the sensor nodes, it increases the traffic of the network; hence, it raises the energy consumption.

5.1.3. Impact of Mobility Model in Energy Consumption. The average energy consumption due to control packet and data

(a) Average end-to-end delay   (b) Data delivery ratio
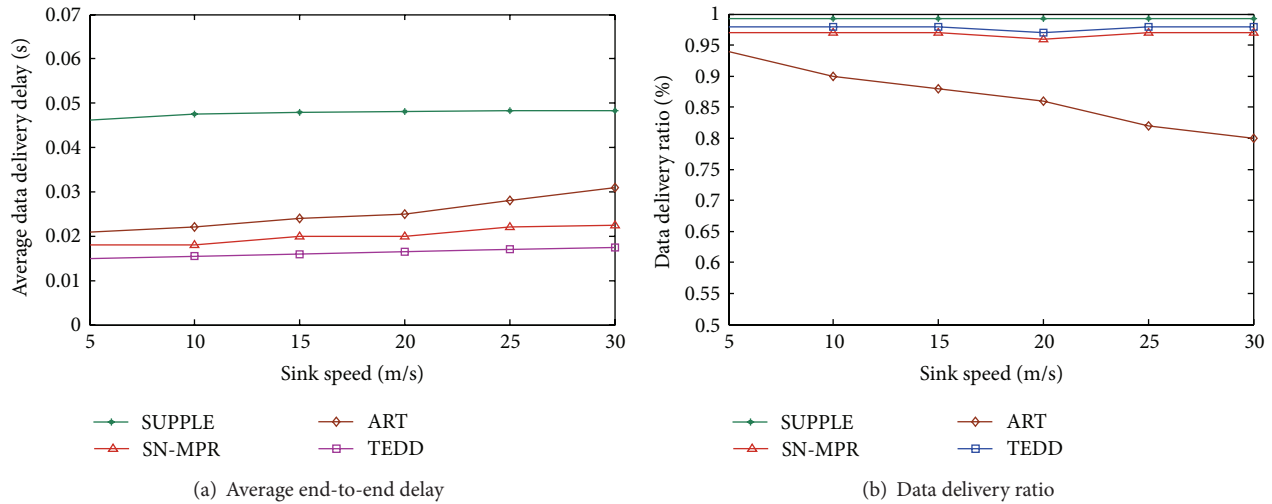
Figure 6: Result with changing sink speed is shown in (a) and (b).

plus control packet by using two different mobility models for sink is illustrated in Figures 4 and 5, respectively. The results shown in Figures 4(a) and 5(a) were obtained by using the Gaussian-Markov mobility model. To observe the impact of mobility, we use another mobility model called the random waypoint model shown in Figures 4(b) and 5(b). It can be observed from Figures 4 and 5 that the cost of the tree reconstruction in SN-MPR and ART protocols with the random waypoint model is higher than the Gaussian-Markov model. This is due to the fact that the affected area is more in the random waypoint model in comparison with the Gaussian-Markov model.

Although the proposed protocol (TEDD) is not affected by the different mobility models, the current position of the sink does not have to disseminate throughout the network. It only affects the one-hop neighbors at a time, and the tree construction is independent of the sink position.

The energy consumption in the SUPPLE protocol remains unchanged since there is no effect in the network with different mobility models.

5.2. Average End-to-End Delay. Delay mainly depends on the time to find the valid path between source and sink. Delay increases if the data generation rate is more than the data reception rate. Figure 6(a) presents the average end-to-end delay with various sink speed using the Gaussian-Markov mobility model.

The time required to reconstruct the tree based on the new position of the sink causes the delay in ART and SN-MPR. In SN-MPR, the affected area is less than that in ART. So ART causes more end-to-end delay than SN-MPR.

In SUPPLE, the sensor data is temporarily stored in the storing nodes. The storing nodes wait for the sink to come in the trajectory. It causes more end-to-end delay than the above protocols.

The proposed protocol TEDD overcomes all the drawbacks of SUPPLE, ART, and SN-MPR because it requires less cost and time to manage the mobility of the sink. It

can be seen from Figure 6(a) that TEDD outperforms the abovementioned protocols in terms of average end-to-end delay.

5.3. Throughput (Data Delivery Ratio). Figure 6(b) shows the data delivery ratio with respect to different sink speeds. Throughput represents the success ratio of the data delivery. SUPPLE performed well because the distance between sink and storing node is one-hop. SN-MPR also performed well due to less affected area and recovery technique. The success ratio for ART decreases as the sink speed rises. The higher sink speed increases the frequency of link failure, which causes data loss. However, the proposed protocol (TEDD) is robust; that is, the link is always maintained between the source and the sink, so the throughput is very high.

5.4. Conclusion. In this paper, we proposed a distributed, robust, and efficient tree-based data dissemination protocol called TEDD. The proposed protocol can effectively and efficiently manage the sink mobility. We simulated the proposed protocol with two different mobility models. The results are compared with the existing protocols such as SUPPLE, SN-MPR, and ART. It was observed that TEDD outperformed the above protocols due to its unique method to handle the mobility of the sink.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

References

[1] L. Popa, A. Rostamizadeh, R. Karp, C. Papadimitriou, and I. Stoica, "Balancing traffic load in wireless networks with curveball routing," in *Proceedings of the 8th ACM International Symposium on Mobile Ad Hoc Networking and Computing*, pp. 170–179, September 2007.

[2] J. Li and P. Mohapatra, "Analytical modeling and mitigation techniques for the energy hole problem in sensor networks," *Pervasive and Mobile Computing*, vol. 3, no. 3, pp. 233–254, 2007.

[3] N. M. Khan, I. Ali, Z. Khalid, G. Ahmed, A. A. Kavokin, and R. Ramer, "Quasi centralized clustering approach for an energy-efficient and vulnerability-aware routing in wireless sensor networks," in *Proceedings of the 1st ACM International Workshop on Heterogeneous Sensor and Actor Networks, HeterSanet 2008*, pp. 67–72, ACM, May 2008.

[4] R. Sudarmani and K. R. S. Kumar, "Energy-efficient clustering algorithm for heterogeneous sensor networks with mobile sink," *European Journal of Scientific Research*, vol. 68, no. 1, pp. 60–71, 2012.

[5] L. Song and D. Hatzinakos, "Dense wireless sensor networks with mobile sinks," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '05)*, pp. 677–680, IEEE, March 2005.

[6] L. Song and D. Hatzinakos, "Architecture of wireless sensor networks with mobile sinks: sparsely deployed sensors," *IEEE Transactions on Vehicular Technology*, vol. 56, no. 4, pp. 1826–1836, 2007.

[7] D. Puthal, B. Sahoo, and S. Sharma, "Dynamic model for efficient data collection in wireless sensor networks with mobile sink," *International Journal of Computer Science and Teleology*, vol. 3, no. 1, pp. 623–628, 2012.

[8] P. Juang, H. Oki, Y. Wang, M. Martonosi, L. S. Peh, and D. Rubenstein, "Energyefficient computing for wildlife tracking: design tradeoffs and early experiences with zebranet," *SIGOPS Operation System Review*, vol. 36, no. 5, pp. 96–107, 2002.

[9] S. Farrell, V. Cahill, D. Geraghty, I. Humphreys, and P. McDonald, "When TCP breaks: delay- and disruption-tolerant networking," *IEEE Internet Computing*, vol. 10, no. 4, pp. 72–78, 2006.

[10] L. Selavo, A. Wood, Q. Cao et al., "Luster: Wireless sensor network for environmental research," in *Proceedings of the 5th ACM International Conference on Embedded Networked Sensor Systems*, pp. 103–116, ACM, New York, NY, USA, November 2007.

[11] H. Luo, F. Ye, J. Cheng, S. Lu, and L. Zhang, "TTDD: two-tier data dissemination in large-scale wireless sensor networks," *Wireless Networks*, vol. 11, no. 1-2, pp. 161–175, 2005.

[12] H. S. Kim, T. F. Abdelzaher, and W. H. Kwon, "Minimum-energy asynchronous dissemination to mobile sinks in wireless sensor networks," in *Proceedings of the 1st International Conference on Embedded Networked Sensor Systems*, pp. 193–204, ACM, November 2003.

[13] K. I. Hwang, J. In, and D. S. Eom, "Distributed dynamic shared tree for minimum energy data aggregation of multiple mobile sinks in wireless sensor networks," *Proceedings of the 3rd European conference on Wireless Sensor Networks*, Springer, Berlin, Germany, vol. 3868, pp. 132–147, 2006.

[14] K. I. Hwang and D. S. Eom, "Adaptive sink mobility management scheme for wireless sensor networks," in *Proceedings of the 3rd International Conference on Ubiquitous Intelligence and Computing*, Lecture Notes in Computer Science, pp. 478–487, Springer, Berlin, Germany, 2006.

[15] A. Carneiro Viana, T. Herault, T. Largillier, S. Peyronnet, and F. Zaïdi, "Supple: A flexible probabilistic data dissemination protocol for wireless sensor networks," in *Proceedings of the 13th ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems*, pp. 385–392, ACM, October 2010.

[16] Y. Faheem and S. Boudjit, "SN-MPR: A multi-point relay based routing protocol for wireless sensor networks," in *Proceedings of the IEEE/ACM International Conference on Green Computing and Communications & International Conference on Cyber, Physical and Social Computing*, pp. 761–767, IEEE Computer Society, December 2010.

[17] N. C. Wang, Y. F. Huang, J. S. Chen, and P. C. Yeh, "Energy-aware data aggregation for grid-based wireless sensor networks with a mobile sink," *Wireless Personal Communications*, vol. 43, no. 4, pp. 1539–1551, 2007.

[18] E. Lee, S. Park, F. Yu, Y. Choi, M. S. Jin, and S. H. Kim, "A predictable mobility-based data dissemination protocol for wireless sensor networks," in *Proceedings of the 22nd International Conference on Advanced Information Networking and Applications*, pp. 741–747, IEEE Computer Society, March 2008.

[19] G. Wang, T. Wang, W. Jia, M. Guo, and J. Li, "Adaptive location updates for mobile sinks in wireless sensor networks," *Journal of Supercomputing*, vol. 47, no. 2, pp. 127–145, 2009.

[20] A. Munari, W. Schott, and S. Krishnan, "Energy-efficient routing in mobile wireless sensor networks using mobility prediction," in *Proceedings of the IEEE 34th Conference on Local Computer Networks (LCN '09)*, pp. 514–521, IEEE, October 2009.

[21] C. Intanagonwiwat, R. Govindan, and D. Estrin, "Directed diffusion: a scalable and robust communication paradigm for sensor networks," in *Proceedings of the 6th Annual International Conference on Mobile Computing and Networking (MOBICOM '00)*, pp. 56–67, ACM, Boston, Mass, USA, August 2000.

[22] W. Zhang, G. Cao, and T. La Porta, "Dynamic proxy tree-based data dissemination schemes for wireless sensor networks," *Wireless Networks*, vol. 13, no. 5, pp. 583–595, 2007.

[23] Crossbow Technology, I. Micaz datasheet. Technical report, San Jose, Calif, USA, http://www.openautomation.net/uploadsproductos/micazdatasheet.pdf.

[24] B. Liang and Z. J. Haas, "Predictive distance-based mobility management for PCS networks," in *Proceedings of the 18th Annual Joint Conference of the IEEE Computer and Communications Societie*, pp. 1377–1384, IEEE, March 1999.

[25] J. Broch, D. A. Maltz, D. B. Johnson, Y. C. Hu, and J. Jetcheva, "A performance comparison of multi-hop wireless ad hoc network routing protocols." in *Proceedings of the 4th Annual ACM/IEEE International Conference on Mobile Computing and Networking*, pp. 85–97, ACM, 1998.