INTERNATIONAL JOURNAL OF COMMUNICATION SYSTEMS
Int. J. Commun. Syst. (2014)
Published online in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/dac.2785

An efficient distributed routing protocol for wireless sensor networks with mobile sinks

Lei Shi[1,2], Zheng Yao[1], Baoxian Zhang[1,*,†], Cheng Li[3] and Jian Ma[4]

[1]*Research Center of Ubiquitous Sensor Networks, University of Chinese Academy of Sciences, Beijing, 100049, China*
[2]*College of Information and Electronic Engineering, Zhejiang Gongshang University, Hangzhou, Zhejiang 310018, China*
[3]*Memorial University of Newfoundland, St. John's, NL A1B 3X5, Canada*
[4]*State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing, 100876, China*

SUMMARY

Introducing mobile sinks into a wireless sensor network can effectively improve the network performance. However, sink mobility can bring excessive protocol overhead for route maintenance and may offset the benefit from using mobile sinks. In this paper, we propose a dynamic layered routing protocol to address this problem. The proposed protocol integrates dynamic layered Voronoi scoping and dynamic anchor selection to effectively reduce the dissemination scopes and frequencies of routing updates as the sinks move in the network. Simulation results show that the proposed protocol can effectively reduce the protocol overhead while ensuring high packet delivery ratio as compared with existing work. Copyright © 2014 John Wiley & Sons, Ltd.

Received 14 March 2013; Revised 10 November 2013; Accepted 3 March 2014

KEY WORDS: distributed routing; protocol overhead; Voronoi scoping; mobile sink; wireless sensor network

1. INTRODUCTION

Advances in high performance and miniature hardware devices have greatly improved the capabilities of sensor nodes and have made the applications of wireless sensor networks (WSNs) become available [1–10]. In many WSN applications such as security patrols, disaster relief, environmental monitoring, industrial applications, and military battlefield, sink nodes often can move. For example, in a WSN-based campus anti-intrusion system, static sensor nodes are deployed to detect unwanted intrusion activities, and security patrols acting as mobile sinks are required to receive such intrusion alert information (if any). Hereafter, such WSNs with mobile sinks are referred to as mWSNs. WSNs with mobile sinks have many advantages such as alleviating hot spot problem, providing longer network lifetime, and relaxing the constraints on network connectivity and the high flexibility to adapt to different data reporting strategies according to applications' requirements [1–3,11–13]. Even though mobile sink is highly desirable, it also poses new challenges for designing efficient distributed routing protocols for such networks. As sink(s) move in the sensor deployment field, it causes unexpected changes of network topology, which can lead to high protocol overhead for searching and maintenance of routes from sensors to mobile sink. The excessive protocol overhead may potentially offset the benefits for using mobile sinks. Therefore, how to effectively reduce the protocol overhead while ensuring high data delivery performance is a critical issue for design of routing protocol for mWSNs.

*Correspondence to: Baoxian Zhang, Research Center of Ubiquitous Sensor Networks, University of Chinese Academy of Sciences, Beijing 100049, China.
†E-mail: bxzhang@ucas.ac.cn

Design of efficient routing protocols for mWSNs has attracted ,much attention, and much work has been carried out in this area. Existing routing protocols for mWSNs can be classified into the following two types: delay tolerant protocols and delay sensitive protocols. In delay tolerant protocols, sensor data can be reported in a delay tolerant fashion (e.g., in tens of minutes or even hours) when a mobile collector passes by the sensors. In delay sensitive protocols, data generated from sensor nodes need to be reported to the sink in a timely fashion. In this paper, we focus exclusively on design of delay sensitive routing protocol. Typical protocols in this aspect (e.g., AVRP [14], DDRP [15], LURP [16], TTDD [17]) can be either topology based or location based for achieving high routing performance. In general, a routing protocol for mWSNs needs to have high packet delivery ratio performance while keeping low protocol overhead. For an elegant survey on routing techniques in mWSNs, please refer to [18].

In this paper, we propose an efficient dynamic layered Voronoi scoping-based routing protocol (LVRP) for mWSNs. The design objective is to effectively reduce the protocol overhead for routing management while keeping high network performance. To achieve this design goal, LVRP adopts two strategies in its protocol design. One is dynamic layered Voronoi scoping, and the other is dynamic anchor selection for each mobile sink. In the original idea of Voronoi scoping [19], a sensor node is required to only reforward a received *interest* packet (for gradient establishment) if the packet came from the closest sink node. In this way, the interest dissemination overhead can be largely reduced. In our LVRP protocol, dynamic layered Voronoi scoping is proposed to further effectively restrict the dissemination scopes and frequencies of routing updates as sinks move and accordingly reduce the routing overhead. The dynamic anchor selection is another strategy introduced for effectively reducing the frequency of each mobile sink's routing update and also providing assistance in determining the reduced routing update dissemination scope. The integration of these two strategies can effectively reduce the total amount of control packets for building and refreshing data delivery structure in an mWSN. Finally, we conduct extensive simulation results, which demonstrate that LVRP can effectively reduce the protocol overhead while preserving high packet delivery ratio as compared with existing work.

The remainder of this paper is organized as follows. In Section 2, we introduce related work. In Section 3, we present the detailed design description of LVRP. In Section 4, we evaluate the performance of LVRP as compared with existing work by extensive simulations. In Section 5, we conclude this paper.

2. RELATED WORK

Existing routing protocols for mWSNs can be divided into the following two types. One is topology-based routing [14, 15], which needs to setup and maintain data forwarding structure and typically depends on flooding of signaling messages for route discovery and network state updates. The other is location-based routing [16, 17, 20–23], which exploits position information of nodes and often uses geographical forwarding for data delivery. Next, we will respectively introduce typical protocols belonging to either type.

2.1. Topology-based routing protocols

Topology-based routing protocols typically require to setup and maintain the whole or local network topology structure for data delivery. Sensor nodes forward data based on the discovered network topology and routing structure and also link availability between neighboring sensor nodes.

In [14], Tian *et al*. designed an anchor-based Voronoi routing protocol (AVRP) and a trail-based routing protocol (TRAIL) for mWSNs, which are applicable to different network scenarios. The AVRP protocol depends on the flooding of interest packets to create and maintain network topology structure. and each sensor node uses the received interest message to create and update its routing table. In order to effectively reduce the control overhead, AVRP introduces Voronoi scoping and anchor node to reduce the diffusion scope of interest packet and also the frequency at which the data delivery topology structure is refreshed. However, the AVRP protocol may lead to larger protocol overhead when the data traffic is relatively light. To effectively reduce the protocol overhead caused

DOI: 10.1002/dac

by sink mobility, in this paper, we further enhance the AVRP protocol to support dynamic layered Voronoi scoping. In the TRAIL protocol, each sensor node needs to find a routing path from its neighbor nodes for data forwarding. When a mobile sink moves, it leaves its trail information in each of its visited sensor nodes by periodic local broadcasting of *beacon* messages. When a packet holder forwards a data packet, it will first query its neighbors for fresher sink trail information. If it receives a response from a neighbor with fresher sink trail information, it forwards the data packet along the trail; otherwise, random walk routing is used, until a fresh trail (or the packet destination) is met, or timed out and thus dropped. The use of random walk routing, however, may largely degrade the network performance in a large-scale mWSN with a relatively small number of mobile sinks. In [15], Shi *et al*. designed a data-driven routing protocol (DDRP) for mWSNs. In the DDRP protocol, each sensor takes advantage of the broadcast feature of wireless medium for route learning, and mobile sinks only need to broadcast beacon message within their respective one-hop radio ranges instead of using either network-wide or partial network flooding. Although DDRP can effectively reduce the control overhead, it may lead to poor network performance when the number of data sources is quite sparse, which largely affects sensor nodes' route learning capability.

2.2. *Location-based routing protocols*

Location-based routing protocols for mWSNs typically require each node to know the location information of its own, its neighbors, and that of the packet destination and exploit geographical forwarding disciplines for packet delivery. A problem in location-based routing protocols is that location information of nodes may not always be available in low cost WSNs.

In [16], Wang *et al*. designed a local update-based routing protocol (LURP), which is a flooding-based location information update protocol. The design objective of LURP is to control the diffusion scope of sink's location information and thus reduce the protocol overhead caused by frequent location updates of mobile sink in an mWSN. In LURP, before a mobile sink moves, it first defines an area that covers itself. Before the mobile sink moves out of the area, it only needs to disseminate its location updates within the area. In [20], Wang *et al*. designed an enhanced version of LURP (ALURP) to further reduce the protocol overhead caused by sink mobility. However, for both LURP and ALURP, when the mobile sink moves out of the predefined area, it still needs to flood its location information throughout the network. Two-tier data dissemination (TTDD) [17] is grid-based and source-oriented location-based routing protocols for multicasting in mWSNs. In TTDD, each data source proactively builds a grid structure, which enables mobile sinks to continuously receive data from remote source sensor nodes by only flooding queries within a grid cell. However, as the number of data sources increases, TTDD can produce excessive protocol overhead for grid creation and maintenance. In [21], Yu *et al*. designed a geographic routing protocol for mWSNs (namely, elastic routing). In the elastic routing protocol, each packet holder encapsulates its locally stored location information of the mobile sink into the data packet to be delivered. Each sensor node takes advantage of the wireless broadcast nature to overhear the transmission of such a data packet for learning fresher location information of a mobile sink. The mobile sink only broadcasts periodically its location information to its neighbors instead of using network-wide flooding. Each intermediate node (starting from the second-to-the-last node) along an active path piggybacks the fresher destination location into each packet to transmit, in order to allow its upstream node on the path to learn such information. Such reverse learning of geographic location can largely reduce the protocol overhead. In [22], Yan *et al*. proposed a hierarchical location service protocol for each sensor node to timely and effectively obtain the up-to-date location information of the nearest mobile sink away from itself currently for data reporting. In [23], Lee *et al*. proposed a predictable mobility-based data dissemination protocol. On the basis of the following information associated with a mobile collector: predicted destination, mobility-related information, and data collection related information, a sensor node can send its data towards a location on the collector's moving path in advance without introducing frequent location updates of mobile collector(s). However, in predictable mobility-based data dissemination, a mobile collector still needs to flood an update to notify the entire network before it changes its moving pattern or destination. In [24], Shin *et al*. proposed a milestone-based predictive routing protocol, which introduces a list of milestone nodes for

DOI: 10.1002/dac

predicting a mobile sink's future location and then spreads the new location information only to the sensor nodes in the vicinity of the most recent trail of the sink (as determined by the milestone node list). In [25], Yu *et al*. proposed a dynamic multi-agent-based local update protocol for mWSNs. The dynamic multi-agent-based local update protocol dynamically sets up a number of agent nodes for transmitting data packets, and a mobile sink only needs to spread its new location information to the last agent instead of the whole network.

In [26, 27], the authors focused on studying how to control the mobility of sinks in a WSN, in a way such that sink always moves to nodes with high residual energy, in order to effectively balance the energy consumption and prolong the lifetime of the network. In [28], the authors focused on studying the rendezvous-based routing protocols for mWSNs. In rendezvous-based data collection protocols, sensor data are first sent to rendezvous points and then are buffered at the rendezvous points, until they are downloaded by mobile devices. Although rendezvous-based data collection protocols can largely reduce the control overhead for routing maintenance, they still require the availability of location information and may lead to hot spot problem in the regions close to the rendezvous points.

3. LAYERED VORONOI SCOPING-BASED ROUTING PROTOCOL

In this section, we will present the detailed design description of LVRP. LVRP assumes that sensor nodes are static while sink nodes can move freely in the network. Each sensor node and mobile sink are assumed to be equipped with an omnidirectional antenna. Wireless channels in the network are assumed to be bidirectional and symmetric. Each sensor node can obtain its one-hop neighbor node list. No location information of nodes is assumed. In this paper, we assume that each mobile sink node can move freely in the network field at relatively low speed.

In LVRP, each sensor node only needs to keep one routing entry to reach only one mobile sink at a given time. The routing entry kept at each sensor node contains the following information.

- *HopDist*, which is the hop distance from the sensor node to its target sink. For an invalid entry, its value is set to infinity.
- *LayerNum*, in which layer the sensor node is located within a Voronoi scope, to which the sensor node is currently belonging. Its initial value is Nil. Without causing confusion, hereafter, we will use the term 'layer' and 'level' interchangeably unless otherwise stated.
- *SinkID*, which is the target sink's identification (ID) for the sensor node to send packets. *SeqNum*, which is the sequence number assigned by the target sink associated with the sensor node and also represents the freshness of the routing entry. Each time a sink disseminates a new routing update message, the sequence number will increase by one.
- *NextHopID*, which is the ID of the next hop for the sensor to reach its target sink.

The procedures for LVRP contain four components: dynamic selection of anchor node, creation of layered Voronoi scope, dynamic update of layered Voronoi scope, and data packet forwarding. The following subsections describe the details of each of the four components.

3.1. Dynamic selection of anchor node

The procedure for dynamic selection of anchor node in LVRP works similarly to that in AVRP [10]. The major purpose of using anchor node is to hide the short distance movement of mobile sink. Besides, in LVRP, the selection of a new anchor node will also inform its associated mobile sink which layer the sink has entered and whether the sink has moved out of its Voronoi scope. The detailed procedure for dynamic selection of anchor node is as follows.

The anchor nodes are chosen to reduce the routing update frequency caused by the mobility of sink nodes. An anchor node is chosen to be the nearest neighbor sensor node of a mobile sink, and it is used to keep track of the corresponding mobile sink and gather sensor data on behalf of the corresponding mobile sink. Selection of the nearest neighbor node as the anchor node of a mobile sink can effectively reduce the frequency of anchor switching. A mobile sink only issues new routing update messages when it chooses a new anchor node.

DOI: 10.1002/dac

To choose an anchor node, a mobile sink first needs to broadcast a *request* message to its current neighbor sensor nodes. Upon receipt of the request message, each of its neighbor sensor nodes will reply with a *response* message that contains its *ID*. Then, the mobile sink can estimate who is the nearest neighbor node based on the received signal strength (RSS) among the received response messages. Then, the mobile sink sends an *anchor-req* message that contains its own *ID*, its current *SeqNum* and the value of layer width *K* to the selected anchor node, where *K* is used for each sensor node receiving a routing update message to compute with which layer it is currently belonging. After the anchor node receives the anchor-req message, it will first reply with an *anchor-ack* message back to the mobile sink to acknowledge its acceptance of the request and then update its routing entry as follows:

- $HopDist \leftarrow 1$,
- $LayerNum \leftarrow 1$,
- $SinkID \leftarrow sink.ID$,
- $NextHopID \leftarrow sink.ID$, and
- $SeqNum \leftarrow sink.SeqNum$.

Here, sink.*xxx* represents the *xxx* property associated with the sink triggering the anchor node selection. The newly selected anchor node will flood an interest packet to build a layered Voronoi scope for the mobile sink with which it is currently associated.

As the sink moves, the mobile sink and its associated anchor node will periodically exchange signaling messages to maintain the link between them while the mobile sink can check the RSS received from the anchor node. When the RSS is lower than a predefined threshold, the mobile sink will reselect a new anchor node among its neighbor sensor nodes and then refresh the routing delivery structure.

In LVRP, each mobile sink selects its anchor node independently. For a sensor node that is currently serving as an anchor node, it will not respond any further request message from mobile sink. In case a regular sensor node receives multiple request messages (asking for serving as anchor node) from two or more mobile sinks, the sensor node will only respond once. Specifically, it broadcasts a *reply* packet to all its one-hop neighbors (including the mobile sinks). Each mobile sink can estimate who is its closest sensor node based on its received reply packets and then sends (unicast) an anchor-req packet to its most preferred sensor node. Upon receipt of an anchor-req packet, a sensor node will serve as anchor node by replying with an anchor-ack packet if it has not accepted any other such request. If not receiving a corresponding anchor-ack packet in a certain time, a mobile sink will try its second closest neighbor sensor node (if any). This process continues until the mobile sink is attached to an anchor node or there exists no available neighbor sensor node that can sever as anchor node for the mobile sink.

3.2. *Creation of layered Voronoi scope*

In LVRP, mobile sinks will disseminate interest packets across the network via their respectively selected anchor nodes to build layered delivery structure covering the whole network using Voronoi scoping. An interest packet carries the following information:

- ID of the mobile sink that initiates the interest dissemination.
- Sequence number (*SeqNum*), which is assigned by the mobile sink issuing the interest message.
- Hop distance (*HopDist*), which records the hop distance from the sensor node forwarding the interest packet to the mobile sink issuing the interest message. Its initial value is one, which is for the anchor node.
- Layer width *K*, which is used for each sensor node to compute which layer it is currently belonging to. For a sensor x receiving an interest packet from a neighbor y, its layer number (denoted by $Layer_x$) can be calculated by as follows:

$$Layer_x = \left\lceil \left(HopDist_y + 1\right)/K \right\rceil \quad (1)$$

- Where $HostDist_y$ represents the hop distance from sensor node y to target sink.

DOI: 10.1002/dac

- The update scope parameter *UpdateLevel*, which indicates only those sensor nodes whose layers are not higher than the value of *UpdateLevel* will be updated. When a mobile sink attempts to construct or reconstruct its whole Voronoi scope, *UpdateLevel* is set to infinity.

In LVRP, the creation of layered Voronoi scope for a mobile sink can be triggered when either of the following cases occurs. One is that the network is in the initialization stage. The other is that a mobile sink moves outside of its current Voronoi scope. For the latter case, a mobile sink can judge whether it has moved out of its current Voronoi scope or not based on its newly selected anchor node. If the newly selected anchor node belongs to another sink's Voronoi scope, then the mobile sink will trigger the creation of a new layered Voronoi scope for itself. Figure 1(a) shows how the initial Voronoi scopes for a two-mobile sink network are built and how the initial layering of the sensor field associated with each mobile sink is done. In Figure 1(a), the vertical dotted straight line in the middle is the boundary of the Voronoi scopes of mobile sinks M_1 and M_2. Figure 1(b) gives an example illustrating how dynamic relayering in a Voronoi scope works as mobile sink M_2 moves. Figure 1(c) gives an example illustrating how Voronoi rescoping works when mobile sink M_2 moves out of its associated Voronoi scope.

When a sensor node receives an interest packet, it will update its routing entry and calculate its belonging layer number according to equation (1), if applicable. Figure 2 (steps 1–9) shows the pseudocodes for a sensor node *i* to execute when it receives an interest packet from a neighbor node *m*. First, sensor node *i* checks the routing update parameter *UpdateLevel* included in the interest packet, which will be set to infinity if the sink wishes to construct or reconstruct its whole Voronoi scope. Then sensor node *i* will accordingly update its routing entry provided that any one of the following three conditions (step 2 in Figure 2) is satisfied:

- Condition 1: Sensor *i*'s routing entry is empty.
- Condition 2: Sensor *i*'s old target sink in its routing entry is the same as that initiating the current interest dissemination, and (a) the interest packet has a *SeqNum* value greater than that recorded in sensor *i*'s routing entry, or (b) the packet has a *SeqNum* equal to that *i* locally stores but takes a path with shorter distance.
- Condition 3: The interest packet is originated from a different sink node but takes a path shorter than that stored in sensor *i*'s routing entry. Note that the length of the path taken by the received interest packet (till *i*) is interest. *HopDist* + 1 instead of interest.*HopDist*.

After sensor *i* accordingly updates its routing entry (steps 3–7), it will forward the interest packet further to its own neighbor sensor nodes (step 8). At the same time, sensor *i* also computes its own layer value (step 6) according to equation (1). After the diffusion of the interest packet terminates, a layered Voronoi scope for the mobile sink is formed.

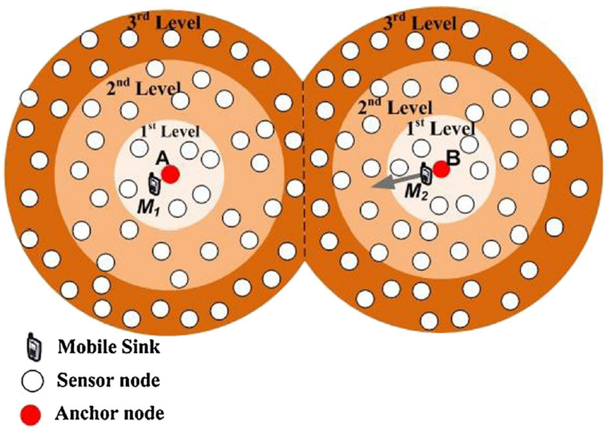
In LVRP, each sink creates its own Voronoi scope independently so that there is only one sink in each Voronoi scope and any two Voronoi scopes do not overlap. When a mobile sink (say *A*) enters another sink's Voronoi scope (say *B*), *A* needs to create a new Voronoi scope for itself, which covers *A*'s old Voronoi scope and also part of the current Voronoi scope of *B*. Accordingly, the Voronoi scope of *B* will shrink and that of *A* will expand. This process of Voronoi rescoping will continue as mobile sinks keep moving in the network.

3.3. Dynamic update of layered Voronoi scope

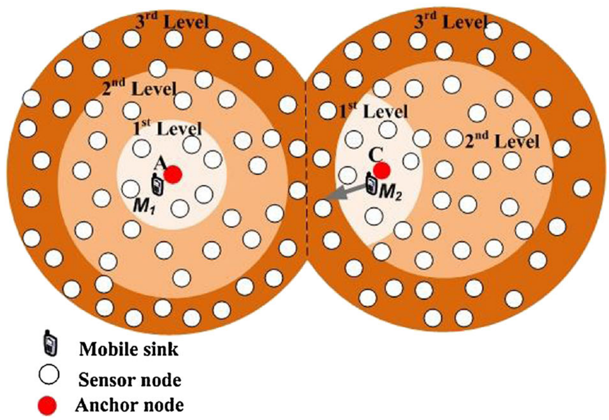
When a mobile sink moves within its current Voronoi scope but selects a new anchor node, it will refresh the data delivery structure within certain layers of its current Voronoi scope and relayer its current Voronoi scope, if applicable, to ensure the validity of data delivery path. The refreshing scope is decided by the layer number of the newly selected anchor node immediately before the refreshing. A mobile sink can recognize which layer of its current Voronoi scope it has entered via the layer number encapsulated in the response message received from the newly selected anchor. Then, the mobile sink will notify the newly selected anchor to flood an interest packet with the value of *UpdateLevel* indicating how to refresh its current Voronoi scope.

According to the value of *UpdateLevel* in the interest packet, there are the three cases to handle for the update of layered Voronoi scope.

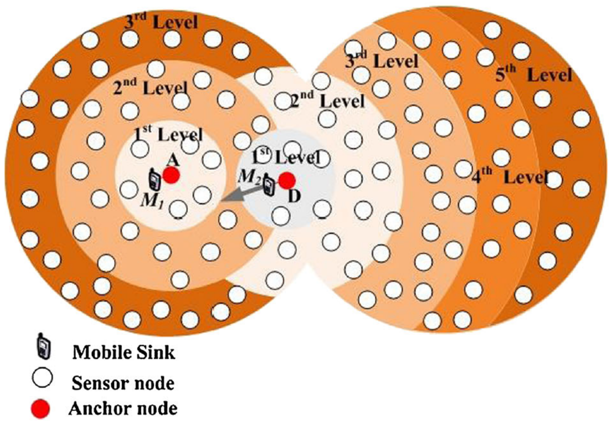
DOI: 10.1002/dac



(a) Creation of layered Voronoi scope in the network initialization stage. There are two mobile sinks M_1 and M_2 in the network.



(b) Updating of Layered Voronoi Scope when mobile sink M_2 leaves its first-layer area and enters higher-layer area of its old Voronoi Scope (see Figure1(a)).



(c) Updating of Layered Voronoi Scope when mobile sink M_2 moves out of its old Voronoi Scope and enters the Voronoi scope of M_1.

Figure 1. An example illustrating how the LVRP protocol works. (a) Creation of layered Voronoi scope in the network initialization stage. There are two mobile sinks M_1 and M_2 in the network. (b) Updating of layered Voronoi scope when mobile sink M_2 leaves its first layer area and enters higher layer area of its old Voronoi scope (Figure 1(a)). (c) Updating of layered Voronoi scope when mobile sink M_2 moves out of its old Voronoi scope and enters the Voronoi scope of M_1.

DOI: 10.1002/dac

```
// Executed when a sensor node i receives an interest message from a neighbor node m
RcvInterest(i,m)
1.   if (interest.UpdateLevel == INFINITY)  //Updating the whole Voronoi scope
2.     if ((node i's route table is empty)                                                        //condition 1
  2.1     or ( interest.sinkID == RouteEntry.sinkID and  ((interest.SeqNum > RouteEntry.SeqNum) or
           (RouteEntry.SeqNum == interest.SeqNum and  RouteEntry.HopDist > interest.HopDist + 1)) )  //condition 2
  2.2     or (RouteEntry.HopDist > interest.HopDist + 1))                                         //condition 3
3.       then RouteEntry.NextHopID ← m
4.            RouteEntry.SinkID    ← interest.SinkID
5.            RouteEntry.SeqNum    ← interest.SeqNum
6.            RouteEntry.LevelNum  ← ⌈(interest.HopDist+1) / K⌉
7.            RouteEntry.HopDist   ← interest.HopDist + 1
8             fwdInterest(i);  /*This function is executed after implementing interest.HopDist ← interest.HopDist +1 */
9.     end if
10.  else //re-layering those sensor nodes with layers ≤ interest.UpdateLevel only
11.    if  (interest.sinkID == RouteEntry.sinkID  and
          RouteEntry.LevelNum ≤ interest.UpdateLevel)
12.    then if  ((RouteEntry.SeqNum < interest.SeqNum) or
             (RouteEntry.SeqNum == interest.SeqNum and  RouteEntry.HopDist > interest.HopDist + 1))
13.       then if (⌈interest.HopDist+1/K⌉< interest.UpdateLevel)
14.            then RouteEntry.NextHopID   ← m.
15.                 RouteEntry.SeqNum      ← interest.SeqNum
16.                 RouteEntry.LevelNum    ←⌈(interest.HopDist+1)/K⌉
17.                 RouteEntry.HopDist     ← interest.HopDist + 1
18.                 fwdInterest(i)
19.            else
20.                 RouteEntry.NextHopID  ← m.
21.                 RouteEntry.HopDist    ← interest.HopDist
22.                 RouteEntry.SeqNum     ← interest.SeqNum
23.                 RouteEntry.LevelNum   ← interest.UpdateLevel
24.                 fwdInterest(i)
```

Figure 2. Procedures for layered Voronoi scoping and relayering of sensor field in case of sink mobility.

3.3.1. Case 1. The value of *UpdateLevel* is equal to one. This means that the mobile sink is still in its first layer Voronoi scope. In this case, the mobile sink only needs to refresh the data delivery structure within its first layer Voronoi scope instead of relayering its whole Voronoi scope. In order to further reduce the protocol overhead, a trail-based auxiliary optimization technique is introduced for this case. If the newly selected anchor node is a neighbor of the old anchor node, then the newly selected anchor node only needs to send a notify message to the old anchor node instead of triggering an interest dissemination. The old anchor node receiving the notify message from the new anchor node will update its *NextHopID* as the new anchor node. If the newly selected anchor node is not a neighbor of the old anchor node, then the newly selected anchor will compose an interest packet carrying the value of *UpdateLevel* ← 1, *SinkID*, and *SeqNum* and floods the interest for updating the delivery structure within the first layer Voronoi scope. In this case, when a sensor node x located in the first layer Voronoi scope receives such an interest packet from a neighbor y, if the received interest packet has a fresher sequence number or the same sequence number but shorter path, it will

DOI: 10.1002/dac

update its routing entry as follows (also including the *LayerNum* option and the *SinkID* option) and continue to forward the interest packet:

- $RouteEntry_x.NextHopID \leftarrow y$.
- $RouteEntry_x.SeqNum \leftarrow interest.SeqNum$.
- $RouteEntry_x.HopDist \leftarrow interest.HopDist + 1$.

Otherwise, the received interest packet will be dropped directly.

Because the aforementioned trail-based optimization technique can lead to the increase of the data forwarding path length, in LVRP, this optimization technique will be only used for the case of $UpdateLevel = 1$.

3.3.2. Case II. The value of UpdateLevel is greater than one but smaller than infinity. This means that the mobile sink has moved out of its first layer Voronoi scope and has entered a higher layer area of its own Voronoi scope. The newly selected anchor will compose an interest packet carrying the value of $UpdateLevel \leftarrow$ the new anchor's old *LayerNum*, that is, its layer value before the refreshing, *SinkID*, *SeqNum*, and $HopDist \leftarrow 1$, and then set its own *LevelNum* to one and flood the interest for relayering those sensor nodes located in layers $\leq$*UpdateLevel* and also updating the corresponding data delivery structure. In Figure 2, steps 10–24 show the detailed procedures for this case. Step 11 tells that the updating will only be restricted to the Voronoi scope of the same sink and further within layers $\leq$*UpdateLevel* carried in the interest packet. Step 12 shows that route updating will be enforced if the received interest packet has a fresher sequence number or the same sequence number but a shorter path. The difference between the routing table updating procedures in steps 14–18 and those in steps 20–24 are as follows. The former are for the case if sensor i is in (or on) $K \times (UpdateLevel\text{-}1)$ hops away from the new anchor node. The layer values of these sensors will be computed as usual (step 16). All other remaining nodes with levels $\leq$*UpdateLevel* will be set to the *UpdateLevel* level without considering how far they are away from the new anchor (step 23). Figure 1(b) illustrates how the relayering of sensor field works when mobile sink M_2 leaves its first layer and enters the second layer (refer to Figure 1(a) for its old first and second layers). As we can see, the relayering in Figure 1(b) only affects those sensor nodes with levels $\leq$2, and those sensor nodes in level 3 are not affected at all.

3.3.3. Case III. The value of *UpdateLevel* is infinity. This means that the mobile sink has moved out of its Voronoi scope and entered other sink's Voronoi scope. For this case, the mobile sink will reconstruct a new layered Voronoi scope for itself. The new Voronoi scope will naturally cover all those sensor nodes in the sink's old Voronoi scope but may contain some new sensor nodes, previously belonging to other neighbor sinks' scopes. Figure 1(c) shows an example illustrating how this works. In Figure 1(c), after mobile sink M_2 moves out of its old Voronoi scope and it selects a new anchor node D, which belongs to the Voronoi scope of mobile sink M_1, then the newly selected anchor node D will first update its routing table and set the value of *UpdateLevel* to be carried by interest packet as infinity and then flood the interest packet to build a brand new Voronoi scope for sink M_2, which now contains five levels. As can be seen, the newly issued interest message will result in a new Voronoi scope covering all those sensor nodes in M_2's old Voronoi scope and also some sensor nodes previously belonging to M_1's Voronoi scope, if such change will lead to shorter path distance from these sensor nodes to M_2 than that to M_1.

As shown in the example in Figure 1 and also the aforementioned three cases, it can be easily seen that in some cases, the path from a sensor node to its target sink may not be the shortest path connecting them because of the restriction enforced on the routing updating scopes as mobile sinks move, in order to reduce the protocol overhead for route management. In some cases, a sensor node even may not be in the Voronoi scope of its closest sink. For example, in Figure 1(a), if mobile sink M_2 chooses to move rightwards away from its old anchor position B, then some sensor nodes sitting immediately right hand to the boundary between the Voronoi scopes of M_1 and M_2 would keep belonging to the Voronoi scope of M_2, although its actual closest sink might have changed to M_1. This is because M_1 has not triggered any new routing update because it has not moved.

DOI: 10.1002/dac

3.4. Data packet forwarding

In LVRP, when a data source has a data packet to report, it will deliver the packet towards the target sink stored in its routing table via hop-by-hop forwarding. Specifically, each packet holder (except anchor nodes) will forward the packet to its next hop until the packet reaches the sink. For an anchor node, before it forwards data packets to its associated mobile sink, it will first check whether its routing table is valid or not. When the RSS between an anchor node and its associated mobile sink is lower than a threshold and the associated mobile sink needs to reselect a new anchor node, the anchor node will first mark its routing table invalid and buffers the received data packets from neighbor nods until it receives a fresher interest message from a newly selected anchor node to refresh its routing table.

It is worthy pointing out that during the hop-by-hop packet forwarding in LVRP, the layers to be visited by a data packet monotonically decrease until it reaches a sink, which can effectively avoid routing loops.

3.5. Route optimization technique

In LVRP, as the refreshing scope of data delivery structure is maximally restricted to certain layers for refreshing a Voronoi scope, if possible, it may lead to the increase of the data routing path's length. A routing update optimization technique is introduced to alleviate this impact, which takes advantage of the broadcast feature of wireless medium. The procedures for this optimization technique are shown in Figure 3. Specifically, when a sensor node i located in the ($UpdateLevel + 1$)th layer overhears an interest packet from its neighbor node m whose layer is less than or equal to the value of *UpdateLevel* in the interest packet, if the neighbor node m sending the interest packet containing a path more than one hop shorter than that recorded in the sensor node i's routing entry (step 1 in Figure 3), then sensor node i will update its routing entry (step 2–3 in Figure 3).

4. PERFORMANCE EVALUATION

In this section, we evaluate the performance of LVRP using NS-2.31 [29]. In the simulations, we compared the following four protocols: the LVRP protocol proposed in this paper, AVRP [14], TRAIL [14], and DDRP [15]. A common feature of all these four protocols is that none of them requires location information of nodes in their respective protocol implementations, and all of them can provide timely data delivery services in mWSNs.

4.1. Simulation scenarios and metrics

In our simulations, 1300 sensor nodes were deployed uniformly initially in a $1000 \times 1000\,m^2$ area. Sensor nodes transmit packets by using the unslotted carrier sense multiple access with collision avoidance channel access mechanism under the IEEE 802.15.4 nonbeacon-enabled mode. The uniform communications range of all nodes (including both sensors and mobile sinks) is 50 m. The channel rate is 250 kbp. The random waypoint model [30] is used as the mobility model for mobile sink. The traffic pattern is periodic reporting such that each source sensor reports its sensed data periodically. The size of data packets and that of control packets are both 50 bytes. For each experiment, the simulation duration is set to 1000 s. The value of layer width K is tuned via extensive experiments

// Executed when a sensor node i overhears an *interest* message from a neighbor node m

OverhearInterest(i,m)

1. **if (**RouteEntry.*HopDist* > interest.*HopDist +1*)
2. **then** RouteEntry.*NextHopID* ← m.
3. RouteEntry.*HopDist* ← interest.*HopDist + 1*
4. **end if**

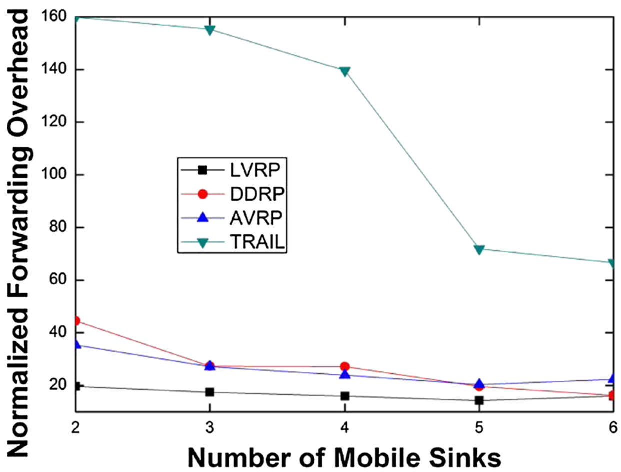
Figure 3. Procedures for routing update optimization technique.

DOI: 10.1002/dac

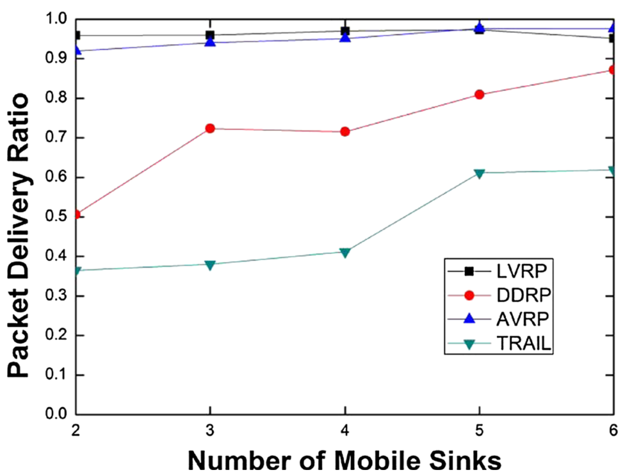
in different simulation scenarios. On the basis of our simulation results, K varies between 2 and 3 for achieving low protocol overhead. Note that larger K will lead to reduced route updating frequency, high protocol overhead for each route update, bigger deviation (on average) away from shortest path distance for data reporting. When K is sufficiently large (compared with the network diameter), LVRP degenerates to AVRP [14], in which case K will have no impact on routing performance.

In our simulations, we used the following two metrics for performance evaluation:

- Normalized forwarding overhead [31] is defined as the ratio of the total number of all packets transmitted (containing data and control packets) to the number of successfully delivered data packets. The number of packets transmitted is the total number of packet transmitted by all nodes (containing sensor nodes and sink nodes), and it also includes those transmissions of the packets that are forwarded or dropped or collided. The smaller the value of this parameter is, the lower the protocol overhead will be and the more efficient a protocol will be.
- Packet delivery ratio is defined as the ratio of the total number of data packets successfully received by sinks to the number of data packets generated by the sensor nodes. This metric represents the data delivery efficiency of a routing protocol.



(a) Normalized forwarding overhead versus number of mobile sinks.



(b) Packet delivery ratio versus number of mobile sinks.

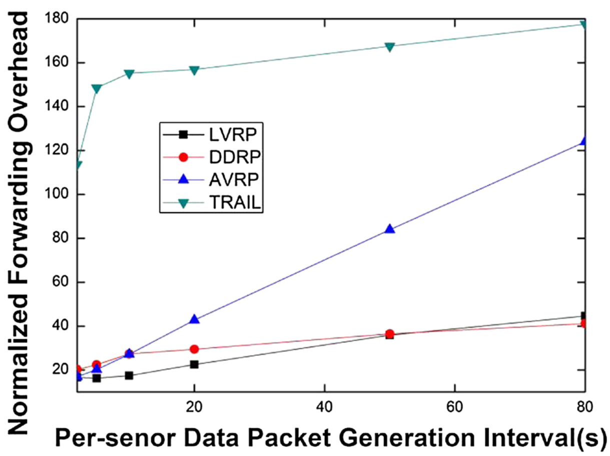
Figure 4. Comparison of routing performance by different protocols versus the number of mobile sinks. (a) Normalized forwarding overhead versus the number of mobile sinks. (b) Packet delivery ratio versus the number of mobile sinks.

DOI: 10.1002/dac

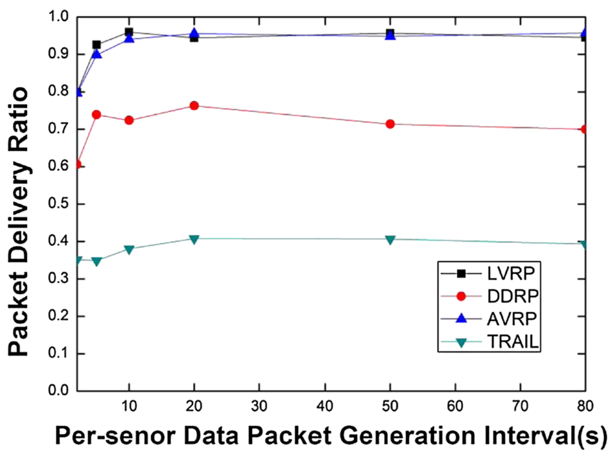
4.2. Simulation results

4.2.1. Impact of the number of mobile sinks. In this experiment, the number of mobile sinks in the network varies from 2 to 6. The per-sensor data packet generation interval is set to 10 s. The velocity of each mobile sink ranges from 1 to 5 m/s. The number of data sources is 100.

Figure 4 compares the normalized forwarding overhead and delivery ratio performance of the four protocols versus the number of mobile sinks. Figure 4(a) shows that LVRP has lower protocol overhead as compared with the other three routing protocols for mWSNs especially when the number of mobile sinks is relatively small. Meanwhile, Figure 4(b) shows that LVRP can keep high delivery success ratio. For TRAIL and DDRP, with the increase of the number of mobile sinks, the probability for a sensor node to discover valid routes will increase, which may lead to lower protocol overhead and higher delivery success ratio. However, in the case that the number of data sources is relatively small and the network size is relatively large, TRAIL and DDRP are not efficient as compared with AVRP and LVRP because of the use of random walk routing in the former two protocols when no route is known. Although as the number of mobile sinks increases, the protocol overhead by DDRP is closer to that of LVRP; the delivery success ratio of DDRP is lower than that of LVRP. For AVRP, its average diffusion scope of interest flooding for Voronoi scoping is larger than that of LVRP, which leads to higher protocol overhead.



(a) Normalized forwarding overhead versus traffic load.



(b) Packet delivery ratio versus traffic load.

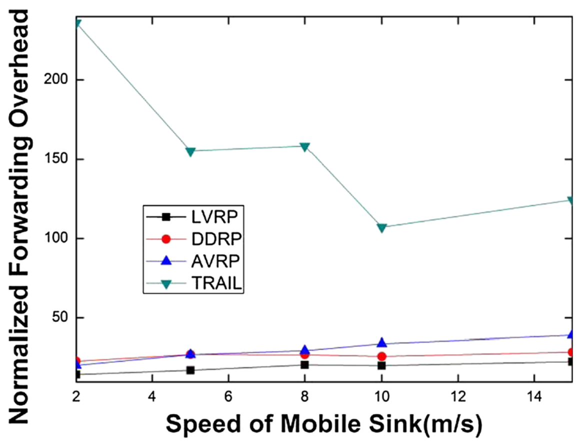
Figure 5. Comparison of routing performance by different protocols versus traffic load. Note that traffic load is inversely proportional to the per-sensor data packet generation interval. (a) Normalized forwarding overhead versus traffic load. (b) Packet delivery ratio versus traffic load.

DOI: 10.1002/dac

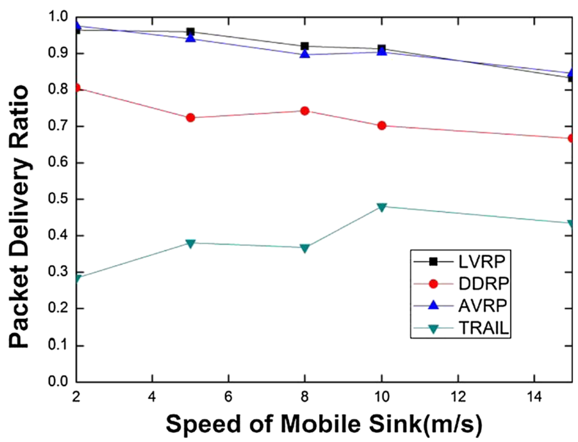
4.2.2. Impact of traffic load. In this experiment, the per-sensor interpacket generation interval ranges from 2 to 80 s. The velocity of mobile sinks ranges from 1 to 5 m/s. The number of mobile sinks is three. The number of data sources is 100.

Figure 5 compared the normalized forwarding overhead and delivery ratio performance of the four protocols versus traffic load. Figure 5(a) and (b) shows that LVRP has lower protocol overhead compared with other there routing protocols while ensuring high delivery success ratio. The reasons are as follows:

- For AVRP, its resulting control overhead depends heavily on the frequency at which dynamic Voronoi scoping (rescoping) is triggered, which is only related to the number and velocity of mobile sinks. As a result, as the traffic load decreases (i.e., as the per-sensor interpacket generation interval increases), the ratio of control packets to all packets will increase. Thus, the normalized protocol overhead by AVRP increases quickly with the per-sensor interpacket generation interval.
- For TRAIL, its high forwarding overhead is due to the frequent enforcement of random walk routing, which causes long data delivery paths and more dropped data packets.
- For DDRP, in a large mWSN with a small amount of data sources, the probability for a sensor node to learn valid routing information will largely decrease, which could lead to high forwarding overhead or low delivery success ratio.



(a) Normalized forwarding overhead versus sink velocity.



(b) Packet delivery ratio versus sink velocity.

Figure 6. Comparison of routing performance by different protocols versus sink velocity. (a) Normalized forwarding overhead versus sink velocity. (b) Packet delivery ratio versus sink velocity.

DOI: 10.1002/dac

• For LVRP, whether the traffic load is light or heavy, it can effectively limit the number of control packets for refreshing data delivery structure by reducing the refreshing scopes and frequencies.

4.2.3. Impact of sink mobility. In this experiment, the maximum velocity of mobile sinks (denoted by MAX_SPEED) varies from 2 to 15 m/s. Accordingly, in the simulations, the velocity of each mobile sink is uniformly distributed in the range of [1, MAX_SPEED] meters per second. The per-sensor data packet generation interval is 10 s. The number of mobile sinks is three. The number of data sources is 100.

Figure 6 compares the routing performance of various protocols versus sink velocity. In Figure 5, it is seen that LVRP has a lower protocol overhead (Figure 6(a)) compared with the other three routing protocols no matter if the sink velocity is low or high. Meanwhile, LVRP can keep high delivery ratio (Figure 6(b)). For DDRP and TRAIL, in a large mWSN with a small amount of data sources, the probability for a sensor node to learn or discover valid route will largely decrease, which could lead to high forwarding overhead and low delivery success ratio. For AVRP and LVRP, with the increase of sink velocity, the frequent reconstruction of whole Voronoi scope (for the former) or part of it (for the latter) leads to higher protocol overhead. Because LVRP can

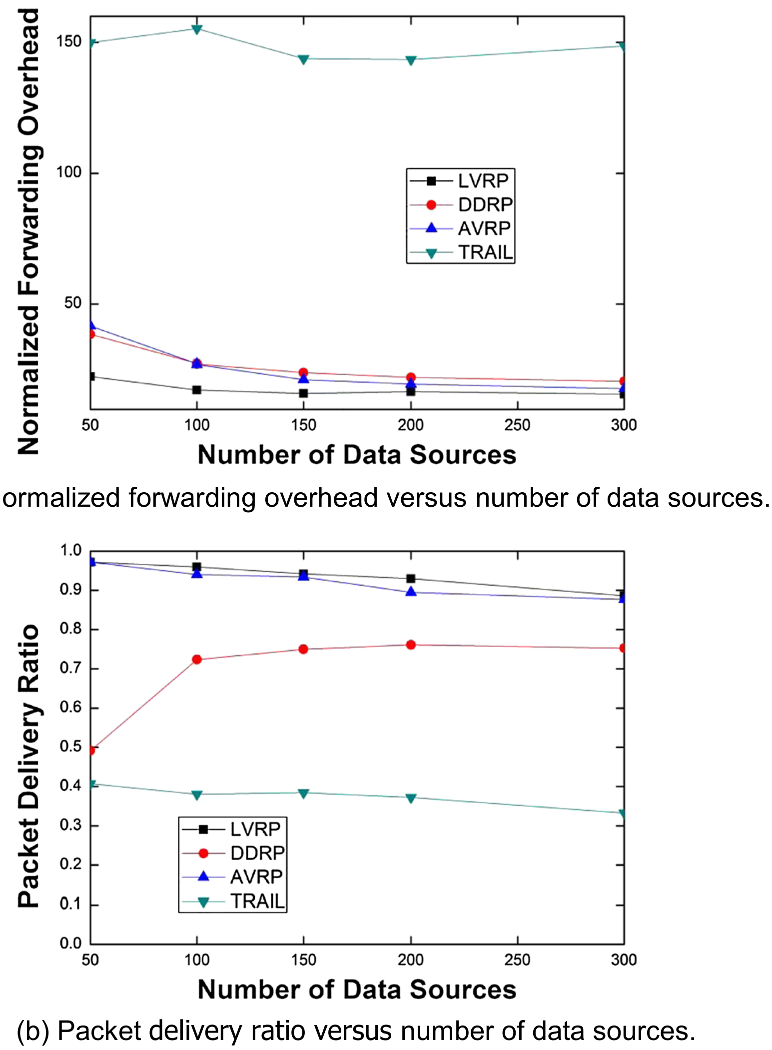


Figure 7. Comparison of routing performance by different protocols versus the number of data sources. (a) Normalized forwarding overhead versus number of data sources. (b) Packet delivery ratio versus number of data sources.

DOI: 10.1002/dac

effectively control the refreshing range (and also frequency), which can more effectively reduce the protocol overhead compared with AVRP.

4.2.4. Impact of the number of data sources. In this experiment, the number of data sources varies from 50 to 300. The per-sensor data packet generation interval is 10 s. The number of mobile sinks is three. The velocity of mobile sinks ranges from 1 to 5 m/s.

Figure 7(a) and (b) shows that LVRP has a lower forwarding overhead compared with the other three routing protocols. For TRAIL and DDRP, with the number of data sources increasing, the probability for a sensor node to learn or discover valid routing information will increase, which could lead to low forwarding overhead and high delivery success ratio. However, when the number of data sources is not high enough, these two routing protocols is inefficient because of the frequent use of random walk routing. For AVRP and LVRP, with the number of data sources increasing, the ratio of control packets to all packets will increase. Thus, the normalized protocol overhead by AVRP and LVRP decreases with the increase of the number of data sources. Because LVRP can effectively limit the amount of control packets, it has a lower protocol overhead as compared with AVRP.

5. CONCLUSION

In this paper, we have designed an efficient distributed routing protocol called LVRP for mWSNs. LVRP exploits dynamic layered Voronoi scoping and dynamic anchor selection in its implementation to effectively reduce the protocol overhead caused by sink mobility. LVRP further introduces certain localized optimization techniques to reduce the protocol overhead. Simulation results demonstrate that LVRP can effectively reduce the protocol overhead compared with existing protocols while keeping high packet delivery ratio under various situations.

ACKNOWLEDGEMENTS

This work was supported in part by the National Natural Science Foundation of China under grant nos. 61101133, 61173158, and 61271041; the National Key Special Program of China under grant no. 2010ZX03006-001-02; and the Natural Sciences and Engineering Research Council (NSERC) of Canada (Discovery Grant 293264-12 and Strategic Project Grant STPGP 397491-10).

REFERENCES

1. Ekici E, Gu Y, Bozdag D. Mobility-based communication in wireless sensor networks. *IEEE Communications Magazine* 2006; **44**(7):56–62.
2. Luo J, Hubaux J. Joint mobility and routing for lifetime elongation in wireless sensor networks. *Proc. IEEE INFOCOM'05*, Miami, FL, USA, March 2005; 1735–1746.
3. Shah R, Roy S, Jain S, Brunette W. Data mules: modeling a three-tier architecture for sparse sensor networks. *Proc. IEEE SNPA'03*, Anchorage, AK, USA, June 2003; 30–41.
4. Cheng L, Chen C, Ma J, Shu L. Contention-based geographic forwarding in asynchronous duty-cycled wireless sensor networks. *International Journal of Communication Systems* 2012; **25**(12):1585–1602.
5. Tang J, Dai S, Li J, Li S. Gossip-based scalable directed diffusion for wireless sensor networks. *International Journal of Communication Systems* 2011; **24**(11):1418–1430.
6. Bai F, Munasinghe K, Jamalipour A. Accuracy, latency, and energy cross-optimization in wireless sensor networks through infection spreading. *International Journal of Communication Systems* 2011; **24**(5):628–646.
7. Huang HJ, Hu GM, Yu FC. Energy-aware geographic routing in wireless sensor networks with anchor nodes. *International Journal of Communication Systems* 2013; **26**(1):100–113.
8. Peiravi A, Mashhadi AHR, Javadi SH. An optimal energy-efficient clustering method in wireless sensor networks using multi-objective genetic algorithm. *International Journal of Communication Systems* 2013; **26**(1):114–126.
9. Wang Y, Shi PZ, Li K, Chen ZK. An energy efficient medium access control protocol for target tracking based on dynamic convey tree collaboration in wireless sensor networks. *International Journal of Communication Systems* 2012; **25**(9):1139–1159.
10. Jiang FC, Huang DC, Yang CT, Lin CH, Wang KH. Design strategy for optimizing power consumption of sensor node with Min(N,T) policy M/G/1 queuing models. *International Journal of Communication Systems* 2012; **25**(5): 652–671.

11. Francesco MD, Das SK, Anastasi G. Data collection in wireless sensor networks with mobile elements: a survey. *ACM Transactions on Sensor Networks* 2011; **8**(1):1–31.
12. Li X, Yang J, Nayak A, Stojmenovic I. Localized geographic routing to a mobile sink with guaranteed delivery in sensor networks. *IEEE Journal on Selected Areas in Communications* 2012; **30**(9):1719–1729.
13. Khan A, Madani S, Hayat K, Khan S. Clustering-based power-controlled routing for mobile wireless sensor networks. *International Journal of Communication Systems* 2012; **25**(4):529–542.
14. Tian K, Zhang B, Huang K, Ma J. Data gathering protocols for wireless sensor networks with mobile sinks. *Proc. IEEE GLOBECOM'10*, Miami, FL, USA, December 2010; 1–6.
15. Shi L, Zhang B, Huang K, Ma J. An efficient data-driven routing protocol for wireless sensor networks with mobile sinks. *Proc. IEEE ICC'11*, Kyoto, Japan, June 2011; 1–5.
16. Wang G, Wang T, Jia W, Guo M, Chen H, Guizani M. Local update-based routing protocol in wireless sensor networks with mobile sinks. *Proc. IEEE ICC'07*, Glasgow, Scotland, June 2007; 3094–3099.
17. Ye F, Luo H, Cheng J, Lu S, Zhang L. A two-tier data dissemination model for large-scale wireless sensor networks. *Proc. ACM MobiCom'02*, Atlanta, Georgia, USA, September 2002; 148–159.
18. Faheem Y, Boudjit S, Chen K. Data dissemination strategies in mobile sink wireless sensor networks: a survey. *Proc. IEEE IFIP Wireless Days 2009*, Paris, France, December 2009; 1–6.
19. Dubois-Ferriere H, Estrin D, Stathopoulos T. Efficient and practical query scoping in sensor networks. *Proc. IEEE MAHSS'04*, October 2004; 564–566.
20. Wang G, Wang T, Jia W, Guo M, Li J. Adaptive location updates for mobile sinks in wireless sensor networks. *Journal of Supercomputing* 2009; **47**(2):127–145.
21. Yu F, Park S, Lee E, Kim S. Elastic routing: a novel geographic routing for mobile sinks in wireless sensor network. *IET Communications* 2010; **4**(6):716–727.
22. Yan Y, Zhang B, Zheng J, Ma J. Hierarchical location service for wireless sensor networks with mobile sinks. *Wiley Wireless Communications and Mobile Computing* 2010; **10**(7):899–911.
23. Lee E, Park S, Yu F, Choi Y, Jin M, Kim S. A predictable mobility-based data dissemination protocol for wireless sensor networks. *Proc. IEEE AINA'08*, Okinawa, Japan, March 2008; 741–747.
24. Shin K, Kim S. Predictive routing for mobile sinks in wireless sensor networks: a milestone-based approach. *Journal of Supercomputing* 2012; **62**(3):1519–1536.
25. Yu J, Jeong E, Jeon G, Seo D, Park K. A dynamic multiagent-based local update strategy for mobile sinks in wireless sensor networks. *Proc. IEEE ICCSA*'11, Santander, Spain, June 2011; 185–196.
26. Basagni S, Carosi A, Melachrinoudis E, Petrioli C, Wang Z. Controlled sink mobility for prolonging wireless sensor networks lifetime. *Wireless Networks* 2008; **14**(6):710–720.
27. Bi Y, Sun L, Ma J, Li N, Khan I, Chen C. HUMS: an autonomous moving strategy for mobile sinks in data-gathering sensor network. *EURASIP Journal on Wireless Communications and Networking* 2007; **2007**, DOI:10.1155/2007/64574, Article ID 64574.
28. Hamida E, Chelius G. A line-based data dissemination protocol for wireless sensor networks with mobile sink. *Proc. IEEE ICC'08*, Beijing, China, May 2008; 2201–2205.
29. http://www.isi.edu/nsnam/ns/ [accessed on 26 March 2014].
30. Johnson D, Maltz D. Dynamic source routing in ad hoc wireless networks. In *Mobile Computing*, Imielinski T, Korth H (eds), chapter 5, Kluwer Academic Publishers, 1996; 153–181.
31. Gerla M, Lee Y, Park J, Yi Y. On demand multicast routing with unidirectional links. *Proc. IEEE WCNC' 05*, New Orleans, LA, USA, March 2005; 2162–2167.

DOI: 10.1002/dac