Relay Hop Constrained Rendezvous Algorithm for Mobile Data Gathering in Wireless Sensor Networks

Wenjun Liu, Jianxi Fan⋆, Shukui Zhang, and Xi Wang

School of Computer Science and Technology, Soochow University,
Suzhou, 215006, China
{w-jliu,wangxi0414}@163.com, {jxfan,zhansk}@suda.edu.cn

Abstract. Recent research shows that significant energy saving can be achieved in wireless sensor networks (WSNs) by introducing mobile collector (MC). One obvious bottleneck of such approach is the large data collection latency due to low mobile speed of MC. In this paper, we propose an efficient rendezvous based mobile data gathering protocol for WSNs, in which the aggregated data will be relayed to Rendezvous Node (RN) within bounded hop d. The algorithm design in the protocol jointly considers MC tour and data routing routes in aggregation trees. The effectiveness of the approach is validated through both theoretical analysis and extensive simulations.

Keywords: Wireless Sensor Networks, NP-Hard, Mobile collector, Rendezvous node.

1 Introduction

In recent years Wireless Sensor Networks (WSNs) have become an attractive technology for a large number of applications, ranging from monitoring [1], localization [2], to target tracking [3]. To design and deploy sophisticated WSNs, many issues need to be resolved such as node deployment, energy conservation, routing in dynamic environments, and so on. Specifically, most of these existing solutions for data collection take advantage of multi-hop routing to relay data. One obvious drawback of this schema is that it leads to unbalanced energy consumption among the sensors on the transmission path to sink [4].

Recent research has shown a rapid transition from traditional data gathering pattern to introduction of mobile elements, which can improve energy efficiency, connectivity, and reliability effectively [5]. A typical application scenario is that a forest ranger who equipped with handheld device roams in the network and gathers the information of detective area. In such an application, mobile user can visit different regions in the network and communicate with the sensors nearby in single hop paradigm, which reduces not only contention and collisions, but also the message loss. However, due to the low velocity of the mobile collector,

---

⋆ Corresponding author.

usually 0.5-2m/s, e.g. Packbot [6], it would incur long latency if every node is traveled in data gathering [5]. Obviously, it can not meet the requirement of time-sensitive applications.

In order to shorten data gathering latency, integrating multi-hop transmission scheme into mobile data gathering is an effective approach [7], in which a subset of sensors is selected as Rendezvous Node (RN). In this pattern, at every transmission hop each node aggregates the local data from its affiliated sensors, until delivers to RN which caches and uploads data to the mobile collector (MC) when it arrives. However, it is necessary that the transmission hop should be constrained as a proper level for several reasons. First, energy saving is considered as the most important concern in WSNs. Adopting multi-hop routing to relay data can easily result in unbalanced energy consumption among sensors, and it is adverse for energy-limited nodes. Second, a big relay hop means that the node acting as RN should have high performance to aggregate and cache data before MC arrives. Third, time-sensitive applications often require the sensing data to be delivered with certain deadline. For instance, in the application of forest fire monitoring, the fire should be detected and reported instantly.

The main contributions of this paper can be summarized as follows: 1) We define the mobile data gathering problem based on RN as MDG-RN, which jointly considers MC tour and routes in aggregation trees, and prove it is NP-Hard. 2) We develop two efficient algorithms to solve the MDG-RN problem. The former is a heuristic algorithm which always selects the node with maximum load from the d-hop neighbors of the current farthest node to BS. The latter caters to the characteristic of WSNs, and selects RN iteratively in distributed manner. On the basis of selected RNs, using algorithm for Traveling Salesman Problem (TSP) to produce MC tour, along which MC periodically visits these RNs and picks up the cached data. 3) Simulation results show that both algorithms can achieve satisfactory performance comparing with existing schemes.

The rest of the paper is organized as follows: Section 2 reviews related work. Section 3 introduces the network model and problem definition. The major contributions are introduced in Section 4 and Section 5. The simulation results are presented in Section 6. Finally, Section 7 concludes this paper.

2   Related Work

Recently, many research efforts have appeared in the literature to explore mobility-enable data collection in sensor networks [5-13]. These approaches may be classified as uncontrollable or controllable in general [5]. The former is obtained by attaching a collector node on certain mobile entity such as an animal or a bus; the latter is achieved by intentionally adding a mobile entity e.g., a mobile robot or an unmanned aerial vehicle, into the network to carry the collector. Clearly, a controlled mobility gives more flexibility for designing a data collection scheme.

The major performance bottleneck of such mobility-enabled WSNs is the increased latency in data collection. There are many approaches address to the

delay problem. The first category is using the single hop transmission scheme. It is not difficult to conclude that direct-contact data collection is generally equivalent to the NP-complete TSP. Nesamony et al. [8] formulated the traveling problem as TSP with Neighborhood, where a MC needs to visit the neighborhood of each sensor exactly once. He et al. in [9] proposed a progressive optimization approach, called CSS, to reduce the tour length, and thus the data collection latency. This kind of approach minimizes the network energy consumption by one hop transmission, but it incurs high latency when collecting data from large sensing fields due to the slow speed of MC.

In second category multi-hop transmissions is adopted. Ma et al. [10] gave a moving path planning algorithm by finding some turning points, which is adaptive to the sensor distribution and can effectively avoid obstacles on the path. In [11], Gatzianas et al. optimized data gathering performance by presenting a distributed maximum lifetime routing algorithm, where a mobile collector sequentially visits a set of anchor points and each sensor transmits data to the mobile collector at some anchor points via multi-hop paths. Such type of approach reduces latency effectively. However, without the hop count constraint, the unbalanced energy consumption leads to untimely network partition.

The last category is a hybrid approach with constraint conditions that usually jointly considers multi-hop data transmissions and the moving tour of MC in data collection. Xing et al. [12] proposed a rendezvous-based data collection approach under the constraint that the tour length of the mobile collector is no longer than a threshold. With the relay hop constraint, Zhao et al. [13] proposed a polling-based mobile data gathering scheme that minimize the tour length of MC and data gathering latency. They give two algorithms to find a set of PPs among sensors. In [7], Rao et al. establishes bounds for multi-hop routing as a function of sensor and MC parameters such as data generation rate, sink speed and sensor density. They developed a framework to parameterizes multi-hop routing using a hop-bound factor k. Their model revealed that for stable mobile sink operation, there exists a feasible range of the hop-bound factor k. The approach studied in this paper falls into this category.

3   Preliminary

3.1   Network Model

We assume N sensor nodes are scattered randomly over the interest area and left unattended to continuously sense and report events. There is a static BS located in the center of sensing area and a mobile collector (MC) moved in controlled mobility. MC knows its own physical locations through the GPS units on it. However, for generality, we do not make such assumption on sensor nodes. Under the consideration of same communication range, the communication links are symmetric. We consider WSN as a undirected graph $G(V, E)$, where the vertex set V represents all the sensors and the edge set E represents the communication links. Two vertices, u and v in V, are *adjacent* if there is a edge $e=(u,v)\in E$, then we say u is neighbor of v, and vice versa. A *path* $P = <v_1, v_2, \cdots, v_l >$ of

length l - 1 for $l \geq 2$ in G is a sequence of distinct vertices such that any two consecutive vertices are adjacent. The *neighbors* of a vertex v, denoted by $N(v)$, is the set of all vertices adjacent to v in G. The *d-hop neighbors* of node v is denoted by $d\text{-}N(v)$.

In data-centric WSNs, data from sources will be sent to RN or BS continuously, thus the data routes should be created in advance. On the basis of underlying topology G, a set of directed aggregation tree $T = \{T_i\}$ represents logic communication topology. For any T_i, $0 < i < N$, its root is the node r_i in RN. For any link $e \in G$, the communication cost is represented by its Euclidean distance. In addition,we assume the N-to-one aggregation model is adopted, in which a node can aggregate multiple data packets it received into one packet before relaying it [14].

3.2   Definitions

In the data collection schema, the RNs cache the data originated from sources and send to the MC via short-range transmissions when it arrives. The requirement is that the total length of MC tour should be minimized under the relay hop constraint. We refer to this problem as Mobile Data Gathering based on Rendezvous Nodes (MDG-RN) which is defined as follow:

Definition 1. *Given a set of sensors $S = \{s_1, s_2, \ldots, s_N\}$ and relay hop d, look for 1) A set of RN R; 2) A MC tour U connected all nodes in R and BS such that $\sum_{(u,v)\in U} |uv|$ is minimized, where (u,v) is a line segment on U and $|uv|$ is its Euclidean distance; 3) A set of aggregation trees $\{T_i(V_i, E_i)\}$ with height at most d that are rooted at $r_i \in R$ such that $\cup_i V_i = S$ and $\sum_i \sum_{(u,v)\in E_i} |uv|$ is minimized.*

From the definition of MDG-RN problem, the distribution of RNs and the data routes in each aggregation tree with the hop constraint should be jointly considered in order to find optimal solution. Thus the MDG-RN problem in this case can be formulated as:

$$Minimize \sum_{(u,v \in U)} |uv| \tag{1}$$

Subject to

$$c_{s_i,r_i}^h = 1, \ \forall s_i \in S, \ \forall r_i \in R, \ 0 \leq h \leq d. \tag{2}$$

$$\sum_{r_i \in R} c_{s_i,r_i}^h = 1, \ \forall s_i \in S, \ 0 \leq h \leq d. \tag{3}$$

$$|s_i r_i| \leq |s_i r_j|, \ \forall s_i \in S, \ \forall r_i \in R, \ r_i \neq r_j \tag{4}$$

For nodes $r, v \in V$ in G, we claim r *covers* v, if there is a path from v to r. A node v is *d-hop covered* by r if this path has the length no lager than d, written as $c_{v,r}^d$. A set of sensors are covered by r means an aggregation tree rooted at r is produced. The *d*-hop cover guarantees that any packet from the sources can be

sent to rendezvous node r within d hops. As aforementioned, in ideal N-to-one aggregation mode, the total length of communication edges in an aggregation tree is more worthy of attention comparing with the number of nodes associated with RNs. We define the transmission cost as load formally. The *load* of a node v, written as $\text{Load}(v)$, is the total edge length associated with it in aggregation tree T in network. If the height of aggregation tree rooted at v is d then its load is called *d-hop load*, written as $d\_\text{Load}(v)$. For node v, its *uncovered d-hop load* is the d-hop load except the edge length connecting the covered nodes in T, and it is written as $\text{unC}\_d\_\text{Load}(v)$.

Theorem 1. *The MDG-RN problem is NP-hard.*

Proof. This problem can be shown to be NP-hard by a polynomial-time reduction from the Euclidean Traveling Salesman Problem. Specifically, a special case of the decision version of MDG-RN problem is to ask if there exists a set of RNs such that all the sources must be RNs. This can be done by modifying the node transmission range R_t. When R_t is small enough, nodes are unreachable from each other. In such case, the relay hop d is equal to 0, and then MC must visit all the RNs to collect data. Thus the MDG-RN problem is NP-hard.   □

4   Algorithm for MDG-RN Problem

Due to the NP-hardness of the MDG-RN problem, in this section, we develop a Load Priority based RN Determination Algorithm (LP-RDA) for this problem. The basic idea of algorithm is to determine a set of RNs such that its total number is minimum and its distribution is near the BS as much as possible under the constraint of relay hop counts, and that the load of RNs is also optimized. The LP-RDA algorithm can be described as the following 3 steps:

```
STEP 1: INITIALIZATION
for any node sᵢ ∈ S
    sᵢ.status := not_Covered;
    computes sᵢ.dist_to_BS base on received signal strength;
    sends FB_Msg(sᵢ.ID, dist_to_BS, hopC) to BS hop-by-hop;
```

At beginning, static BS broadcasts "BEACON" message network-wide at a certain power. Each node computes the approximate distance to BS, dist_to_BS, base on the received signal strength. After that, every node sends message FB_Msg() to BS hop-by-hop. BS obtains the information of nodes in network after receiving these feedback messages.

```
STEP 2: ITERATION
BS determines a appropriate starting node x;
for any node sⱼ ∈ d-N_G(x)
    find a new RN rᵢ which has maximum unC_d_Load(sⱼ).
    rᵢ sends Declar_Msg() to d-N(rᵢ) and rⱼ.status:= Covered;
```
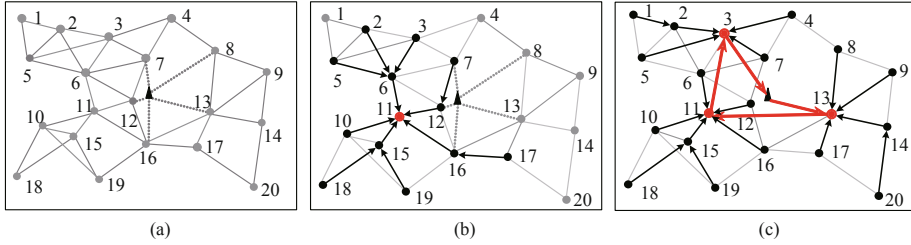
Fig. 1. An example to illustrate LP-RDA algorithm (N=20, d=2). (a) Initial topology. (b) The cover after the first RN is determined. (c) The final RNs, aggregation trees, and MCs tour.

```
for any s_j receiving Declar_Msg() from r_i
    if s_j.status = not_Covered then
        it joins its RN and s_j.status := Covered;
```

The iterative process to determine node status as shown in STEP 2. Depending on whether it is d-hop covered by RN, each node is set in one of two states: "Covered" or "not_Covered". Initially, all nodes are in the state of "un_Covered". The node with maximum $hopC$, x, is selected as the starting node. For the same $hopC$, the node with little dist_to_BS is selected. For d-$N(x)$, each node computes its unC_d_Load() by local message exchange. Next, algorithm tests the nodes in d-$N(x)$ toward the direction of BS such that the node with maximum unC_d_Load() is determined as new RN. The selected RN declares its identity by sending declaration message within its d-hop neighbors. Those uncovered nodes received this message will register as its member node and mark itself as covered. While there are uncovered nodes in the network, algorithm selects a new starting node again. Repeat this process, until all the nodes are covered by RNs.

```
STEP 3: OPTIMIZATION
for any s_j receiving Declar_Msg() from r_i
    if s_j.status = Covered and |s_j r_i | < |s_j r_j |) then
        changes its RN from r_j to r_i when receives r_i's message;
```

We notice that a part of nodes in d-$N(r_i)$ may have become the member of other tree T_j, $j < i$ already. In order to optimize the load of aggregation tree, if these covered nodes are closer to a new RN r_i, then they will disaffiliate themselves from original roots and join r_i. The optimization pseudo-code as shown in STEP 3 above.

An example demonstrates the execution of LP-RDA as shown in Fig. 1. The solid circles represent sensors and the black ones indicate that they are covered by RNs. The gray line segments show the connectivity, and the directed line segments represent the data routes in aggregation trees. Initially, no node is covered. Although 18 and 1 have the same 4 hops, the former with smaller

dist_to_BS is selected as the starting node. Its uncovered 2-hop neighbor set is {10, 11, 15, 19}. 11 is the desirable one which will send Declar_Msg() to recruit its members. After its neighbors at most 2-hop away joining in this RN, they are covered as shown in Fig. 1b. Similarly, node 1 with 4 hops is selected as the next starting node. In the process of construction of aggregation trees, a part of nodes change their routes whenever shorter distance to new RN arises, e.g., node 8 changes its RN from 3 to 13, with which the optimization is accomplished. Fig. 1c gives the final result which produce the data gathering tour of MC as highlighted by the red line segments.

Theorem 2. *LP-RDA has the time complexity of $O(N^2+Nd)$, where N is the number of sensors in network, d is the relay hop.*

Proof. During the initial stage, every node sends feedback message to BS hop by hop after receiving the BS's "BEACON" message. It takes $O(N)$ time for BS to gather network information. Next, BS starts the iterative RN selection process. At every turn, LP-RDA selects the farthest node v as the starting node, it will no larger than N even in the worst case. Moreover, in each turn it takes $O(d)$ time for v's d-hop neighbors d-$N(v)$ compute their load, the RN declares its identify, and the MN joins new RN, respectively. Thus the iteration requires $O(dN)$ time. Finally, adopting existing approximate algorithm for TSP to produce the MC tour will take $O(N^2)$ time. Thus, the total time spent is $O(N) + O(3dN) + O(N^2)$. The time complexity of LP-RDA is $O(N^2 + dN)$.   □

5   Distributed Algorithm for MDG-RN Problem

According to the assumption above, every node only knows the existence of its direct neighbors, thus the information acquisition of d-hop neighbors is mainly completed via d-hop information exchange. The execution of Algorithm needs BS's schedule and it can not be executed in fully distributed pattern. In the following, we present a Tree based Distributed RN Determination Algorithm (T-DRDA), which can be identified as 3 steps.

```
STEP 1: INITIALIZATION
Construct SDT T under the constraint of Rt;
si.status:= Suspensive;
```

The initialization pseudo-code executed by each senor as shown in STEP 1. Initially, each node has the same status "Suspensive". We claim the branches of a node in T are its sub-trees rooted at its direct children, and the *local height of a branch* (LHB) is the tree height from current node to its known farthest child in local message exchange. For any node x, its LHB of i-th branch is noted as $x.br\_$LHB$[i]$. Every node sends local exchange message Exg_Msg() to its father within d hops along T. When any node in the network receives messages from its children, it will perform relevant statistics and then forward or destroy the message depending on specific conditions. After d-hop message propagation, each node has the information of its d-hop neighbors.
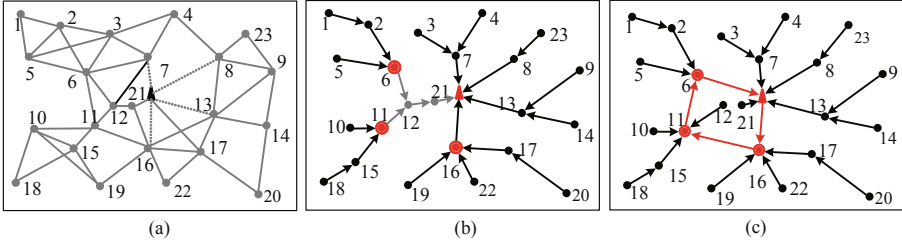
Fig. 2. An example to illustrate T-DRDA (N=23, d=2). (a) Initial topology. (b) RNs and their MNs after iteration 1. (c) The final RNs, aggregation tree, and MC tour.

```
STEP 2: STATUS DETERMINATION
if s_i.status = Suspensive then
   for each branch of s_i
      if s_i.br_LHB[j] < d-1 and s_i.parNode != Null then
         s_i.status:= MN;
      else if s_i.br_LHB[j] = d-1 or (s_i.parNode = Null and
      s_i.br_LHB[j] < d-1) then
         s_i.status:= RN and sends Declar_Msg() to d-N(s_i) in T;
```

According to the obtained local information, each node makes decision of its status as shown the code in STEP 2. If the LHB of each branch is less than d and its parent is not null, then this node becomes a member node (MN), then it will wait for a Join_Msg(). If there is a node whose LHB is exactly d, or it includes a branch whose LHB is less than d and its parent is null, then node turns into RN, and send declaration message Declar_Msg() to recruit its members within d hops range along tree including its parent and children. However, if current node's LHB is lager than d, then its status is still undetermined.

```
STEP 3: JOIN_RN
Upon receiving Declar_Msg(r_j, h) in T
if s_i.status = Suspensive then
   s_i.status:= MN and sends Join_Msg() to register as MN with r_j;
```

For STEP 3, whenever nodes receive Declar_Msg(), they change their status as "Covered", and register as MN with the sender. If multiple such messages are received, the nearest sender is chosen. Next, MN sends Join_Msg() to inform RN of its joining. After receiving the join messages, RN registers these nodes as its members and performs necessary maintenance and management. Note that when any MN determines a RN, it will be deleted from T. The remainder nodes in sub-tree repeat this procedure until every node becomes a RN or MN.

Fig. 2 illustrates the execution of distributed algorithm. The gray, red and black nodes represent MN, RN and undetermined nodes, respectively. Fig. 2a shows the initial network topology under the constraint R_t. Fig. 2b depicts the node statuses after the first iteration. Fig. 2c gives the final statuses of

all nodes and the MC tour. Finally, we give the following properties which show the complexity of the T-DRDA algorithm.

Theorem 3. *T-DRDA has the time complexity of $O(N)$, where N is the number of sensors.*

Proof. T-DRDA adopts SDT T as underlying communication topology which can be constructed in $O(1)$ time. In the worst-case, sensor will experience N/d iterations at most. In each iteration, it takes $O(d)$ time for node to obtains d-hop neighbors information by local exchange. With the gathered information, each node makes its decision independently by $O(1)$ time. After that, RN and MN will send declaration and join messages with $O(d)$ time, respectively. Therefore, The total time complexity in T-DRDA is $O(1)+O(N/d)*O(3d+1)=O(N)$. □

Theorem 4. *T-DRDA has the message exchange complexity of $O(N + d)$ per node, where N is the number of sensors, and d is the relay hop counts.*

Proof. SDT T in T-DRDA can be constructed with message complexity $O(N)$. During each iteration, each node generates d messages at most which are sent to its parent within d-hop in T. In the decision stage, each RN sends a declaration message to its d-hop neighbors in T. After receiving the declaration, its neighbors register with this node as MN by sending a join message. Both messages are restricted in d hop during the broadcast. That is, the number of messages that forwarded by single node in T will no more than d. Therefore, the total number of messages that a node has to handle is at most $d + 1 + d$. Thus the message complexity of T-DRDA is $O(N) + O(d) = O(N + d)$ per node. □

6   Performance Evaluation

In this section, we evaluate the performance of proposed algorithms and present the simulation results. The performance metrics are mainly the number of RNs (N_{RN}), the iterations, and the tour length of MC (L_{MCT}). We first evaluate the performance by varying the parameters, and then compare them with two existing mobile data gathering schemes, SPT-DGA and PB-PSA [13]. SPT-DGA is a centralized algorithm, in which within the relay hop bound it iteratively finds an set of PPs among the sensors on a shortest path tree. Whereas, PB-PSA obtains the desirable solution in a distributed manner. We adopt the Nearest-Neighbor (NN) algorithm [15] in the simulation to determine the moving tour.
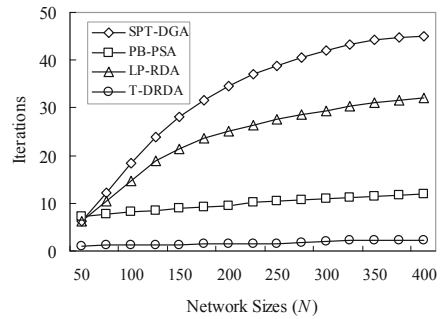
Fig. 3 shows the performance of LP-RDA and T-DRDA under different transmission ranges (R_t). We can see that N_{RN} in both algorithms decreases quickly with the increase of R_t. The reason is that under the same node density a big R_t leads to the d-hop neighbors of a sensor increasing significantly, which means that less RNs can cover all nodes in the network. Obviously, in such case the load of each RN will increase with decrease of N_{RN}. The increase of R_t makes $hopC$ reduced. Therefore, for the iterations in LP-RDA, it is consistent with $hopC$, and decreases dramatically in Fig. 3b. It is worth pointing out that small iterations
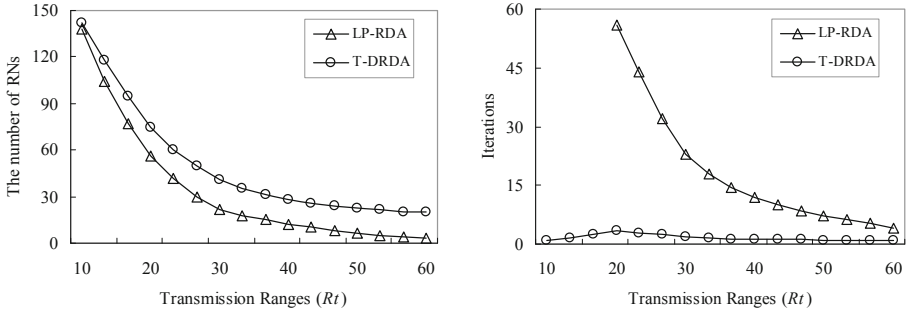
Fig. 3. Performance of LP-RDA and T-DRDA under different transmission ranges R_t. (a) The number of RNs versus R_t. (b) The iterations versus R_t.
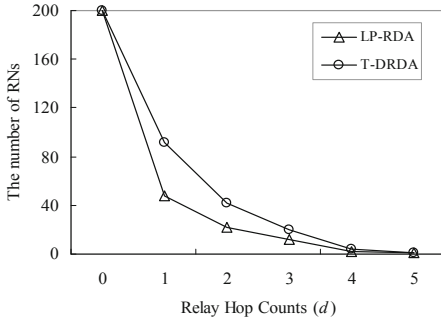
**Fig. 4.** The number of RNs   **Fig. 5.** The iterations versus network sizes

are at the cost of a large of local exchange messages. However, T-DRDA adopts a distributed approach to determine RNs, thus its iterations are influenced mainly by the height of tree T.

Fig. 4 plots the relationship between N_{RN} and d. In the figure, under fixed R_t = 30m, N_{RN} in both algorithms decreases with d. The revelation of this result is that a tradeoff should be made between local message overhead and latency. On one hand, under fixed R_t a small relay hop means the data can be aggregated to RNs quickly, but MC tour length will increase inevitably, which will cause a long latency. On the other hand, if the relay hop is too large, then the load of RNs will increase, accordingly, which not only calls for high node performance, but also result in unbalanced energy consumption.

In the following, we simulate the performance of LP-RDA and T-DRDA comparing with PB-PSA and SPT-DGA under different network sizes. Fig. 5 depicts the iterations of different algorithms as a function of network nodes N. We can see that comparing with that the iterations of centralized algorithms increasing with network sizes significantly, distributed schemes keeps a low growth and has excellent efficiency. For example, T-DRDA needs only 1 iteration when N is less

than 100. Even under the case $N = 400$, algorithm needs 3.15 rounds on average. The reason LP-RDA excels SPT-DGA is that at every turn the former selects RN from the node with maximum d-hop load within its uncovered d-hop neighbors, which produces as less N_{RN} as possible. Furthermore, during the execution, the iteration of algorithm is scheduled by BS, which has unlimited functionality.
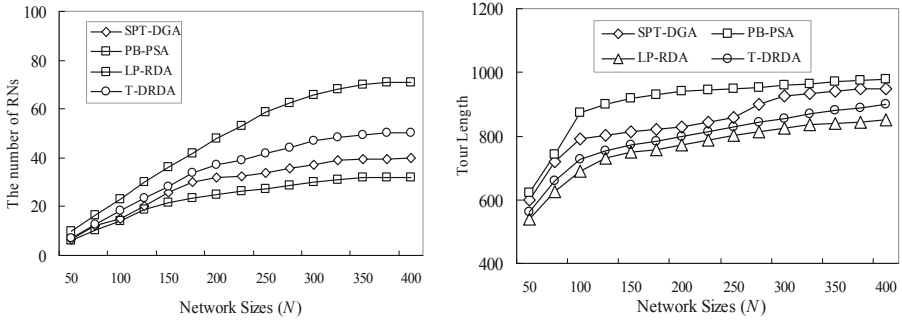
Fig. 6. Performance of LP-RDA and T-DRDA under different network sizes. (a) The number of RNs versus networks. (b) Tour length versus network sizes.

Finally, Fig. 6 depicts N_{RN} and L_{MCT} under different network sizes N. By contrast, the centralized algorithm receives more optimized N_{RN}. In order to ensure a short MC tour, the selection of RN in algorithm design mainly considers two factors: one is approaching its location to static BS, the other is decreasing their number. Under fixed d, a smaller L_{MCT} means a short latency of data gathering. We can see that the centralized algorithms are superior to distributed algorithms and the proposed algorithms outperform the other two algorithms.

7   Conclusions

In this paper, we study relay constrained mobile data gathering with mobile collector. We develop two efficient rendezvous based data gathering algorithms. One is a heuristic algorithm which always selects the node with maximum load from the d-hop neighbors of the current farthest node to BS. The other caters to the characteristic of WSNs, and selects RN iteratively from far to near in distributed manner. Both of them jointly consider MC tour and data routing routes in aggregation trees. The effectiveness of our algorithms is validated through both theoretical analysis and extensive simulations.

References

1. Xu, N., Rangwala, S., Chintalapudi, K.K., Ganesan, D., Broad, A., Govindan, R., Estrin, D.: A Wireless Sensor Network for Structural Monitoring. In: Proc. of ACM SenSys, pp. 13–24 (2004)
2. Yang, Z., Liu, Y.H.: Quality of Trilateration: Confidence based Iterative Localization. IEEE Trans. Parallel Distrib. Syst. 21(5), 631–640 (2010)
3. Samarah, S., Al-Hajri, M., Boukerche, A.: A Predictive Energy-Efficient Technique to Support Object-Tracking Sensor Networks. IEEE Trans. Veh. Technol. 60(2), 656–663 (2011)
4. Batalin, M.A., Rahimi, M., Yu, Y., Liu, D., Kansal, A., Sukhatme, G.S., Kaiser, W.J., Hansen, M., Pottie, G.J., Srivastava, M., Estrin, D.: Call and Response: Experiments in Sampling the Environment. In: Proc. of SenSys, pp. 25–38 (2004)
5. Francesco, M.D., Das, S., Anastasi, G.: Data Collection in Wireless Sensor Networks with Mobile Elements: A Survey. ACM Trans. Sensor Netw. 8(1) (2011)
6. Somasundara, A.A., Ramamoorthy, A., Srivastava, M.B.: Mobile Element Scheduling with Dynamic Deadlines. IEEE Trans. Mobile Comput. 6(4), 395–410 (2007)
7. Rao, J., Biswas, S.: Analyzing Multi-hop Routing Feasibility for Sensor Data Harvesting using Mobile Sink. J. Parallel Distrib. Comput. 72(6), 764–777 (2012)
8. Nesamony, S., Vairamuthu, M.K., Orlowska, M.E.: On Optimal Route of a Calibrating Mobile Sink in a Wireless Sensor Network. In: Proc. of INSS, pp. 61–64 (2007)
9. He, L., Pan, J.P., Xu, J.D.: A Progressive Approach to Reducing Data Collection Latency in Wireless Sensor Networks with Mobile Elements Sinks. IEEE Trans. Mobile Comput. 12(7), 1308–1320 (2013)
10. Ma, M., Yang, Y.: SenCar: An Energy-Efficient Data Gathering Mechanism for Large-Scale Multihop Sensor Networks. IEEE Trans. Parallel Distrib. Syst. 18(10), 1476–1488 (2007)
11. Gatzianas, M., Georgiadis, L.: A Distributed Algorithm for Maximum Lifetime Routing in Sensor Networks with Mobile Sink. IEEE Trans. Wireless Commun. 7(3), 984–994 (2008)
12. Xing, G.L., Li, M.M., Wang, T., Jia, W.J., Huang, J.: Efficient Rendezvous Algorithms for Mobility-Enabled Wireless Sensor Networks. IEEE Trans. Mobile Comput. 11(1), 47–60 (2012)
13. Zhao, M., Yang, Y.Y.: Bounded Relay Hop Mobile Data Gathering in Wireless Sensor Networks. IEEE Trans. Comput. 61(2), 265–277 (2012)
14. Madden, S., Franklin, M.J., Hellerstein, J.M., Hong, W.: TAG: A Tiny Aggregation Service for Ad-Hoc Sensor Networks. In: Proc. of OSDI, pp. 131–146 (2002)
15. Cormen, T.H., Leiserson, C.E., Rivest, R.L., Stein, C.: Introduction to Algorithms, 2nd edn. The MIT Press (2001)