Finding Good Mobile Sink Information Collection Paths with Quicker Search Time: A Single-Particle Multi-dimensional Search Algorithm-Based Approach

Pengyu Huang[1], Fuping Wu[2], Wei Wang[3](✉), Haiyan Liu[4], and Qin Liu[1]

[1] School of Telecommunications Engineering, State Key Labs of ISN, Xidian University, Xi'an 710071, China
{pyhuang,qinliu}@mail.xidian.edu.cn

[2] School of Physics and Optoelectronic Engineering, Xidian University, Xi'an 710071, China
fpwu@xidian.edu.cn

[3] School of Information Science and Technology, Northwest University, Xi'an 710127, China
wwang@nwu.edu.cn

[4] University of Leeds, Leeds LS2 9JT, UK
h.liu1@leeds.ac.uk

Abstract. Energy consumption is the first-class constraint for the battery-powered Internet of Things (IoT) devices and sensors. By moving around a sensor network to gather data, a mobile sink (MS) can greatly save sensor energy for multi-hop communication. To unlock the potential of mobile sinks, we need to carefully plan the path a mobile sink moves within the network for collecting information without compromising its coverage and its battery life. This paper presents a new way to find the optimal information collection path for mobile sinks. We achieve this by formulating the optimization problem as a classical Traveling Salesman Problem with Neighborhoods (TSPN). We then design a novel solver based on the particle multi-dimensional search algorithm to quickly locate a good path schedule in the TSPN optimization space. As a significant departure from prior work which uses multiple particles to explore multiple potential solutions, our method uses only one particle for problem-solving. Doing so significantly reduces the complexity of the algorithm, allowing it to scale to a larger sensor network. To ensure the quality of the chosen solution, we have carefully designed the evolutionary process for problem-solving. We show that our approach finds a solution with similar quality as those given by a multi-particle-based search, but with significantly less time. Simulation results show that our

Supported in part by the Key Research and Development Project of Shaanxi Province under Grant 2018GY-017, 2017GY-191 and 2019GY-146, in part by the Foundation of Education Department of Shaanxi Province Natural Science (15JK1742), and in part by the Foundation of Science Project of Xi'an (201805029YD7CG13(4)).

H. Song and D. Jiang (Eds.): SIMUtools 2020, LNICST 370, pp. 385–398, 2021.
https://doi.org/10.1007/978-3-030-72795-6_32

approach can find a high-quality path schedule compared to the state-of-the-art algorithm in a large sensor network.

Keywords: Internet of Things · Mobile sink · TSPN · Single particle · Shortest path

1 Introduction

With the explosive growth of the Internet of Things (IoT), it has been widely used in environmental monitoring, fire monitoring, battlefield exploration, and other application scenarios [1]. In these applications, a large number of sensor nodes are typically deployed over a wide area. Whenever a target event happens, the sensor nodes of this field could perceive it, and send the relevant data to the static sink node by multi-hop routing. We can see, these data will gradually converge to the static sink node. And as time goes on, the sensor nodes that close to the static sink node will inevitably consume more energy due to the heavy burden of data forwarding. Whenever these sensor nodes run out of energy, the static sink node also can't receive data anymore. This phenomenon is called as energy hole [2], and it could greatly decrease the lifetime of the whole network.

To address the 'energy hole' problem, researchers introduce mobile sinks (MS) into sensor networks. The mobile sink is a sink node been installed on mobile equipment, such as drones. The mobile sink collects data from sensor nodes along the tour that composed of multiple collection sites (CSs). While the mobile sink passing through a CS could collect the data from the sensor nodes close to it directly. As gathering data in this way, the sensor node no longer needs to forward data for each other. Therefore, by using the mobile sink could greatly reduce the energy-consuming of data forwarding and simplify the process of data collection. So the mobile sink is a good solution to balance the energy consumption among the sensor nodes and expand the lifetime of the whole networks [3–5], as compared to the static sink node. Moreover, the mobile sink is closer to the sensor node during communication than the static one, so it could achieve lower frame error rate and better data collection quality. Besides, the mobile sink can be used for node positioning and gain a significant improvement in accuracy [6,7].

But in practice, IoT is often been used in a wide-open area. And for the mobile sink, its ability for data collection is limited by the speed range and energy efficiency of the carrier platform. Compared with the static sink node, the mobile sink may need more time to complete the data collection of the whole network. But for now, it's still difficult to improve the capability (including speed or power supply) of the carrier platform. So, the most efficient way is to shorten the length of the data collection tour. The shorter the data collection tour, the lower the data collection latency for the whole network. So, it's an urgent issue to discover the optimal tour planning method for the mobile sink to minimize the data collection latency. In this paper, we focus on how to obtain the shortest tour of the mobile sink, so as to implement the minimum data collection latency.

During the data collection, the mobile sink travels along a pre-planned tour. Usually, the transceiver of the mobile sink is off for most of the time to save energy, and it will be available for data collection while reaching the collection sites. Base on this, we can see that how to determine the positions of the CS is the key problem. So we can divide the tour planning algorithms of mobile sink into two types according to whether use the sensor node's location as a CS:

(1) Only using the sensor node's location as the CS, the tour planning method of mobile sink could be reduced to a traveling salesman problem (TSP) problem [8–13,17–19].
(2) Without limiting the position of the CS, this tour planning problem could be reduced to a traveling salesman problem with neighborhoods (TSPN) [14,20–23].

For the TSP-based data collection methods, the mobile sink directly traverses the position of each sensor node, without considering the communication range of the sensor node. This method could simplify data collection. However, TSP is an NP-Hard problem. Especially for large-scale IoT, TSP-based method would consume a lot of computing resources, and this makes it difficult to be applied in real applications.

Moreover, for TSP-based methods, the mobile sink must visit every sensor node one by one. This means even with the fixed deployment field, the length of the data collection tour will increase with a growing number of the sensor nodes, and leading to greater data collection latency and energy consumption. As a result, the TSP-based method has many problems in scalability and practicability.

In contrast to the TSP-based method, TSPN-based methods will be better. In this kind of method, the mobile sink doesn't directly traverse the positions of sensor nodes, replaced by firstly finds out some collection sites (CSs) and only traverses them. And the CS could be any location of the deployment area. We can see, TSPN-based methods could be more flexible.

At the CS point, the MS can collect data from every sensor node within its communication range. So TSPN-based method only needs to traverse fewer CSs to complete the data collection, which means shorter data collection tour length and shorter data collection latency. So the TSPN-based method could make full use of the wireless communication capability and achieve high data collection efficiency. Furthermore, in TSPN-base methods, the mobile sink directly gathers data from every sensor node, so there are no more needs for support of network structure or protocol. So it could extremely simplify the network structure and reduce energy consumption.

To realize the TSPN-based method, the key challenge is how to choose the CSs. To address this challenge, we propose a novel multi-dimensional searching algorithm base on a single particle.

This algorithm inherits the concept of the particle from the particle swarm optimization (PSO) algorithm. The particle is composed of a series of dimensions, and each dimension represents the position of a CS. In this way, a particle could represent a complete data collection tour.

The traditional PSO algorithms usually use hundreds of particles, and each particle needs to keep tracking the historical optimal solution of its own and the population. Which requires extensive computation, then will seriously affect the effectiveness of it in practice. And it will be worse to apply PSO for resource-constrained IoT nodes.

To perform the advantages of the PSO and bypass the above problems, this paper innovatively proposes a method with only 'one' particle to solve the TSPN problem. Based on one particle, we transform the searching problem of optimal data collection tour for the mobile sink into the searching problem of a single particle with different dimensions. Besides, to further reduce the complexity of the algorithm, we employ a predefined non-uniform searching grid for each dimension. In this way, the continuous searching for each dimension is converted into a discrete searching, which could greatly reduce the complexity and computation of the algorithm.

The main contributions of this paper can be summarized as follows.

(1) In the process of using PSO to solve the best path of mobile sink, 'one' particle is used to reduce the computation.
(2) The global search for PSO is transformed into a local search for each dimension in 'one' particle.
(3) Preset grid is used in local search to further reduce the complexity of the search.

With these approaches, our algorithm could greatly reduce computation and energy consumption and gain availability in real scenarios. Simulation results show that our algorithm could dramatically reduce the computation, at the same time, it could guarantee the quality of data collection tour. So it could reduce the data collection latency significantly.

2 Related Work

As a TSP-based method, [8] studies the relationship between the moving speed of the mobile sink and the data collection efficiency in a random deployment. MASP (Maximum Amount Shortest Path) is proposed to transform the finite-time information collection problem into an integer linear programming problem. The optimal path solution will be obtained by using the genetic algorithm and a distributed approximation algorithm. [9] proposes a data acquisition protocol DCPD for the mobile sink. This protocol selects collection points according to the distribution of the sensor nodes and obtains the shortest data collection route through these collection points based on the quantum genetic algorithm. [10] summarizes the problem of data collection path length minimization as a single hop data collection problem (SHDGP), then solve it by transforming SHDGP into a mixed integer programming problem. [11] proposes an optimal selection method of the mobile sink path based on the priority of virtual points. This method could achieve the optimization objective that minimizes the energy consumption of the whole network with the restriction of time delay. It could reduce the time complexity of the algorithm in case of a small increase in energy consumption. [12,13] propose mobile sink data collection mechanisms based on the information RP (rendezvous point). In these mechanisms, the sensor node sends the collected information to the nearest RP node in a multi-hop manner, and the mobile sink traverses each PR successively to collect data. [17] proposes a routing algorithm called cluster-chain mobile agent routing (CCMAR). CCMAR introduces data compression in the process of constructing the mobile sink path. [18] proposes a path planning which is called EARTH. The main idea of this algorithm is to consider the relationship between path planning and data produce speed and buffer size of sensors. In [19], considering the relationship between the packet loss rate and mobile sink's speed, a path planning method with effective energy and the minimum packet loss rate is proposed.

In the methods based on TSPN, in [14], the shortest information collection path problem is regarded as a traveling salesman problem with the adjacent area (TSPN). This paper proposes a heuristic algorithm to construct an information collection path. [20] introduces MC-WSN architecture. Through active network deployment and topology design, the number of hops from the sensor to the moving sink is limited, thus improving the effect of path planning. [21] proposes an algorithm is called BR-CTR. In this algorithm, the multi-hop transmission is considered to reduce the delay time of data gathering. [22] considers an energy balanced tree-based data collection strategy to improve the sensing efficiency of the sensor network. To solve the problem of the coverage of the sensing area and the coverage holes problem in WSN, [23] proposed an energy-efficient solution based on the mobile sink.

3 Single Particle Multi-dimensional Particle Swarm Optimization Algorithm

3.1 Algorithm Overview

In this section, to simplify the problem, we assume the communication radius of the sensor nodes and mobile sinks are the same and the distribution of sensor nodes is already known. Furthermore, we assume the transmission time between the MS and sensor nodes is negligible as compared with the traveling time of the MS [15]. With these assumptions, the data collection mission can be accomplished as the traveling tour been finished.

For now, the key to achieving the MS's optimal data collection tour depends on the answer of two challenges: how to choose the optimal CSs and how to connect these CSs to form a shortest data collection tour. So the focus of this paper can be summarized as follows: under the condition of given sensor node distribution, select the appropriate CSs, and then connect these CSs to form a shortest data collection tour. Mobile sink visits all CSs along this tour to collect data and finally achieves the minimal data collection latency. The tour construction problem, in this case, can be formulated as:

$$min\,|T|\; s.t.\forall s_i \in S, \exists st_j \in ST, |s_i, st_j| \leq d \quad T \in \Gamma \tag{1}$$

- $|T|$ is the length of the tour;
- S is the set of sensor nodes;
- ST is the set of collection sites;
- Γ is the set of possible tours;
- $|s_i, st_j|$ is the Euclidean distance from s_i to st_i;
- d is the effective communication distance of the sensors;

According to the above discussion, the first step is to select some CSs that could cover all the sensor nodes within their communication range. Then the second step is to figure out the optimal TSP tour involves all the CSs. Since there are many mature methods to calculate the optimal TSP tour, so how to realize the first step is the key challenge for us.

According to relevant studies, we know different sets of CSs will directly cause the different length of MS's tour and the size of data collection latency. Therefore, the key to obtaining the optimal data collection tour is to find the best number and the positions of CSs. However, the distribution of sensor nodes is random in the real application. Moreover, even if the distribution of sensor nodes is fixed, there are infinite possibilities for the sets of CSs.

For the data collection tour is a multi-dimensional structure composed of CSs, so finding the best set of CSs is a typical multi-dimensional searching problem. The most difficult part of this problem is how to find the best one from the infinite combinations of CSs. We find that the PSO algorithm is one of the most effective methods to solve this kind of problem. In the traditional PSO algorithm, a particle is used to represent a feasible tour of the mobile sink. Each particle involves several dimensions, and each dimension represents the coordinates of a CS.

As shown in formula 2 and 3, in the traditional PSO-based information collection tour solution, x is an ordered vector consisting of the positions of CHs to represent a feasible solution. According to formula 2 and 3, x is iterated continuously to obtain the optimal solution.

$$\nu_{id}^{k+1} = \omega \times \nu_{id}^{k} + c_1 \times \gamma_1 \times (pbest_{id}^{k} - x_{id}^{k}) + c_2 \times \gamma_2 \times (gbest_{id}^{k} - x_{id}^{k}) \quad (2)$$

$$x_{id}^{k+1} = x_{id}^{k} + \nu_{id}^{k} \quad (3)$$

- x_{id}^{k}: The d-th dimension (the location of the d-th CH) of particle i at time k;
- ν_{id}^{k}: The speed of d-th dimension of particle i at time k;
- ω: Inertia weight;
- c_1: The step-length for the particle which follows to the best solution of itself;
- c_2: The step-length that the particle which follows to the best solution of the group;
- γ_1 and γ_2: Random values between [0,1];

Therefore, a particle can be considered as a multi-dimensional vector. The data collection can be achieved by gradually moving the mobile sink along the dimensions (CSs) of the particle.

However, the traditional PSO algorithm usually involves hundreds of particles, and each particle needs to track the historical optimal solution of its own and the entire swarm. These methods will consume more memory space and more energy. But in the realistic application, MS's computing capacity and power supply are very limited, so it is difficult for them to implement the tour searching by PSO. To address this problem, we improve the traditional PSO algorithm from 4 aspects.

(1) We don't use a large number of particles but only one particle, and it becomes unavailable to use the tracking method of the traditional PSO algorithm. This means the particle does not need to record the historical optimal solution of the other particle anymore. In this way, it can greatly reduce the complexity of the algorithm, and the requirement of computing resource.
(2) Using a single particle, the main challenge is how to realize multi-dimensional searching without traditional evolution proceeding. Here we propose a novel multi-dimensional searching method that evolves from random searching. In this method, we repeatedly select a dimension randomly, then search a better position for it, iterating until finding the optimal data collection tour.
(3) While searching in each dimension, to further reduce the complexity, we introduce a pre-built searching grid template. After selecting a dimension randomly from the particle, we use the searching grid template to determine the alternative dimension set. Since a dimension represents the position of a CS, it is necessary to ensure that the mobile sink can still cover the same or more sensor nodes when moving from the current position to the alternative position. So, we must decide the alternative dimension positions one by one and select these valid positions to form the final alternative dimension set. After that, calculate the length of all TSP tour when the current dimension is replaced by the dimension in the final alternative dimension set. Then select the alternative dimension with the shortest tour length to update the current dimension. Iterate the above process to get the best tour.
(4) As the algorithm runs, some of the dimensions in the particle may get closer. Because this phenomenon could reduce the quality of the data collection tour, we design a dimension merge method to merge the dimensions that too closed. Through this method, we can optimize the data collection tour.

3.2 Algorithm Flow

Without loss of generality, the communication range of mobile sink and sensor nodes is set as R. That is, while in the circle with the mobile sink as the center and the R as the radius, the sensor nodes can transfer data to the mobile sink. We assume the position of the sensor nodes is known.

At first, by using the traditional TSP algorithm, we get a TSP tour and use it as the initial data collection tour. The next step is to randomly select a dimension (CS) from the initial data collection tour. Then performing the third step that the local searching for the selected CS. And this is right the key of the whole algorithm. In order to simplify the searching process and reduce complexity, we design a pre-build search grid template. Its principle is shown in Fig. 1.

First of all, we assume the randomly selected CS is RP_i and $\widehat{RP}_i$ is the new position that RP_i will move to. The black dots in Fig. 1 represent the sensors in the WSN. From Fig. 1, we can see that the greater the distance between the RP_i and $\widehat{RP}_i$, the less overlap their coverage. Furthermore, the smaller the overlap area, the fewer sensor nodes in the overlap area.

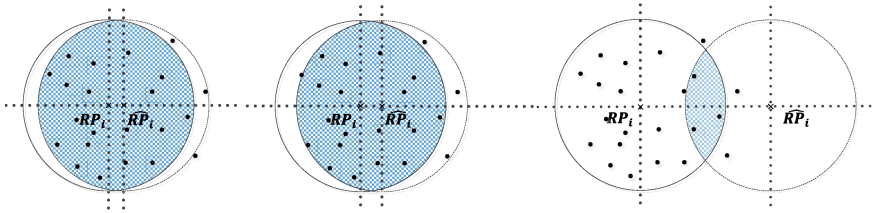

Fig. 1. The relationship between the distance of CSs and overlapping areas.

MS only collects data from sensor nodes at CSs. So we can see, if the MS moves from RP_i to $\widehat{RP}_i$, as shown in Fig. 1(a), the probability of it to cover the sensor nodes becomes greater. In the case of Fig. 1(c), the probability is small. Therefore, we can move the position of CS bit by bit while maintaining the same coverage, and at the same time optimize the data collection tour gradually.

The next challenge is how to find the best $\widehat{RP}_i$. Here we propose a searching grid template to solve this problem. Since we take the CS's coverage as a circle, we set the area covered by RP_i as S1 and the overlap area of RP_i and $\widehat{RP}_i$ as S2. According to the percentage S2 over S1, a searching grid template with 4 layers is constructed, and the searching circles from inside to outside represent the percentage 95%, 90%, 80%, and 70%. The radius of the searching circle can be calculated from the percentage and R. The same number of grid points is set on every searching circle. Finally, we construct a searching grid template with 4 layers and 32 grid points, as shown in Fig. 2.

As shown in Fig. 2, we select the points in 8 directions of each layer as grid points. For convenience, we mark the grid points as $gp_i^k, i \in \{0.95, 0.9, 0.8, 0, 7\}, k \in \{1, 2, ..., 8\}$. Where subscript i represents the percentage of overlap, superscript k represents the index of the grid point and the value from 1 to 8.

So the algorithm flow can be described as follows:

Step1: Generate a TSP tour based on all sensor nodes as the initial tour.

At this time, the position of every sensor node serves as the CS. The tour is mapped to a single particle. Generate the searching grid template as shown in Fig. 2.

Step2: Multi-dimensional searching for single particle.

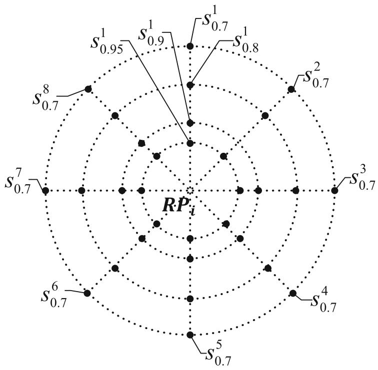

Fig. 2. The search grid template of RP_i.

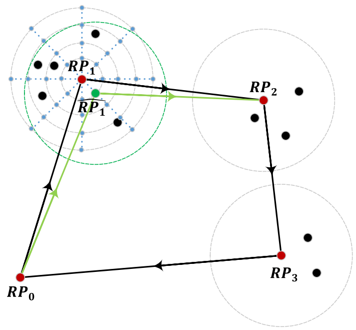

Fig. 3. The flow of the single particle multi-dimensional searching algorithm.

Select a random dimension (CS) in the particle;

The searching grid template is superimposed on the position of the CS selected in the previous step to obtain the alternative position set according to the searching grid template.

Search for each position in the alternative position set. Maintaining coverage, determine which position will reduce the tour length most. Select this position to update the selected CS.

Step3: Merge CSs.

Calculate the Euclidean distance between adjacent CSs. If the distance is less than 0.1R, arrange the sensor node sets covered by these two CSs respectively. If there is one CS that could cover all the sensor nodes set by the others, then delete the other CSs from the tour.

Step4: Calculate the fitness value.

Fitness value is used to define the effectiveness of the tour. In this algorithm, we select the tour length as the fitness value of the particle. The smaller the fitness value, the better the tour.

Step5: Exit condition judgment.

In order to improve the computing speed of the algorithm, we set the algorithm exit condition as followed: if the rate of change of fitness value is less than 0.1% in 100 iterations, the algorithm exits. If the algorithm doesn't terminate, return to Step2.

Step6: Output the optimal data collection tour and the fitness value.

Figure 3 is an example of our algorithm.

4 Analysis of Simulation Results

We evaluate the performance of our algorithm and compare it with the traditional TSP and the COM algorithm [16]. For convenience, we refer to our algorithm as APMDSA.

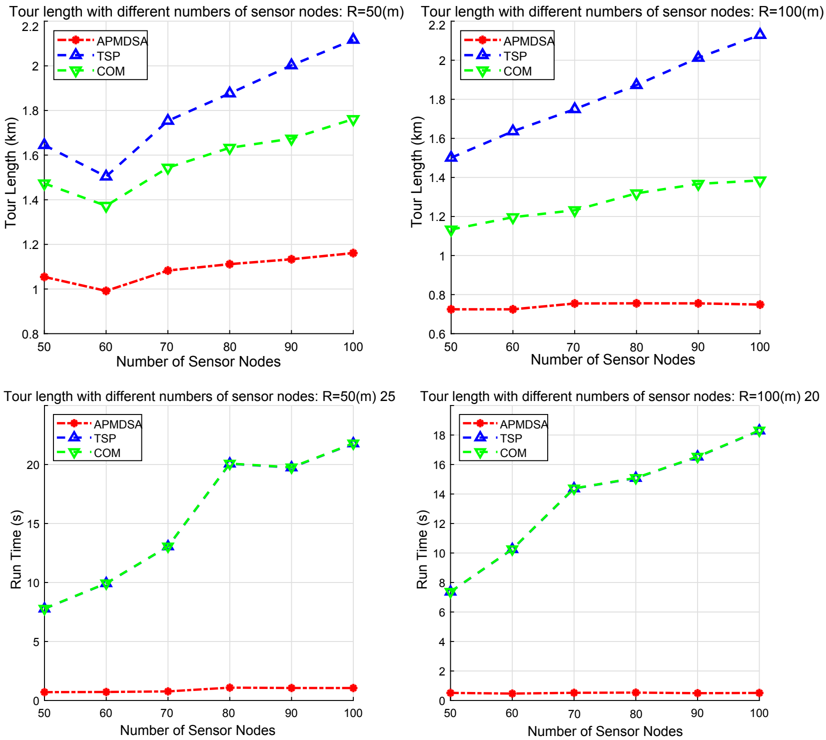

Fig. 4. Tour length and run time with different communication radiuses.

In the simulation, we use the same settings as in [16]. We consider a sparse square sensing field with size 500 m 500 m, where 50 to 100 nodes are uniformly deployed at random. The sensor node and the mobile sink have the same communication range, which is set from 20 m to 100 m. We generate 100 topologies for each case, and each topology is simulated 50 times. Our simulation is run on a 3.00 GHz CPU and 8G memory.

At first, the communication radius is 50 m and 100 m, and the number of sensor nodes increases from 50 to 100 step by step. Simulation is performed for these three algorithms, and the results are shown in Fig. 4. Then we fix the number of sensor nodes as 50 and 100, and the communication radius of sensor nodes gradually increases from 20 m to 100 m. The three algorithms were simulated again, and the results are shown in Fig. 5.

In Fig. 4, fixing the communication radius, the optimal tour length obtained by these three algorithms gradually increases no matter the number of sensor nodes is 50 or 100. Because as the number of sensor nodes increases, the number of CSs that the mobile sink needs to traverse also increases simultaneously, and this will increase the tour length. With a different number of sensor nodes, we can see that the performance of TSP is the worst. Because in the TSP, the position of each sensor node is the CS, which means the mobile sink needs to traverse every sensor node's position to perform the data collection. So the tour length obtained by TSP is the longest.

Compared with the TSP, the performance of the COM has been greatly improved. Since the COM merges adjacent CSs according to the overlap degree of adjacent CSs' coverage. The tour generated by the COM comprises fewer CSs, so the tour length is relatively better. Our APMDSA algorithm has the best performance. In the case that the communication radius is 50 m and the node's number is 50, the tour length generated by APMDSA is 30% shorter than the COM algorithm. If the number of sensor nodes increases to 100, the advantage of APMDSA expands to 33%. As evaluate time consumption, while the number of nodes is 50 and the communication radius is 50m, the running time of APMDSA is 12.5% of COM. While the number of sensor nodes increases to 100, the running time of APMDSA is only 4.6% of COM.

The number of sensor nodes is fixed as 50 and 100, while the communication radius increased from 20 m to 100 m, the simulation results are also similar and the TSP is still the worst. With the communication radius increase, since both COM and APMDSA take into account the communication range, their performance will be improved accordingly. Moreover, with a larger communication radius, the mobile sink can collect data from the sensor nodes within a larger range. Therefore, with the increase of communication radius, the performance improvement of COM and APMDSA expands, and APMDSA improves more. As can be seen from Fig. 5, when the number of sensor nodes is 50 and the communication radius is 100 m, the tour length of APMDSA is reduced by 35% as compare to COM. If the number of sensor nodes increases to 100, the advantage of APMDSA increases to 45%. In terms of time consumption, the advantages of our algorithm are more obvious. While the number of sensor nodes is 50 and the communication radius is 100 m, the operation time of APMDSA is 7% of COM. While the number of sensor nodes increases to 100, the running time of APMDSA only is 1% of COM.

These results show that APMDSA has achieved excellent performance under different simulation settings. APMDSA can effectively reduce the length of the data collection tour of the mobile sink, thus reducing the data collection latency of the whole network.

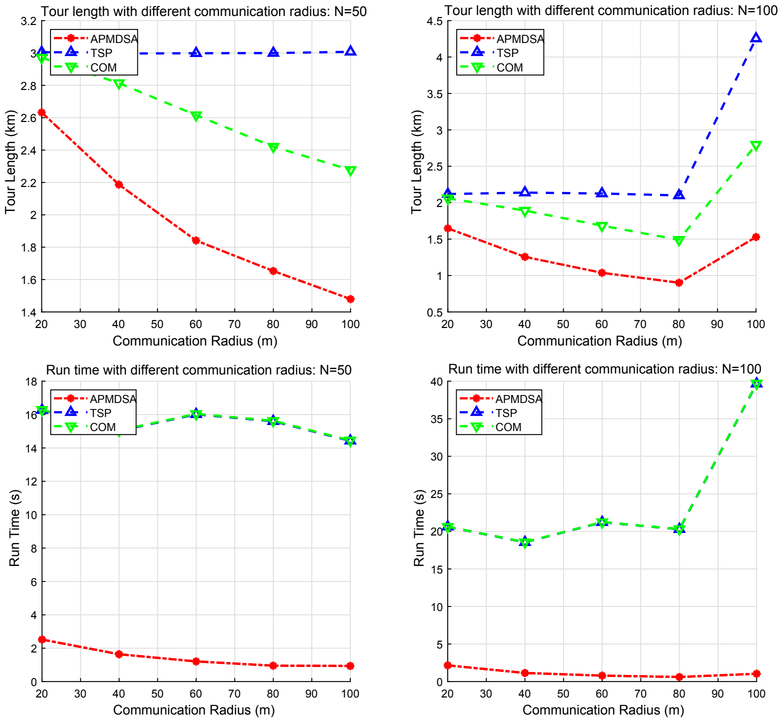

Fig. 5. Tour length and run time with different numbers of sensor nodes.

5 Conclusions

Data collection is very important for IoT applications. The use of the mobile element makes it possible to collect data within the communication range during the movement of the mobile sink. At the same time, this method can reduce the complexity of the network protocol. In this paper, we propose a novel algorithm to finding the optimal data collection tour for the mobile sink in IoT applications. We fully consider the wireless communication ability of the sensor node. Our algorithm selects several CSs and obtains the shortest tour by the multi-dimensional searching method within single-particle. This algorithm greatly reduces the complexity and computation of the searching method. The simulation results show that as compared with the TSP and COM, it can obtain a better data collection tour and further extend network lifetime. As compared to the other TSPN-based algorithms, the proposed algorithm has a simpler structure and better expansibility. Therefore, our algorithm is very suitable for large-scale networks.

References

1. Ou, C.H.: A localization scheme for wireless sensor networks using mobile anchors with directional antennas. IEEE Sens. J. **11**(7), 1607–1616 (2011)
2. Olariu, S., Stojmenovi, I.: Design guidelines for maximizing life time and avoiding energy holes in sensor networks with uniform distribution and uniform reporting. In: IEEE INFOCOM, pp. 1–12. Society Press (2006)
3. Zhao, M., Yang, Y.: Bounded relay hop mobile data gathering in wireless sensor networks. IEEE Trans. Comput. **61**(2), 265–277 (2012)
4. Lin, K., Chen, M., Zeadally, S., et al.: Balancing energy consumption with mobile agents in wireless sensor networks. Future Gener. Comput. Syst. **28**(2), 446–456 (2012)
5. Guo, S., Wang, C., Yang, Y.: Joint mobile data gathering and energy provisioning in wireless rrechargeable sensor networks. IEEE Trans. Mob. Comput. **13**(12), 2836–2852 (2014)
6. Liao, W.H., Lee, Y.C., Kedia, S.P.: Mobile anchor positioning for wireless sensor networks. IET Commun. **5**(7), 914–921 (2011)
7. Chen, H., Shi, Q., Tan, R., et al.: Mobile element assisted cooperative localization for wireless sensor networks with obstacles. IEEE Trans. Wirel. Commun. **9**(3), 956–963 (2010)
8. Gao, S., Zhang, H., Das, S.K.: Efficient data collection in wireless sensor networks with path-constrained mobile sinks. IEEE Trans. Mob. Comput. **10**(4), 592–608 (2010)
9. Jian, G., Lijuan, S., Ruchuan, W., Xiao, F.: Path planning of mobile sink for wireless multimedia sensor networks. J. Comput. Res. Dev. **47**(z2), 184–188 (2010)
10. Ma, M., Yang, Y., Zhao, M.: Tour planning for mobile data-gathering mechanisms in wireless sensor networks. IEEE Trans. Veh. Technol. **62**(4), 1472–1483 (2013)
11. Shuai, G., Hongke, Z.: Optimal path selection for mobile sink in delay-guaranteed. ACTA Electronica Sinica **39**(4), 742–747 (2011)
12. Xing, G., Wang, T., Jia, W., et al.: Rendezvous design algorithms for wireless sensor networks with a mobile base station. In: ACM International Symposium on Mobile Ad Hoc Networking and Computing 2008, pp. 231–240. ACM (2008)
13. Xing, G., Wang, T., Xie, Z., et al.: Rendezvous planning in wireless sensor networks with mobile elements. IEEE Trans. Mob. Comput. **7**(12), 1430–1443 (2008)
14. Yuan, Y., Yuxing, P., Shanshan, L.: Efficient heuristic algorithm for the mobile sink routing problem. J. Commun. **32**(10), 107–117 (2011)
15. Chipara, O., et al.: Real-time power-aware routing in sensor networks. In: 14th IEEE International Workshop on Quality of Service 2006, pp. 83–92. IEEE (2006)
16. He, L., Pan, J., Xu, J.: A progressive approach to reducing data collection latency in wireless sensor networks with mobile elements. IEEE Trans. Mob. Comput. **12**(7), 1308–1320 (2013)
17. Sasirekha, S., Swamynathan, S.: Cluster-chain mobile agent routing algorithm for efficient data aggregation in wireless sensor network. J. Commun. Netw. **19**(4), 392–401 (2017)
18. Wang, Y.C., Chen, K.C.: Efficient path planning for a mobile sink to reliably gather data from sensors with diverse sensing rates and limited buffers. IEEE Trans. Mob. Comput. **18**(7), 1527–1540 (2018)
19. Alami, H.E., Najid, A.: MS-routing-G i: routing technique to minimize energy consumption and packet loss in WSNs with mobile sink. IET Netw. **7**(6), 422–428 (2018)

20. Abdelhakim, M., Liang, Y., Li, T.: Mobile access coordinated wireless sensor networks - design and analysis. IEEE Tran. Signal Inf. Process. Over Netw. **3**(1), 172–186 (2016)
21. Chien-Fu, C., Chao-Fu, Y.: Mobile data gathering with bounded relay in wireless sensor networks. IEEE Internet Things J. **5**(5), 3891–3907 (2018)
22. Sha, C., Song, D., Yang, R., et al.: A type of energy-balanced tree based data collection strategy for sensor network with mobile sink. IEEE Access **7**, 85226–85240 (2019)
23. Gharaei, N., Malebary, S.J., Bakar, K.A., et al.: Energy-efficient mobile-sink sojourn location optimization scheme for consumer home networks. IEEE Access **7**, 112079–112086 (2019)