U.P.B. Sci. Bull., Series C, Vol. 81, Iss. 3, 2019 ISSN 2286-3540

A MOBILE AND LOW-ENERGY-CONSUMPTION MECHANISM FOR ENERGY REPLENISHMENT IN RECHARGEABLE WIRELESS SENSOR NETWORKS

Chan JIANG[1], Junbin LIANG[2], Junchen LIU[3]

Rechargeable wireless sensor networks (RWSN) are a kind of new sensor networks appear in recent years. In order to extend working periods of the networks, a mobile wireless charging vehicle (MWCV) is often used to ramble in the networks to charge the nodes with low energy. However, it is a big challenge for MWCV to maximize the total energy replenishment within given time interval. In this paper, a novel mechanism named MerLEC (Maximize Energy Replenishment with Low Energy Consumption) is proposed. Firstly, each node exchanges energy information with its neighbors and the base station constructs a tree to collect energy information of all nodes in the networks. The energy information will be used to compute a length-limited path for the charging vehicle to visit nodes with low energy. Then, each node is assigned a weight, and the nodes with higher weights can attract the charging vehicle to visit them through a shorter path and they should be charged more energy. Next, the nodes whose energy levels are lower than a cordon will be chosen as initial charging targets, which ensures energy of these nodes could be maintained for a long time. Finally, given a path with limited length, MWCV keeps choosing nodes with high weights as charging targets. By this way, the total energy replenishment in each round of recharging is improved, and the charging frequency of MWCV is reduced. Theoretical analyses and simulation results show that, comparing with existing works, MerLEC can ensure persistent operation of the network meanwhile improve the total energy replenishment in each round, and achieve lower energy consumption in data collection.

Keywords: Rechargeable wireless sensor networks, Energy replenishment, Path optimization, Low-energy-consumption, Data collection

1. Introduction

A wireless sensor network (WSN) is a multi-hop network that has large numbers of sensor nodes with limited wireless communication ability and data

[1] Guangxi Key Laboratory of Multimedia Communications and Network Technology, School of Computer and Electronics Information, Guangxi University, China, e-mail: jiangchan@gxu.edu.cn

[2] Guangxi Key Laboratory of Multimedia Communications and Network Technology, School of Computer and Electronics Information, Guangxi University, China, e-mail: liangjb@gxu.edu.cn. (Corresponding author)

[3] Guangxi Key Laboratory of Multimedia Communications and Network Technology, School of Computer and Electronics Information, Guangxi University, China, e-mail: junchenl@gxu.edu.cn

processing capability. The network is widely used in real-time environment monitoring. The data collected by WSN can be sent to users for further processing [1]. WSN attracts widespread attentions in the fields of industry, agriculture, medicine and national defense [2-5].

In applications, sensor nodes are generally deployed in harsh environments, such as desert and swamp. Replacing batteries of the nodes is difficult and unrealistic. Therefore, how to preserve the energy of the nodes becomes an important problem in wireless sensor networks. Current solutions for this problem can be summarized as the following aspects: optimizing layout of the network [6]; balancing energy consumption of the nodes; designing efficient network layer protocols or data link layer protocols for data collection [7]; controlling sampling frequencies of the nodes [8] and reducing redundant data in each transmission. However, no matter which solutions uses, the extension of network lifetime is limited because the energy of the nodes is limited.

In recent years, with the development of charging technology, rechargeable wireless sensor networks (RWSN) attract more and more attentions. In RWSN, a node’s energy can be replenished by a MWCV [9-10]. If all nodes can be charged in time, the networks can work permanently.

RWSN has many practical applications. For example, Guo et al. [11] proposed a framework of joint wireless energy replenishment and anchor-point based mobile data gathering mechanism in RWSN by considering various sources of energy consumption and time-varying nature of energy replenishment, which can be used in bird habitat monitoring on isolated islands. Shu et al. [12] studied the problem of replenishing batteries of all nodes in the network by a mobile charger that periodically travels along a certain trajectory in the network deployment area, which can be used in volcano monitoring. Lin et al. [13] designed an on-demand charging architecture for large-scale RWSN, which is suitable to be used for animal tracking in primeval forest.

This paper proposed a novel algorithm named MerLEC (Maximize Energy Replenishment with Low Energy Consumption), allowing the charging vehicle replenish energy to more nodes in a limited delay, which significantly enhance the total supply energy of charging vehicle and reduce the charging frequency of charging vehicle.

2. System Model and Problem Statement

2.1. Network Model

A schematic diagram of RWSN’s network model is shown in Fig. 1. There are n static rechargeable sensor nodes and a mobile wireless charging vehicle (MWCV) in the network. All nodes are randomly deployed in a square area, the size of the area is $A=M\times M$ field. The center of the area is selected as a deployment point of base station (BS). MWCV will collect sensing data of nodes and replenish energy to nodes. The entire sensor network will consist of an undirected graph denoted as $G(V, E)$, this graph may contain one or more connected sub-graphs. Place a management node in each connected sub-graph to manage information of internal nodes within its own connected sub-graphs, the information includes location information and energy information of nodes, and this manage mode will be responsible for communicating with the base station node. Wherein, V is the set of rechargeable nodes in G, that are labeled as $V=v_1,v_2,\ldots, v_{n-1}, v_n$, respectively, and $|V|=n$. E is the set of edges in G. If two nodes v_i and v_j can communicate with each other, then $(v_i, v_j)\in E$, the number of edge is denoted as $|E|=m$. Data size of sensing node is fixed and cannot be aggregated with the received data. Nodes and base station cannot move after deployment; Different nodes may have different initial energy; the communication radius of all the nodes in the network is r, where $r<<M$; the management node is assumed to have infinite power and will not collect data in the network; there is only one management node in each connected sub-graph; MWCV is assumed to have infinite power supply and can move freely in the network. Furthermore, we assume that at the same time, mobile wireless charging vehicle can only charge one node.

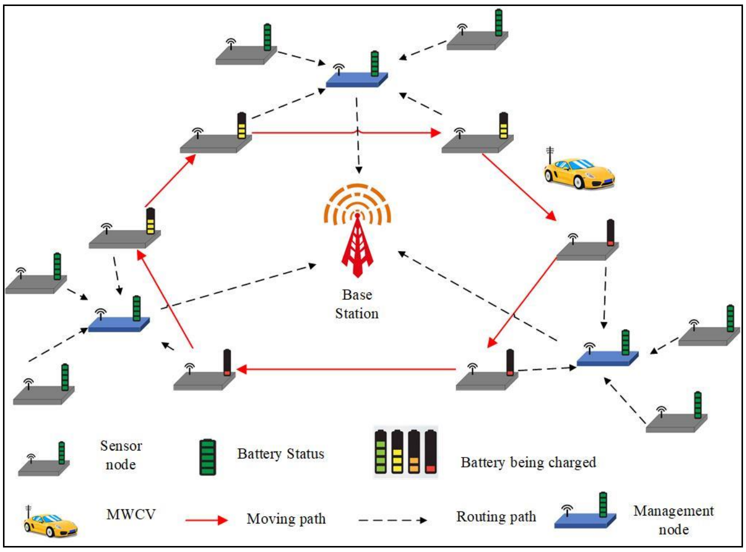


Fig.1 Network model of RWSN

2.2. Definitions

Definition 1: Base station is a special node in the network, which has infinite energy supply and powerful computation ability.

Definition 2: Target node in the network is determined as a charging target node.

Definition 3: Remaining node is node in the network other than the target node.

Definition 4: Management node is responsible for managing its own internal node information within its connected component, and takes charge for communicating with the base station node

Definition 5: A round is defined as the process of mobile wireless charging vehicle traverses all the location of target nodes and recharges energy to all target nodes.

Definition 6: Energy threshold. When the energy of node is below a threshold, the node will be chosen as the target node, the threshold is called energy threshold.

Definition 7: Replenishable energy is defined as the difference value of the maximum power and current power of node.

Definition 8: Reference path is defined as the TSP shorted path that constitute by unselected nodes and target nodes (including base station) selected by the network.

Definition 9: The weight value of node is defined as the ratio of replenishable energy to reference path.

2.3. Problem Statement

Our problem is how to select reasonable charging targets in the circumstance of defined path length, so that persistent operation of the network can be achieved. Moreover, the total energy replenishment in each round of data collection is maximized.

3. Design of MerLEC

3.1. Basic Idea of MerLEC

The paper takes account of the idea of integrated charging mode, considering the location information and energy information of nodes, and then proposes a novel charging strategy named MerLEC. Firstly, each node exchanges energy information with its neighbors, and the base station constructs a tree to collect energy information of all nodes in the networks. The energy information will be used to compute a length-limited path for the charging vehicle to visit nodes with low energy. Then, MerLEC selects the target nodes that the mobile charging vehicles must access within each connected component to ensure that the entire network data can be collected. Next, according to the replenish energy of the remaining nodes and the TSP path length of the selected target nodes, and then a weight value will be assigned to each of the remaining node. The higher of weight value, a short path can be formed to supply more energy when select this node. In a defined length of the path, the charging vehicle will constantly choose the node with high weight value as a target node, increasing the total amount of supply energy in each round.

3.1.1. Collecting energy information

On one hand, in order to make all nodes know their neighbors' energy information, the nodes should exchange energy information with their neighbors. Each node maintains a table to record the information. Based on the energy information of neighbors, the nodes can help the charging vehicle to find its charging target as soon as possible.

On the other hand, the charging vehicle needs to compute a shortest path to visit as many as possible low energy nodes. Therefore, the base station should collect the topology information of the network and energy information of the nodes in the network. A tree-based information collection method is used.

Firstly, the base station broadcasts a message contains its level 0. Any node v_i that receives the message will take the base station as its parent and set its level as 1, and then broadcasts a new message contains its level l_i to its neighbors. An arbitrary node v_j that does not decide its level and receives a message contains v_i, it will take v_i as its parent and set its level l_j as v_i +1, and then broadcasts a new message contains its level l_j to its neighbors. The above process continues until all nodes set their levels. After the process terminates, a tree rooted at the base station is constructed. Based on the tree, each node will transmit information of its neighbors and its energy to the base station.

3.1.2. Preliminary Selection of Target Nodes

In practical application, nodes are often randomly deployed, there may be many connected components in the network. To ensure MWCV can collect the entire sensing data of nodes in the network, we must ensure that there is at least one node that it will be selected as the target node within each connected component. First, check the energy of all the nodes within each connected component. To prevent nodes dying due to untimely charge, when the energy of a node is lower than the threshold, the node will be selected as the target node. For the connected components that their all nodes' energy is higher than the energy threshold, we will calculate the weights value of all the nodes within the connected component. The base station will be chosen as a selected target node, the weight value of node *i* can be calculated as follows:

$$W_i = (FullEnergy - E_i)/TSPNN(Target) \quad (1)$$

where, FullEnergy is the highest chargeable energy for node *i*, E_i is the current energy of node *i*, Target is the target set for the target nodes (initially only node *i* and the base station BS are in this target set), TSPNN is the nearest neighbor algorithm [14], which is used to compute the reference path length L_i of node *i.* The greater of the weight value, MWCV can use a shorter path to supply more energy. After recording the weight value of W_i, the node *i* will be removed from the target set. When the energy of all nodes within a connected component is higher than the energy threshold, the node with the maximum weight value will be chosen as the target node. So far, there is at least one target node in each connected component, which can ensure that the charging vehicle can access all the connected components, the whole data can be collected in the network. In addition, after selecting a target node in a connected component with a higher density of nodes, when the energy level of the remaining nodes is relatively high, possibly causing there is only one final target node in this connected component. When the target node starts to collect data of other nodes, it will consume a lot of energy, and may cause early death. To avoid this happening, we continue to select the node with the maximum weight value as the target node within each connected component, so that make the number of target nodes in each connected component is at least $\lfloor N_i/r \rfloor$, where N_i represents the number of nodes in the connected component, the coefficient r is the length of node's communication radius. More intensive of nodes in the connected component, more nodes will be continuously selected as the target.

3.1.3. Determination of Final Target Node

When initially select a target node, there is at least one node within the Target set in each connected component. At this time, with the defined-length path, we need to select a target node among the remaining nodes to join the Target set. When the final Target set is determined, the TSP loop length constituted by all the nodes in the set is just less than the defined-length path L_{TSP}. The main steps to determine the final Target set is:

(1) Calculate the reference path formed by each remaining node and nodes in Target set.

(2) Calculate the weight value of remaining nodes whose reference path length is less than the defined path L_{TSP}. Select the node with the maximum weight value to add to the target set, and then perform the first step again.

(3) Algorithm will stop when there is no reference path that is less than the defined path L_{TSP} among remaining nodes.

Thus, the final Target set is determined.

3.2. Distributed implementation of MerLEC Algorithm

When the size of rechargeable sensor network is large, there are also a large number of nodes in the network. Since nodes may be external damaged by environment, such as earthquakes, floods and other uncontrollable factors, which often make the nodes die easily, thus, it is hard to manage all nodes for the base station. At this time, the connected components can be divided into many regions; each management node only manages the internal nodes within its connected components. Based on this application scenario, this paper proposed a distributed implementation of MerLEC. Management node in each connected component will select a node with the maximum weight value and submit the information of this node to the base station. For a defined-length path, the base station will select the node with the maximum weight value among the submitted nodes to add to the set of target nodes and notify the management node in the selected connected component. Repeating the above steps, all management nodes distributed continuously submit nodes to base station for its selection, until the TSP loop length constituted by target nodes is just less than the defined-length path L_{tsp}.

3.2.1. Preliminary Selection of Target Node

Similar to many centralized algorithm, in order to ensure charging vehicle can access to each connected component, first of all, we select at least one node to add to the set of target nodes within connected components. Each management node will submit the nodes that their energy is lower than the energy threshold within its own connected components to the base station and add the nodes to the set of target nodes to prevent the death of nodes by untimely charging result. If the energy of all nodes within connected components is higher than the energy threshold, the weight value of each node will be calculated (only the base station in the set of target nodes). The node with the maximum weight value will be submit to the base station, and also add to the set of target nodes. According to centralized algorithm and the intensity of nodes within the connected components, some nodes with higher weight value will be alternatively selected to join the set of the set of target nodes.

3.2.2. Determination of the Final Target Node

After the determination of initial set of target nodes, each management node will submit the node with the maximum weight value within its connected components to the base station, and then the nodes will be further selected by the base station. The main steps of determining the final target node is:

(1) All the connected components will be labeled by 1,2, ..., g, and matrix I with 1 row and g columns will be established.

(2) Each management node calculates the weight value of all nodes within its own connected component, the node with the maximum weight value will be submitted to the base station. Matrix $I(1,1)$, $I(1,2)$, ..., $I(1, g)$ will record the id number of the submitted nodes.

(3) TSP path length constituted after each node joining the set of target nodes will be determined by the base station, without exceeding the defined-length path of L_{tsp}, choose a node with the maximum weight value to join the set of target nodes. If the selected node is from the connected component i, then $I(1, i)$ is set to 0.

Where, algorithm continuously determines whether $I(1,1)$, $I(1,2)$, ..., $I(1 , g)$is 0. When $I(1, i)$ is 0, the base station will notify the management node of the connected component i. At this time, the management node of connected component i will resubmit a node with the maximum weight value to base station among its internal remaining nodes. The network will work distributed throughout the network, management nodes of each connected components will not interfere each other; they only manage their own internal nodes within their connected components.

4. Performance Analysis of MerLEC

4.1. Hardness of MerLEC Algorithm

Theorem 1. The MerLEC problem is a NP-hard Problem.

Proof: Since the TSP problem is a typical NP-hard problem, thus, a polynomial-time reduction will be used to equivalently transform TSP problem to a special case of MerLEC problem, and then prove MerLEC problem is a NP-hard problem. First, given a TSP instance undirected graph G (V, E), and construct a graph $G'(V', E')$ that has the some topology as G, G' (V',E') is an instance of MerLEC, where, V' represents all nodes in the network, E' represents the edge between any two nodes. We can reduce the communication radius of node to a small value, which makes any two nodes in the network cannot communicate with each other. Reducing the communication radius of nodes such that each node cannot communicate with each other is a very intuitive operation and can be completed in a polynomial time. Under the circumstances that all nodes in the network cannot communicate with each other, in order to collect the entire sensing data of nodes in network, MWCV need to access all nodes to collect data in the network, this means that all nodes are the root nodes. At this time, path length for data collection in graph G' is equal to the total weight value of TSP in graph G. If and only if path length for data collection in graph G' of MerLEC has the minimum value, then, TSP in graph G will have the path with a minimum distance cost. Therefore, the MerLEC problem is a NP-hard Problem.

4.2. Time Complexity of MerLEC

Theorem 2. The time complexity of MerLEC is $O(n^3)$ under the worst circumstance.

Proof: Here we will analyze in detail the time complexity of MerLEC algorithm according to phases in MerLEC.

In the first phase, MerLEC first calculates the connected components of the entire network; n nodes calculate their neighbor nodes, and then according to the information of neighboring nodes, the nodes that can communicate through multi-hop method can be classified as a connected component, the time complexity is $O(n^2)$. Then, initially identified the target node, under the worst circumstance, the energy of all nodes is higher than the energy threshold, MerLEC needs to calculate the weight value of these n nodes, and then selects the node with the maximum weight value to add to the set of target nodes. The time complexity for centralized algorithm and distributed algorithm are all $O(n^2)$. Therefore, under the worst circumstance, in the first phase of centralized algorithm and distributed algorithm, the time complexity is $O(n^2)$.

In the second phase, MerLEC will determine the final set of target nodes. In a centralized algorithm, when the defined length is large enough, and there is only one connected component, and the energy of nodes is all higher than the threshold value, all the nodes will be determined as the target nodes in this step. Under the worst circumstance, MerLEC needs to calculate the weight value for all the nodes within the defined-length path, the time complexity of centralized algorithm is $O(n^3)$. When execute the distributed algorithm, under the worst circumstance, the time complexity for the management nodes of each connected component submitting the node with the maximum weight value to the base station is $O(n^3)$. The time complexity for the base station selecting node with the maximum weight value among the submitted nodes is $O(n^2)$. Therefore, under the worst circumstance, in the second phase of centralized algorithm, the time complexity is $O(n^3)$, and the time complexity of distributed algorithm is $O(n^3)+O(n^2)$.

In summary, under the worst circumstance, the time complexity of centralized MerLEC algorithm is $O(n^2)+O(n^3)$, that is to say, the time complexity of centralized algorithm is $O(n^3)$, and the time complexity of distributed MerLEC algorithm is $O(n^2)+O(n^3)+O(n^2)$, that is to say, the time complexity of distributed algorithm is $O(n^3)$.

5. Performance Evaluation

We conduct extensive simulations to evaluate the performance of MerLEC. The experiments are performed in a square field of 100×100, in which nodes are randomly dispersed. The parameters are: the energy dissipated for node to receive data is e_r=50nJ/bit, the energy dissipated for node to send data is nearly two times larger than the energy dissipated for node the receive data, thus, $e_t=2e_r$[15]. The other parameters are shown in table 1. Currently, integrated charging mode can acquire a highest total energy supply in each round. In order to analyze the effectiveness of MerLEC, this section will execute a series of simulation experiments of centralized MerLEC algorithm named MerLEC-CA, distributed MerLEC algorithm named MerLEC-DA, a typical integrated charging algorithm named WerMDG. The main experimental comparison is divided into the following three groups: the first group will test the influence of total energy supply and energy consumption in each round of data collection when the number of nodes is changing; the second group will analyze the influence of total energy supply and energy consumption in each round of data collection when the energy threshold is changing; the third group will analyze the tour path length of MWCV for each algorithm. In addition, the simulation is carried out on the Matlab platform and the experiments are performed 200 rounds, and their average values are taken as the final results.

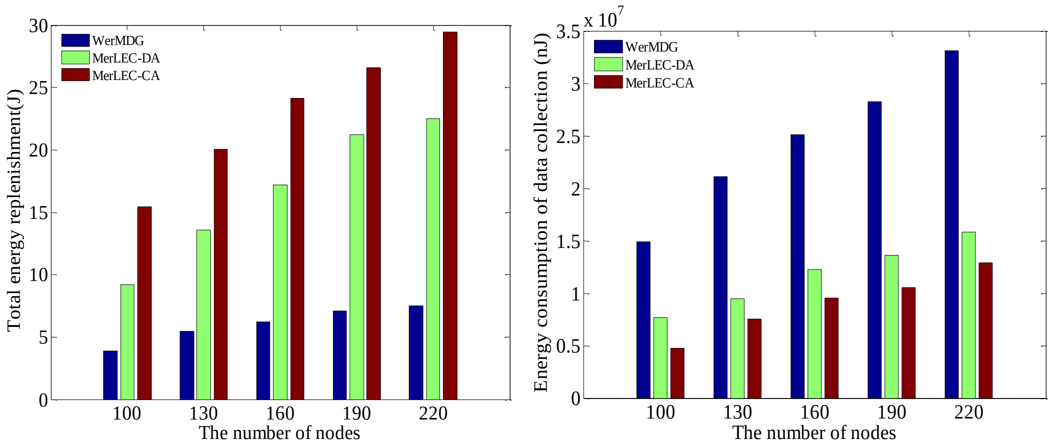
Table 1

Experimental Parameters

Notations	Values	Meanings
r	15m	communication radius
d_r	1024 bits	size of each packet
e_t	100nJ/bit	energy consumption of delivering 1 bit data
e_r	50nJ/bit	energy consumption of receiving 1 bit data
e	[0.1J,1J]	initial energy of node
L_{tsp}	400m	defined-length path

5.1. Results of total energy replenishment and energy consumption of data collection

The energy threshold used in this section is 4×0.01J; other experimental parameters are shown in table 1. We tested the number of nodes 100,130,160,190 and 220, respectively, the total energy replenishment and energy consumption of data collection for centralized algorithm MerLEC-CA, distributed algorithm MerLEC-DA, and integrated charging algorithm WerMDG will be tested. The results are shown in Fig. 2.



(a) Total energy replenishment (b) Energy consumption of data collection

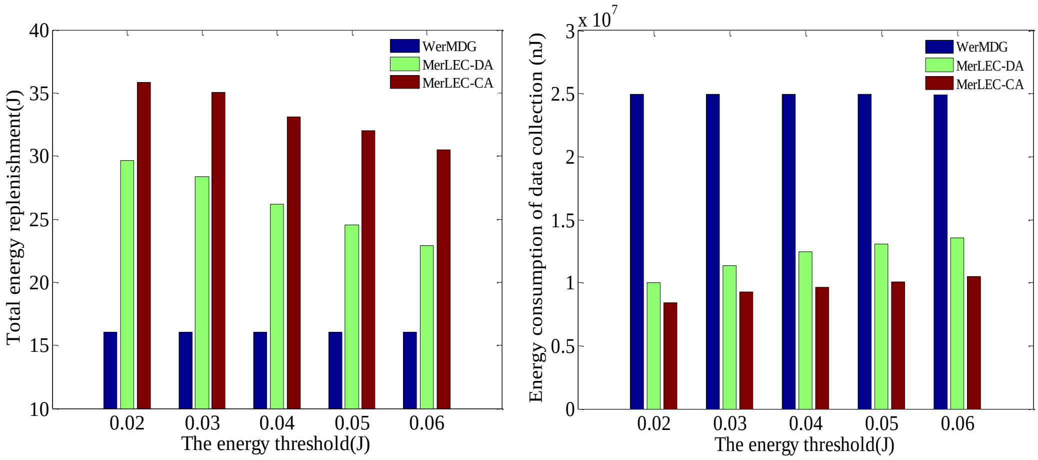
Fig.2 Comparisons of total energy replenishment and energy consumption of data collection

We can see from Fig.2(a), the total energy replenishment of these three algorithms is proportional to the number of nodes in the network. This is because the number of nodes increases, the number of low-energy nodes in the network increases too. WerMDG chooses some part of nodes with the lowest energy within a defined path, with the number of nodes increases, the energy of selected nodes in WerMDG further low, thus, the total energy replenishment of WerMDG increases with the increasing of the number of nodes. For centralized algorithm MerLEC-CA and distributed algorithm MerLEC-DA, an increase in the number of nodes in the network means that the optional selected nodes increase, thereby enabling the nodes with high weight value increase, thus, the total energy replenishment of these algorithms is proportional to the number of nodes in the network. In addition, we can find that the total energy replenishment of MerLEC-CA and MerLEC-DA has been significantly improved comparing to that of WerMDG, and with the number of nodes increases, this gap is more obvious. This is because these two algorithms are based on the ideal of weights; the principle of selecting target nodes is to charge the nodes with a shorter path to supply more energy. While the number of nodes in the network increases, there are more available nodes to choose for the base stations. This enables the weight value of nodes selected by base station increase, thus the total energy replenishment is higher than WerMDG algorithm.

On the other hand, we can see from Fig.2(b) that the energy consumption of data collection for these three algorithms is proportional to the number of nodes in the network. The main reason is that with the increase in the number of nodes in the network, there are more packets need to forward and the forwarding number of nodes also increase, resulting in the increase of energy consumption for data collection. In addition, as the number of nodes increases, MerLEC-CA and MerLEC-DA consume less energy in data collection than WerMDG. This is because MerLEC-CA and MerLEC-DA select target nodes based on weight value; the number of nodes selected by these two algorithms is larger than WerMDG, the number of root nodes is also increased. With the increased number of the root nodes, the spanning trees in the network will increase too, but the overall height of tree will reduce, which can reduce energy consumption of nodes when forwarding data in the non-aggregation mode. In addition, in the preliminary phase to select target nodes, the distribution of target nodes is more uniform, and ultimately the height of many formed data collection trees is closer to each other, which can reduce the energy consumption of the entire network in data collection. MerLEC-CA will produce more target nodes than MerLEC-DA, thus, the energy consumption of MerLEC-CA in data collection will be further reduced.

5.2. Impact of energy threshold for algorithms

In this section, the number of nodes is 160 and the other experimental parameters are shown in Table 1, we will test the total energy replenishment and energy consumption of data collection for RSEP and WerMDG. For a fair comparison, in each experiment, the network topology and the information of nodes (such as energy of nodes, etc.) are the same. The experimental results are shown in Fig. 3.



(a) Total energy replenishment (b) energy consumption of data collection

Fig.3 Performance comparison of different energy thresholds for algorithms

We can see from Fig. 3(a) and Fig. 3(b), the total energy replenishment and energy consumption of data collection for WerMDG in each round remains unchanged. This is because in each experiment, using the same algorithm WerMDG initial data, the increase in the power threshold does not affect the results for each round. In each experiment, WerMDG uses the same initial data; the increase of energy threshold will not influence the results for each round. When compare with WerMDG, MerLEC-CA and MerLEC-DA acquire the highest total energy replenishment and the lowest energy consumption in data collection when the energy threshold is set to 2×0.01J. With the increase in energy threshold, this gap is getting smaller and smaller. This is because when the energy threshold increases, the charging vehicles must access more nodes, the number of target nodes need to be selected in the initial stage is corresponding increase. In defined-path length, the number of remaining nodes selected according to the weight value will be corresponding decrease, so that the superiority of MerLEC algorithm is reduced. When the energy threshold continues to increase, the total energy replenishment and energy consumption of data collection for MerLEC-CA, MerLEC-DA and WerMDG will be getting closer and closer.

6. Conclusions

In this paper, we study the path planning program of charging vehicle in rechargeable sensor networks; we also maximize the total energy replenishment of charging vehicle in every round. The efficient energy supply strategy presents in this paper can reduce energy consumption in data collection, and at the same time the strategy can enhance the total energy replenishment. Simulation results show that the MerLEC can greatly increase the total energy replenishment while effectively reducing the energy consumption of data collection in each round, which especially applicable for higher density network.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (Grant no. 61562005), Guangxi Natural Science Foundation (Grant No. 2018GXNSFBA281169), and Thousands of Young and Middle-aged Backbone Teachers Training Program for Guangxi Higher Education (Education Department of Guangxi (2017) No.49).

REFERENCES

[1] Haipeng Dai, Qiufang Ma, Xiaobing Wu. CHASE: Charging and Scheduling Scheme for Stochastic Event Capture in Wireless Rechargeable Sensor Networks [J]. IEEE Transactions on Mobile Computing, 2018, 11(8): 101 - 115.

[2] Tengjiao He, Kwan-Wu Chin, Sieteng Soh, Changlin Yang. On Optimizing Max Min Rate in Rechargeable Wireless Sensor Networks with Energy Sharing [J]. IEEE Transactions on Sustainable Computing, 2018, 9(3): 64 - 78.

[3] Ruilong Deng, Shibo He, Peng Cheng, Youxian Sun. Towards balanced energy charging and transmission collision in wireless rechargeable sensor networks [J]. Journal of Communications and Networks, 2017, 19(4): 341 - 350.

[4] Jagdhuber T., Hajnsek I., Bronstert A.. Soil moisture estimation under low vegetation cover using a multi-angular polarimetric decomposition [J]. IEEE Transactions on Geoscience and Remote Sensing, 2013, 51(4): 2201-2215.

[5] Qian Zhang, Fan Li, Yu Wang. Mobile Crowd Wireless Charging Toward Rechargeable Sensors for Internet of Things. IEEE Internet of Things Journal, 2018, 5(6): 5337 - 5347.

[6] Lingfeng Liu, Maode Ma, Chunfeng Liu, Yantai Shu. Optimal Relay Node Placement and Flow Allocation in Underwater Acoustic Sensor Networks [J]. IEEE Transactions on Communications, 2017, 65(5): 2141 - 2152.

[7] Weifa Liang, Zichuan Xu, Wenzheng Xu. Approximation Algorithms for Charging Reward Maximization in Rechargeable Sensor Networks via a Mobile Charger [J]. IEEE/ACM Transactions on Networking, 2017, 25(5): 3161 - 3174.

[8] Fei-Yun Wu, Kunde Yang, Rui Duan, Tian Tian. Compressive Sampling and Reconstruction of Acoustic Signal in Underwater Wireless Sensor Networks [J]. IEEE Sensors Journal, 2018, 18(14): 5876 - 5884.

[9] Raipeng Dai, Yang Zhao, Guihai Chen. Robustly Safe Charging for Wireless Power Transfer [C]. Proceedings of IEEE Conference on Computer Communications (INFOCOM), 2018: 378 - 386.

[10] Chi Lin, Zhiyuan Wang, Jing Deng. mTS: Temporal-and Spatial-Collaborative Charging for Wireless Rechargeable Sensor Networks with Multiple Vehicles [C]. IEEE Conference on Computer Communications (INFOCOM), 2018: 99 - 107.

[11] Songtao Guo, Cong Wang, Yuanyuan Yang. Joint Mobile Data Gathering and Energy Provisioning in Wireless Rechargeable Sensor Networks [J]. IEEE Transactions on Mobile Computing, 2014, 13(12): 2836-2852.

[12] Yuanchao Shu, Hamed Yousefi, Peng Cheng, Jiming Chen, Yu (Jason) Gu, Tian He, Kang G. Shin. Near-Optimal Velocity Control for Mobile Charging in Wireless Rechargeable Sensor Networks [J]. IEEE Transactions on Mobile Computing, 2016, 15(7): 1699-1713.

[13] Chi Lin, Ding Han, Jing Deng, Guowei Wu, P2S: A Primary and Passer-By Scheduling Algorithm for On-Demand Charging Architecture in Wireless Rechargeable Sensor Networks [J]. IEEE Transactions on Vehicular Technology, 2017, 66(9): 8047-8058.

[14] Cormen T H, Leiserson C E, Rivest R L, et al. Introduction to Algorithms[M], London: MIT Press, 2001.

[15] Intanagonwiwat C, Govindan R, and Estrin D. Directed diffusion: A scalable and robust communication paradigm for sensor networks[C]. Proceedings of the 6th Annual Int'l Conference on Mobile Computing and Networking (MOBICOM). New York: ACM Press, 2000: 56−67.