Shortest path planning for mobile chargers

Huiyang Yi\textsuperscript{1,a}, Xiaojian Shen\textsuperscript{1*,b}, Jiaqi Wang\textsuperscript{2,a}

\textsuperscript{a}School of Mathematics and Statistics, Chongqing Jiaotong University, Chongqing, 400074, China
\textsuperscript{b}School of Information Science and Engineering, Chongqing Jiaotong University, Chongqing, 400074, China
\textsuperscript{*}Corresponding author’s e-mail: xiaojianshen@mails.cqjtu.edu.cn

Abstract. This paper studies the shortest path that a mobile charger should travel when charging a wireless rechargeable sensor network. Under the condition that the latitude and longitude of each node in the wireless rechargeable sensor network are known, we combine the actual working conditions of the mobile charger to establish a TSP problem model and use two improved modern optimization algorithms and three common optimization algorithms to solve. By comparing the results, we conclude that the improved genetic algorithm can obtain the shortest path length of 11485 m with the least number of iterations. The improved genetic algorithm has better applicability to the problem in this background.

1. Introduction
With the rapid development of wireless communication technology, WRSN has been widely used in various fields of daily life in recent years. The application of WRSN has become more and more extensive. WRSN effectively solves the problem of sensor energy replenishment by introducing mobile charger MC. The life cycle of networks depends on the path planning of mobile chargers, so MC’s charging path planning has become a hot topic of related research. Paper [1] proposed a multi-period charging planning strategy and an online algorithm for sensor charging using the attention mechanism, which effectively solved the problem of sensor death. Paper [2] designed an outlier firework algorithm based on population entropy to solve the wireless charging and data collection problems of mobile devices. Paper [3] designed a multi-objective ant colony optimization algorithm to solve the problem of data collection and charging of mobile charger. The simulation results showed that the nodes saved more energy when the energy was limited. Paper [4] proposed a path planning algorithm to minimize the number of MCs and a deployment algorithm for charging base stations. It overcame the limitations of the existing methods and fully considered the relationship between the operating cycle of MC and the operating life of sensor nodes. Compared with the existing methods, the results of this method were greatly improved. Paper [5] analyzed the energy consumption of each sensor node and determined the corresponding revenue. An ant system-based algorithm was proposed to solve the charging path with higher charging efficiency. In the above-mentioned research, scholars mostly put forward the wandering path strategy of MC charging sensor nodes and collecting node data from the perspective of open source and throttling. This comprehensive consideration does not work well in some basic environments.

Based on previous research, this article only considers the mobile charger to charge the sensor node. By analyzing the actual working condition of the mobile charger, the TSP problem model is established, and two improved optimization algorithms and three common optimization algorithms are
used to work out the problem. The modified genetic algorithm has better solution results. In the context of the wireless rechargeable sensor network, we conclude that the modified genetic algorithm has better applicability.

2. Problem Description
This paper analyzes a wireless rechargeable sensor network in a practical situation, and its geographical distribution scatters diagram is shown in Fig. 1. We know the latitude and longitude of 29 sensors and 1 data center. In Fig. 1, the blue dots represent sensors, and the red dots represent data centers. To effectively work out the issue of supplementing the energy of each sensor in WRSN, we can send a portable charger from the data center. The mobile charger drives at a constant speed according to a certain charging path, charges each sensor on the path, and finally returns to the data center. Next, the shortest charging path of the mobile charger is obtained by establishing a TSP model, so that the mobile charger can charge each sensor node at the fastest speed.

![Fig. 1 Scatter plot of node distribution](image)

3. Model establishment and solution

3.1. The introduction of the distance matrix
In order to plan an appropriate charging route so that the mobile charger can complete a charging task in the shortest distance, we need to introduce a distance matrix to represent the distance between any two sensors. When the latitude and longitude of any two points are known, we use the Haversine formula and the transformation of trig functions to calculate the distance between two points on the sphere [6], as shown in (1) and (2)

\[
\text{dist}(p_1, p_2) = R \times \Delta\sigma
\]

\[
\Delta\sigma = 2 \arcsin \left( \sqrt{\frac{\sin^2(x_1 - x_2)}{2} + \cos(x_1) \cos(x_2) \frac{\sin^2(y_1 - y_2)}{2}} \right)
\]

Among them, \(p_1, p_2\) represent any two points on the earth, the radius of the earth \(R=6372.795\) km, \(x_1, y_1, x_2, y_2\) represent the latitude and longitude of location 1 and location 2, respectively.

On this basis, we define a distance matrix \(D\) of 30×30. And number as follows, the data center number is 0, sensor 1 to sensor 29 are respectively numbered 1-29.

The element \(d_{ij}\) in \(D\) shows the distance between the element numbered \(i\) and the element numbered \(j\).
3.2. Model establishment

Objective function:

$$\min \sum_{i \neq j} d_{ij} x_{ij}$$  \hspace{1cm} (3)

Where $d_{ij}$ shows the distance between the two sensors $i$ and $j$, $x_{ij}=0$ or 1 ($x_{ij}=1$ means the path from sensor $i$ to sensor $j$, otherwise there is no).

Restrictions:

$$\sum_{j=1}^{n} x_{ij} = 1, \quad i = 1, 2, \ldots, n$$

$$\sum_{i=1}^{n} x_{ij} = 1, \quad j = 1, 2, \ldots, n$$

$$\sum_{i,j \in S} x_{ij} \leq |s| - 1, \quad 2 \leq |s| \leq n - 1$$

$$x_{ij} \in \{0, 1\}, \quad i, j = 1, 2, \ldots, n, \quad i \neq j$$

Where $n$ is the number of sensors, and $|s|$ represents the number of elements in the set and $s \subset \{1, 2, \ldots, n\}$. [7]

3.3. Hybrid particle swarm algorithm solution

We will only explain the improvement of hybrid particle swarm optimization algorithm on the basis of particle swarm optimization algorithm.

The initial velocity of each particle $v_0=0$, record the next iteration velocity of the $j$-th particle as $v^{(j)}$.

$$v^{(j)} = w \cdot v_0 + A + B$$  \hspace{1cm} (5)

$$A = c_1 \cdot \text{rand} \cdot (P^{(j)} - X^{(j)})$$  \hspace{1cm} (6)

$$B = c_2 \cdot [q \cdot \text{rand} \cdot (P_{G} - X^{(j)}) + (1 - q) \cdot \text{rand} \cdot (P_{L}^{(j)} - X^{(j)})]$$  \hspace{1cm} (7)

Where $B$ represents social factors, which are composed of global social factors and local social factors. It is called the radius of action of the local factor, which is related to the average density of particle distribution. $P_{L}^{(j)}$ is the local optimal solution, which can be understood as the position of the particle with the highest degree of adaptation among all particles contained in a sphere with the particle as the center and radius. $q$ is the proportion of global social factors, the higher the proportion, the smaller the weight of local social factors. [8]

3.4. Improved genetic algorithm solution

Same as the narrative form above, we take 102 genes per chromosome in the model as an example to illustrate the improved part of the genetic algorithm.

The specific design of our improved crossover operation is as follows: sort the parents according to the size of the fitness function, and then pair the large and large objective function values, and the small and small ones. We take a random initial value in the range of $(0,1)$, and iteratively generate 1 chaotic value in the range of $(0,1)$ using $g(n+1) = 4g(n)(1 - g(n))$. We save more chaos value to produce the next generation of cross-terms of the initial value of chaos iteration, and these values multiplied by 100 and 2, respectively, finally rounded up for a quick cross location.

Our improved mutation operation design is as follows: First, according to the given variation rate of 0.02, we randomly select two integers between 2 and 101, and use the gene value of the current position as the initial value to perform an appropriate number of iterations using
\( g(n + 1) = 4g(n) - 4g^2(n) \). We get the new gene value after mutation, and then get new chromosome. [9-10]

3.5. *Ordinary optimization algorithm solution*
At the same time, the ant colony algorithm, simulated annealing algorithm and tabu search algorithm were used to compare the solution results with the above improved algorithm, and the detailed steps of the solution will not be described.

4. *Model solution results*

4.1. *Comparison of results before and after improvement*

![Fig. 2 PSO roadmap](image1)
![Fig. 3 PSO iteration diagram](image2)
![Fig. 4 Hybrid PSO roadmap](image3)
![Fig. 5 Hybrid PSO iteration diagram](image4)

From the above results, it can be seen that the optimal path of the particle swarm algorithm is chaotic and the deviation between the average solution and the optimal solution is large. At the same time, as the iterations increase, the result becomes worse and worse. The optimized path of the improved particle swarm algorithm is clear, and it converges to a stable value of 11998 around the 140th generation. The result is better than the particle swarm algorithm.
It can be seen from Fig. 7 that the average solution of the genetic algorithm has a large deviation from the optimal solution, and the optimal solution converges to 13485 after iteration 200, and the trends of the two curves are almost the same. The improved genetic algorithm converges to a stable value of 11485 around the 17th generation, and its convergence speed is faster, and the result is better.

4.2. Ordinary optimization algorithm results

Results by above knowable, average solution of ant colony algorithm and the optimal solution has a certain difference. Both curves have the feature that the shortest distance decreases with the increase of iterations. Its optimal solution converges to a stable value of 11879 around the 25th generation.
Fig. 12 SA roadmap 
Observe the above results can be found that the average solution of simulated annealing algorithm and differ from that of the optimal solution. The optimal solution converges to a stable value of 11674 around the 25th generation, and the convergence rate is faster.

Fig. 13 SA iteration diagram 

Fig. 14 TS roadmap 
The above several tells us that the average solution of tabu search algorithm with small differences in the optimal solution, the two curves almost unanimously. The optimal solution converges to a stable value of 11686 around the 1250 generation, and the convergence speed is slow.

Fig. 15 TS iteration diagram

4.3. Selection of solution algorithm
After comparing the above solution results, we can see that the improved genetic algorithm can converge to the optimal value of 11485 with the least number of iterations. Therefore, we choose the improved genetic algorithm to solve the TSP problem model of mobile charger path planning in this background.

5. Model analysis
In this part, we mainly carry out the sensitivity analysis of the shortest distance to the population size $M$. We take the hybrid particle swarm algorithm as an example to illustrate. In a hybrid particle swarm algorithm to solve TSP problem, we set the population size $M=1000$ by ourselves. Next, we will analyze the sensitivity of the shortest loop distance to the population size $M$.

We keep the values of other parameters unchanged, change the value of $M$ every 50, observe the change of the shortest distance, and get the result as shown in Fig. 16.
Fig. 16 Sensitivity analysis results

It can be seen from Fig. 16 that when \( M < 900 \), the shortest distance of the loop decreases with the increase of \( M \), indicating that the system is not stable at this time and the result obtained is unreliable. But when \( M > 900 \), the shortest distance is almost constant, the system has stable results, proved that the hybrid particle swarm algorithm of \( M = 1000 \) is reasonable, and the result is correct and stable.

6. Conclusion

Based on the above results and discussion, the following conclusions are drawn: In the mobile charger path planning problem under this background, we take the number of iterations and the shortest path length as indicators, and compare the results of two improved optimization algorithms with multiple common optimization algorithms. It is concluded that the improved genetic algorithm can converge to the optimal result 11485 with the least number of iterations, while the hybrid particle swarm algorithm requires about 140 iterations to converge to 11998. Therefore, the improved genetic algorithm has better applicability to this kind of practical problem.

Acknowledgments

This study is particularly grateful for the support of the national college students innovation training program (project no. 202010618007).

References

[1] Chen, H., Yu, Y., Liu, Y., Wu, J. (2021) Charging wireless rechargeable sensor network utilizing attention mechanism. J. Huazhong Univ. of Sci. Tech., 49(02): 102-107.
[2] Wang, X., Wei, Z., Lv, Z., Ji, J. (2019) Periodic charging path planning based on PE-FWA algorithm in WRSNs. Journal of Electronic Measurement and Instrumentation., 33(05): 118-124.
[3] Wei, Z., Sun, R., Lv, Z., Han, J., Shi, L., Xu, J. (2018) Path planning algorithm for WCE with joint energy replenishment and data collection based on multi-objective optimization. Journal on Communications., 39(10): 22-33.
[4] Jiang, G., Lam, S., Sun, Y, et al. (2017) Joint charging tour planning and depot positioning for wireless sensor networks using mobile chargers. IEEE/ACM Transactions on Networking, 25(4): 2250-2266.
[5] He, L., Kong, L., Gu, Y, et al. (2015) Evaluating the on-demand mobile charging in wireless sensor networks. IEEE Transactions on Mobile Computing, 14(9): 1861-1875.
[6] Chopde, N. R., Nichat, M. (2013) Landmark based shortest path detection by using A* and Haversine formula. International Journal of Innovative Research in Computer and Communication Engineering, 1(2): 298-302.
[7] Chen, Y., Liu, J. (2006) Design and application of picking VRP model in warehouse. Computer Engineering and Applications, 6: 209-212.
[8] Yang, H., Gu, L. (2021) Research on global path planning of agricultural robot based on Hybrid Particle Swarm optimization. Agricultural mechanization research, 10: 33-36.

[9] Potts, J. C., Giddens, T. D., Yadav, S. B. (1994) The development and evaluation of an improved genetic algorithm based on migration and artificial selection. IEEE transactions on systems, man, and cybernetics, 24(1): 73-86.

[10] Dandy, G. C., Simpson, A. R., Murphy, L. J. (1996) An improved genetic algorithm for pipe network optimization. Water resources research, 32(2): 449-458.