Path Planning Method Based on a Single Mobile Beacon Node (Postprint)

Authors: Qiao Xuegong, Yaqing Duan

Date: 2018-12-13T00:00:00+00:00

Abstract

In wireless sensor networks based on mobile beacon nodes, the path planning problem for mobile beacon nodes directly affects localization performance. Existing path planning methods inadequately consider the distribution of unknown nodes within the network, leading to problems of low localization coverage and high network costs. To address these issues, this paper designs a path planning method based on a single mobile beacon node. The approach first determines the positions and number of virtual beacon nodes according to the distribution of unknown nodes in the network. Subsequently, an improved grey wolf optimization algorithm featuring a nonlinear dynamically varying convergence factor based on a Gaussian decreasing strategy is proposed to solve the path planning problem via the TSP algorithm, thereby obtaining the shortest mobile path for the beacon node. Simulation results demonstrate that the proposed method effectively enhances the localization coverage of unknown nodes within the network while significantly reducing network costs.

Full Text

Path Planning Method Based on Single Mobile Beacon Node

Qiao Xuegong, Duan Yaqing
(Institute of Information & Computer, Taiyuan University of Technology, Taiyuan 030024, China)

Abstract: In wireless sensor networks based on mobile beacon nodes, the path planning problem of mobile beacon nodes has important influence on positioning performance. However, existing path planning methods do not take full account of the distribution of unknown nodes in the network, resulting in low positioning efficiency and high cost. Therefore, this paper designs a path planning method for mobile beacon nodes. First, the positions and number of virtual beacon nodes are determined by making full use of sensor node distribution. Then, a grey wolf optimization algorithm with nonlinear dynamic convergence factor based on Gaussian decreasing strategy is proposed. This algorithm is applied to solve the path planning problem in the TSP algorithm and can obtain the shortest moving path for the mobile beacon node. Simulation results show that the proposed method can effectively improve localization coverage and save localization cost.

Key words: wireless sensor network; mobile beacon node; virtual beacon node; grey wolf optimization (GWO); path planning

---

0 Introduction

Wireless sensor networks (WSN) are typically deployed in specific areas to perform particular tasks, with node localization being one of the key fundamental technologies. Generally, more beacon nodes yield higher localization coverage. However, due to limited energy, size, and communication capabilities of nodes in WSN, which are usually randomly deployed in remote or harsh environments at relatively large scales, traditional localization methods using fixed beacon nodes impose high requirements on the number and distribution of beacon nodes, often leading to high overall network costs. Mobile beacon nodes can address this issue by using virtual beacon nodes to replace fixed ones. Through reasonable path planning for mobile beacon nodes, network costs can be reduced while overcoming the limitations of fixed beacon node localization, making the approach more practical. In mobile beacon-assisted localization, the beacon node's movement path significantly impacts node positioning.

Numerous studies have investigated static path planning methods. Reference [?] proposed SCAN and double-SCAN curve-based path planning methods for mobile beacon nodes, where the beacon node moves along pre-defined curves. The double-SCAN approach effectively solves the collinearity problem inherent in the SCAN method. Reference [?] introduced a Z-curve-based path planning method that avoids beacon node collinearity with shorter movement paths and lower energy consumption, but suffers from low unknown node localization coverage. These methods are static path planning approaches suitable for uniformly distributed network topologies but demonstrate poor applicability and low localization coverage in randomly distributed network topologies.

In response to these issues, dynamic path planning research has increased, aiming to reduce beacon node quantity, lower network costs, and improve efficiency. Reference [?] proposed a dynamic path planning method based on tabu search, where the mobile beacon node communicates with surrounding nodes during movement to dynamically select movement directions. However, for irregular areas with network disconnections, this method cannot obtain better movement directions based solely on unknown node density or tabu sets, resulting in redundant searches and lengthy movement paths. Reference [?] presented a path planning method using an improved virtual force approach to determine the beacon node' s next movement direction, which effectively overcomes the problem of lengthy paths caused by uneven unknown node distribution but achieves relatively low localization coverage. Additionally, reference [?] proposed a mobile sink path optimization method based on virtual point priority, which divides virtual points using a grid method and has the sink collect sensor node data along the shortest path solved by the TSP algorithm.

Obtaining the shortest path for mobile beacon nodes to reduce network energy consumption is a critical issue. Considering that sensor nodes have circular communication areas with radius equal to their communication range, the problem can be modeled as a traveling salesman problem (TSP) with circular neighborhoods.

This paper proposes a path planning method for a single mobile beacon node based on the distribution of unknown nodes, applicable to scenarios with non-uniform unknown node distribution. It effectively solves the problem of lengthy movement paths, improves localization coverage of unknown nodes in network topologies, and reduces network costs. The specific implementation combines sensor node distribution in the network to select stopping positions for the mobile beacon node (i.e., positions of virtual beacon nodes). The mobile beacon node then moves along the shortest path solved by a TSP algorithm based on improved grey wolf optimization. TSP is a classic NP-complete problem, and various improved swarm intelligence algorithms have been proposed for solving TSP [?, ?, ?, ?] with good results. GWO [?] is a novel intelligent optimization algorithm inspired by grey wolf pack hunting behavior. Reference [?] studied path planning based on GWO, and comparative results showed that the GWO-based TSP algorithm yields better paths. Therefore, in this paper, the mobile beacon node moves along the shortest path solved by a TSP algorithm based on improved GWO.

---

1 Virtual Beacon Node Selection

In wireless sensor networks based on mobile beacon nodes, the path planning problem directly affects localization performance. Existing path planning methods do not fully consider the distribution of unknown nodes in the network, resulting in low localization coverage and high network costs. Considering both localization performance and network cost, this paper establishes the following principles for virtual beacon node selection: (a) Minimize the number of virtual beacon nodes while ensuring localization coverage, as the number directly affects path length; (b) For localization performance, since sensor nodes require location information from at least three non-collinear beacon nodes to estimate their position through trilateration, the selection of virtual beacon node positions must avoid three nodes being collinear, any two being too close, or all three being too close to each other.

Let A, B, C be three beacon nodes and O be the node to be localized. If the three beacon nodes are collinear, as shown in [Figure 1: see original paper], or if two beacon nodes are close (e.g., A and B), as shown in [Figure 2: see original paper], node O may be localized at its mirror position O' . If A, B, and C are all close to each other, localization failure may occur, as shown in [Figure 3: see original paper]. Here, d_{A,B}, d_{A,C}, d_{B,C}, d_{O,A}, d_{O,B}, d_{O,C} represent distances between A and B, A and C, B and C, O and A, O and B, and O and C, respectively.

Therefore, this paper introduces parameter δ during virtual beacon node selection. A smaller δ indicates that the triangle formed by the three virtual beacon nodes is closer to an equilateral triangle, which provides the highest localization accuracy. Thus, the group with the minimum δ value is selected.

Sensor nodes can estimate their position only when receiving location information from at least three non-collinear beacon nodes. If a sensor node falls within the intersection area of three circles centered at three non-collinear beacon nodes with communication radius R, it can achieve self-localization. Based on this principle, the virtual beacon node selection process is as follows:

a) Randomly deploy N sensor nodes in the wireless sensor network, assigning IDs 1, 2, ···, N. With communication radius R, establish an N×N distance matrix D by converting signal strength values to distances using RSSI ranging through inter-node communication.

b) Group every three non-collinear sensor nodes, forming K groups numbered 1 to K. Create matrix G where G(i,j) stores the IDs of three sensor nodes in group i (j = 1, 2, 3).

c) Determine whether each group in matrix G is meaningful. A meaningful group is defined as having other sensor nodes within the common area of three circles centered at the group' s three sensor nodes with radius R that can communicate with each other. Create a K×N matrix S recording sensor nodes that can communicate with all three nodes in each group, where i represents group ID and j represents sensor node ID.

d) Calculate the sum of each row in matrix S, storing results in K$×$1 column vector LV. Each LV(i) represents the number of sensor nodes that can communicate with all three nodes in group i.

e) Reorder rows of matrix S in descending order of row sums, simultaneously updating rows of matrix G accordingly, maintaining one-to-one correspondence between group IDs.

f) Update S(i,j) by column: for j from 1 to N, starting from the first element in column j, if S(i,j) = 1, set all elements from row i to K in column j to 0.

g) Recalculate row sums of matrix S into LV. If LV(i) = 0, set G(i,j) = 0 (j = 1,2,3) and mark group i as redundant.

h) Process row by row starting from row i: if $0 < \text{LV(i)} < 3$, the three sensor nodes in group i are determined as virtual beacon nodes; if $\text{LV(i)} \geq 3$, group the column indices of elements with value 1 in row i of matrix S (i.e., sensor nodes) into sets of three non-collinear nodes. If no such group exists, the three sensor nodes in group i are determined as virtual beacon nodes. If such groups exist, calculate δ for each group using equations (1) and (2), and select the three sensor nodes with minimum δ as virtual beacon nodes.

2.1 Improved Grey Wolf Optimization Algorithm

The grey wolf optimization algorithm is a bionic swarm intelligence algorithm inspired by the social hierarchy and hunting behavior of grey wolf packs in nature. To mathematically model the social hierarchy, GWO considers the best solution as α, the second and third best solutions as β and δ respectively, and remaining candidates as ω. In GWO, hunting (optimization) is guided by α, β, and δ, with ω following these three wolves.

The first step in hunting is encircling prey, mathematically expressed as:

$$D = |C \cdot X_P(t) - X(t)| \quad (1) \tag{1}$$
$$X(t+1) = X_P(t) - A \cdot D \quad (2) \tag{2}$$

where X_P is the prey position, X is the grey wolf position, t represents current iteration, t_{max} is maximum iterations, D is encircling step length, and A and C are coefficient vectors:

$$A = 2a \cdot r_1 - a \quad (3) \tag{3}$$
$$C = 2 \cdot r_2 \quad (4) \tag{4}$$

The convergence factor a linearly decreases from 2 to 0 with iterations. r_1 and r_2 are random vectors in [0,1].

During hunting, other grey wolves update their positions based on α, β, and δ:

$$D_\alpha = |C_1 \cdot X_\alpha(t) - X(t)| \quad (5) \tag{5}$$
$$D_\beta = |C_2 \cdot X_\beta(t) - X(t)| \quad (6) \tag{6}$$
$$D_\delta = |C_3 \cdot X_\delta(t) - X(t)| \quad (7) \tag{7}$$

$$X_1(t) = X_\alpha(t) - A_1 \cdot D_\alpha \quad (8) \tag{8}$$
$$X_2(t) = X_\beta(t) - A_2 \cdot D_\beta \quad (9) \tag{9}$$
$$X_3(t) = X_\delta(t) - A_3 \cdot D_\delta \quad (10) \tag{10}$$

$$X(t+1) = \frac{X_1(t) + X_2(t) + X_3(t)}{3} \quad (11)$$

In the attack phase, exploration and exploitation are controlled by A and C. During iterations, convergence factor a gradually decreases, reducing the fluctuation range of A, which is a random value in [-a, a]. When $|A| \leq 1$, the grey wolf pack attacks prey for local search; when $|A| > 1$, the pack moves away from prey to enhance global exploration. From equation (7), C remains random throughout iterations, randomly strengthening or weakening prey' s influence in the distance equation and continuously enhancing search to avoid local optima.

This paper improves GWO by modifying the convergence factor. The standard GWO uses linear decreasing strategy for a, which does not match the actual nonlinear optimization process.

Reference [**?**] proposed nonlinear dynamic convergence factor update methods based on cosine (cosGWO) and quadratic functions (2GWO) following equations (12) and (13), with simulation results showing superiority over sine, tangent, and logarithmic functions.

$$a(t) = a_{initial} - (a_{initial} - a_{final}) \times \cos\left(\frac{t}{t_{max}}\right) \quad (12)$$

$$a(t) = a_{initial} - (a_{initial} - a_{final}) \times \left(\frac{t}{t_{max}}\right)^2 \quad (13)$$

Reference [**?**] proposed an exponential function-based method (eGWO):

$$a(t) = a_{initial} \times \exp\left(-\lambda \cdot \frac{t}{t_{max}}\right) \quad (14)$$

where $a_{initial}$ and a_{final} are initial and final values of a, t is current iteration, t_{max} is maximum iterations, and λ is a 调节系数 (adjustment coefficient) with reference value.

This paper proposes a Gaussian decreasing strategy for nonlinear dynamic convergence factor:

$$a(t) = a_{initial} \times \exp\left[-e \cdot \left(\frac{t}{t_{max}}\right)^2\right] \quad (15)$$

where $a_{initial}$ and a_{final} are initial and final values. When $a_t - a_{final} < T$, $a_t = a_{final}$, with T being the stopping threshold (T = 0.001) and e being the extension constant (e = 0.6). This provides high global search capability early, then rapidly decreases a for local search, accelerating convergence while maintaining strong local search capability later to improve convergence precision. To verify effectiveness, comparative experiments are conducted in the simulation section with $a_{initial} = 2$, $a_{final} = 0$.

Algorithm Pseudocode:

```
Initialize iteration count (t)
Initialize grey wolf population Xi (i = 1, 2, ⋯, N)
Initialize a, A, and C
Calculate fitness of each search agent
X$\alpha$ = best search agent
X$\beta$ = second best search agent
X$\delta$ = third best search agent
while (t $\leq$ max iterations)
    for each search agent
        Calculate convergence factor a by equation (16)
        Update $\omega$ position by equations (7)~(10) and (14)
    end for
    Update a, A, and C
    Calculate fitness of all search agents
    Update X$\alpha$, X$\beta$, and X$\delta$
    t = t + 1
end while
return X$\alpha$
```

2.2 Fitness Function Design

The improved grey wolf optimization-based TSP algorithm solves for the optimal movement path. Solution evaluation typically involves designing a fitness function based on the objective function. In this paper, shorter movement paths for mobile beacon nodes result in lower network energy consumption and extended network lifetime. Therefore, the objective function is minimum path length. Based on the permutation order of grey wolf individuals, the total distance between individuals (i.e., path length) is calculated. The reciprocal of path length is set as the fitness value—higher fitness indicates better individuals.

2.3 Path Planning Process Based on Improved GWO Algorithm

a) Initialize population: grey wolf individuals are encoded as random permutations of 1 to n (virtual beacon node count), with population size N. Set current iteration t = 0 and maximum iterations tmax. Initialize control factor a with initial value 2 and final value 0.

b) Calculate fitness value for each individual. Sort individuals by fitness in descending order, setting the first as α, second as β, third as δ, and remaining as ω.

c) Update individual positions using equations (7)~(10) and (14).

d) Calculate fitness values and update α, β, δ individuals. Increment t = t + 1.

e) If t > tmax, stop searching; otherwise, return to step b).

---

3 Simulation Experiments and Algorithm Analysis

To verify algorithm effectiveness and rationality, simulation experiments were conducted using MATLAB 2016(b). Simulation conditions: 100 sensor nodes randomly distributed in a 100 m × 100 m square area, beacon node communication radius R, population size 100, maximum iterations 1,000.

[Figure 5: see original paper]~[Figure 7: see original paper] show the mobile beacon path planning for R = 30 m, R = 40 m, and R = 50 m respectively, all achieving localization coverage of 1. [Figure 8: see original paper] compares the number of virtual beacon nodes across different network topology boundary lengths between the proposed method and traditional SCAN, DOUBLE-SCAN, and SLMAT algorithms from reference [?]. SLMAT combines LMAT [?] and SCAN, with virtual beacon nodes at equilateral triangle vertices and side length $\sqrt{3}$ times the communication radius, following SCAN-like movement. The simulation compares virtual beacon node counts at different boundary lengths when R = 40 m.

[Figure 9: see original paper] compares localization coverage across different R values. [Figure 10: see original paper] compares localization coverage for different numbers of sensor nodes in a 100 m square area with R = 30 m. [Figure 11: see original paper] compares movement path lengths across different R values for 100 nodes. [Figure 12: see original paper] compares movement path lengths across different network topology boundary lengths for 100 nodes with R = 40 m.

In [Figure 5: see original paper]~[Figure 7: see original paper], solid circles represent virtual beacon node positions and hollow circles represent unknown node positions. Comparison shows that larger communication radius reduces virtual beacon node count and path length, verified by [Figure 11: see original paper]. The proposed method achieves good localization coverage while effectively eliminating redundant movement.

[Figure 8: see original paper] shows the proposed algorithm' s virtual beacon node count changes little with network topology boundary length, occasionally decreasing because the count is determined by unknown node distribution. When network area increases with constant communication radius, coverage decreases, resulting in fewer virtual beacon nodes. SCAN, DOUBLE_{SCAN}, and SLMAT algorithms determine virtual beacon node count based on boundary length and communication radius, with simulation results showing significant increases with boundary length. The proposed algorithm depends primarily on unknown node distribution, making it less affected by boundary length and more widely applicable.

[Figure 9: see original paper] shows localization coverage increases with communication radius. When R $<$ 40 m, the proposed algorithm achieves much higher coverage than SCAN, DOUBLE_{SCAN}, and SLMAT. When R $>$ 40 m, the proposed algorithm' s coverage approaches 1, SLMAT reaches over 95%, and SCAN/DOUBLE_{SCAN} exceed 90%. In practice, beacon node communication radius is limited, making the proposed algorithm more practical and better performing.

[Figure 10: see original paper] demonstrates the proposed algorithm maintains higher localization coverage than SCAN, DOUBLE_{SCAN}, and SLMAT across different sensor node counts, confirming good performance.

[Figure 11: see original paper] and [Figure 12: see original paper] show the proposed improved GWO-based TSP algorithm produces shorter paths than GWO with quadratic, cosine [**?**], and exponential [**?**] nonlinear convergence factor update methods under different communication radii and network boundary lengths, verifying higher precision and stronger optimization capability. Mobile beacon nodes moving along the shortest path solved by the improved GWO-based TSP effectively save network energy and extend network lifetime.

The proposed method achieves shorter movement paths than SCAN, DOUBLE_{SCAN}, and SLMAT under equivalent conditions. While those algorithms' path lengths are determined by network boundary length and communication radius (increasing with boundary length), the proposed algorithm is less affected by boundary length, demonstrating greater practicality.

---

4 Conclusion

This paper proposes a mobile beacon node path planning method that selects stopping positions based on unknown node distribution in the network, aligning with dynamic topology characteristics of WSN and offering strong flexibility. The algorithm reduces beacon node stopping positions, with mobile beacon nodes moving along the shortest path solved by the TSP algorithm based on nonlinear dynamic convergence factor GWO, saving beacon node energy, extending network lifetime, improving system availability, and reducing network costs. The algorithm also improves WSN localization coverage, verified through simulations. The beacon node positions approach equilateral triangles when covering nodes to be localized, theoretically improving localization precision. However, this paper focuses on path planning with insufficient research on localization technology; designing high-precision localization algorithms is the main direction for future work.

References

[1] Suja G J, Jose S, Raman R B, et al. Performance analysis of big data gathering in wireless sensor network using an EM based clustering scheme [C]//Proc of the 5th International Conference on Advances in Computing and Communications. 2015: 109-113.

[2] Koutsonikolas D, Das S M, Hu Y C. Path planning of mobile landmarks localization in wireless sensor networks [J]. Computer Communications, 2007, 30(13): 2577-26592.

[3] Rezazadeh J, Moradi M, Ismail A S, et al. Superior path planning mechanism for mobile beacon-assisted localization in wireless sensor networks [J]. IEEE Sensors Journal, 2014, 14(9): 3052-3064.

[4] Yin Wenzheng, Jiang Weidong, Tao Jin. Strategy of dynamic path planning based on tabu search algorithm for AUV [J]. Journal of Nanjing University: Natural Sciences, 2017, 53(1): 144-150.

[5] Geng Feng, Xue Shengjun. Mobile beacon node path scheme in arbitrary region for wireless sensor networks [C]//Proc of International Conference on Electronic and Mechanical Engineering and Information Technology. 2011: 12-14.

[6] Zhang Xiwei, Shen Lin, Jiang Yifeng. Optimizing path selection of mobile Sink nodes mobility-assistant WSN [J]. Journal on Communications, 2013, 34(2): 85-93.

[7] Elloumi W, Abed H E, Abraham A, et al. A comparative study of the improvement of performance using a PSO modified by ACO applied to TSP [J]. Applied Soft Computing, 2014, 25: 234-241.

[8] Wang Kefu, Huang Quanzhen, et al. Improved fruit fly optimization algorithm for TSP problems [J]. Computer Engineering and Design, 2014, 35(8): 2789-2792, 2821.

[9] Wang Peidong, Tang Gongyou, Yang Xixin, et al. An improved ant colony algorithm for traveling salesman problems [J]. Periodical of Ocean University of

China: Natural Sciences, 2013, 43(1): 93-97.

[10] Ma Jihui, Xu Mingjie, Chen Xinjie, et al. Hama-based parallel ant colony optimization algorithm mode and TSP applied research [J]. Journal of Transportation Systems Engineering and Information Technology, 2016, 16(3): 168-173, 180.

[11] Mirjalili S, Mirjalili S M, Lewis A. Grey wolf optimizer [J]. Advances in Engineering Software, 2014, 69: 46–61.

[12] Zhang Sen, Zhou Yongquan, Li Zhiming, et al. Grey wolf optimizer for unmanned combat aerial vehicle path planning [J]. Advances in Engineering Software, 2016, 99(C): 121-136.

[13] Wei XuePeng, Zhenglei, Zhao Hui, Li Mudong, et al. A grey wolf optimization algorithm based on nonlinear adjustment strategy of control parameter [J]. Journal of Air Force Engineering University: Natural Science Edition, 2016, 17(3): 68-72.

[14] Luo Jia, Tang Bin. Grey wolf optimization algorithm based on nonlinear convergence factor dynamic changing [J]. China Sciencepaper, 2016, 11(17): 1991-1997.

[15] Han Guangjie, Yang Xuan, Liu Li, et al. A disaster management-oriented path planning for mobile anchor node-based localization in wireless sensor networks [J]. IEEE Trans on Emerging Topics in Computing, 2017: 1-1.

[16] Han Guangjie, Yang Xuan, Liu Li, et al. Path planning using a mobile anchor node based on trilateration in wireless sensor networks [J]. Wireless Communications and Mobile Computing, 2013, 13(14): 1260-1276.

Note: Figure translations are in progress. See original paper for figures.