

Positioning algorithm based on improved dragonfly optimization

Yue Zhang^{1,2}, Hongping Pu^{1,2} and Wei Chen^{1,2}

¹ School of Automation and Information Engineering, Sichuan University of Science and Engineering, Yibin, 644005, Sichuan, China

² Artificial Intelligence Key Laboratory of Sichuan Province, Sichuan University of Science and Engineering, Yibin 644005, Sichuan, China.

ABSTRACT

Aiming at solving the nonlinear equation of indoor arrival time difference positioning, a multi-strategy improved dragonfly optimization algorithm is proposed. The initial population is improved by chaotic mapping, and then nonlinear factors and Cauchy mutation operators are introduced to rapidly converge the balanced global search and local search. At the same time, simulation and comparison experiments with other algorithms show that the algorithm has a higher positioning effect.

KEYWORDS

Dragonfly algorithm; Positional accuracy; Cauchy disturbance.

1. INTRODUCTION

With the continuous development of wireless sensor technology, people have increasingly high requirements for positioning accuracy, which has made indoor positioning a hot research topic. However, there are various obstacles in indoor environments that can cause non-line-of-sight errors and affect the accuracy of indoor positioning. Therefore, achieving an accurate wireless positioning system first requires obtaining target location information. When conducting indoor positioning, wireless sensor technology can be used to collect and process signal data. By deploying a series of sensor nodes distributed in space and combining corresponding algorithms and models, estimation and inference of target positions can be achieved. This includes receiving information such as signal strength and time delay from the target device's transmissions and comparing them with a pre-established reference database for matching purposes. Various methods and techniques can also be employed to improve indoor positioning accuracy. For example, in known map or floor plan situations, fingerprint recognition algorithms can be used to match received signal features with corresponding position features recorded in a pre-existing fingerprint library; other types of sensors such as sound, optical or inertial sensors can also be fused with machine learning methods. However, there are still challenges and limitations in practical applications. The complex and variable nature of indoor environments along with numerous sources of interference (such as walls or furniture) may lead to signal attenuation, reflection or even blocking issues; difficulties related to multipath effects causing aliasing problems or timing estimation errors also need to be addressed. Overall, as wireless sensor technology continues to advance through innovation and development, people's demands for indoor positioning accuracy are increasing day by day. By comprehensively applying various methods and techniques while optimizing designs based on specific scenario requirements it is possible to obtain target location information more accurately while meeting the growing needs of individuals.

Time Difference of Arrival (TDOA) is a positioning method based on the principle of hyperbolic curves, which determines the location of a target by identifying the hyperbolic curve formed by two base stations at equal distances. However, it only requires synchronization between the base stations to achieve accurate positioning of the target.

$$R_i = \sqrt{(x_i - x)^2 + (y_i - y)^2} \quad i = 1, 2, 3 \quad (1)$$

$$\begin{aligned} R_{i,1} &= c(ti - t1) = R_1 - R_2 \\ &= \sqrt{(x_i - x)^2 + (y_i - y)^2} - \sqrt{(x_1 - x)^2 + (y_1 - y)^2} \quad i = 1, 2, 3 \end{aligned} \quad (2)$$

(x_i, y_i) is the coordinate of the base station, (x, y) is the anchor point coordinate, $R_{i,1}$ is the positioning difference of the time difference between the fixing point and the fixed base station.

2. DRAGONFLY OPTIMIZATION ALGORITHM

Dragonfly algorithm (DA) is a new intelligent optimization algorithm proposed by Seyedali Mirjalili in 2016 [1]. Its main inspiration comes from the static and dynamic clustering behavior of dragonflies in nature, and it has the characteristics of strong optimization ability. Dragonfly algorithm is a new intelligent swarm optimization algorithm, whose principle is to simulate the behavior of dragonfly hunting prey in nature.

Dragonfly life habits can be summarized into five types of behavior: separation, queuing, alliance, hunting for prey and avoiding predators.

(1) Separation is the act of separating each individual dragonfly from its own kind

$$S_i = -\sum_{j=1}^N (X - X_j) \quad (3)$$

In the formula N is the number of neighboring individuals, S_i is the position vector of the separation behavior between the I-to dragonfly species.

(2) The queue is the speed match between each individual dragonfly and its neighbors during flight

$$A_i = \frac{\sum_{j=1}^{N'} V_j}{N'} \quad (4)$$

A_i Is the position vector of the queue behavior of the I-th dragonfly individual, V_j is the flight speed of adjacent individuals.

(3) Alliance refers to the group behavior of dragonflies and their neighbors coming together with each other

$$C_i = \frac{\sum_{j=1}^{N'} X_j}{N'} - X \quad (5)$$

C_i Is the position vector of the alliance behavior of the I-th dragonfly individual.

(4) Hunting is the act of searching for prey in order to survive

$$F_i = X^+ - X \quad (6)$$

F_i is the position vector of the hunting behavior of the I-th individual dragonfly, X^+ is the position of the prey to be hunted.

(5) Avoiding natural enemies is an individual's survival instinct, and it is necessary to be always alert to the behavior of natural enemies.

$$E_i = X^- + X \quad (7)$$

E_i Is the position vector of the behavior of the i-dragonfly individual to escape predators, X^- is where the dragonfly hunts.

The step vector represents the dragonfly's flight direction and step length.

$$\Delta X_{t+1} = (sS_i + aA_i + cC_i + fF_i + eE_i) + \omega' \Delta X_t \quad (8)$$

In nature, dragonflies are in motion most of the time for survival, so their location needs to be updated in real time. Update the vector of dragonfly individual location.

$$X_{t+1} = X_t + \Delta X_{t+1} \quad (9)$$

To achieve the purpose of further strengthening the performance of the algorithm, when there is no adjacent solution near the same type of individuals, The Levy flight method was used to fly around the search space to update the dragonfly position.

$$X_{t+1} = X_t + \text{levy}(d)\Delta X_t \quad (10)$$

3. IMPROVED DRAGONFLY OPTIMIZATION ALGORITHM

This paper presents a new Dragonfly optimization algorithm based on mixed theory. Common Dragonfly algorithm, like other intelligent optimization algorithms, is easy to fall into local optimal rather than global optimal in the process of re-iteration. To solve this problem, the chaotic function is introduced in the re-initialization process to get a more comprehensive initialization population, and the nonlinear factor is introduced to improve the convergence speed of the algorithm.

1) Based on Bernoulli chaos function

Population initialization is an important part of intelligent algorithm, which will affect the quality and efficiency of the solution. The original dragonfly optimization algorithm used to randomly generate the initial position of the initial population, which made it impossible to obtain the target positioning stably. Chaotic mapping has the characteristics of randomness, ergodicity, and regularity, which can effectively maintain the diversity of the initial population and effectively avoid the algorithm prematurity.

$$z_{k+1} = \begin{cases} z_k / (1 - \lambda) & z_k \in (0, 1 - \lambda) \\ (z_k - 1 + \lambda) / \lambda & z_k \in (1 - \lambda, 1) \end{cases} \quad (11)$$

$$x_j^i = lb + (1 + z_{k+1}) * (ub - lb) / 2 \quad (12)$$

2) Nonlinear dynamic factors are introduced.

In the original Dragonfly algorithm, the inertia weight formula is a linearly changing value, and the calculation formula is shown as follows. However, with the increase of the number of iterations of the algorithm, the convergence speed becomes lower, and the search accuracy becomes worse. Literature [] introduces dynamic adaptive weights to achieve the purpose of balancing exploration and development by adjusting the search space adaptively.

$$\omega = 0.9 - t(0.5 / t_{\max}) \quad (13)$$

$$\omega_1 = 2 * \exp(-5 * (\frac{t}{t_{\max}})) \quad (14)$$

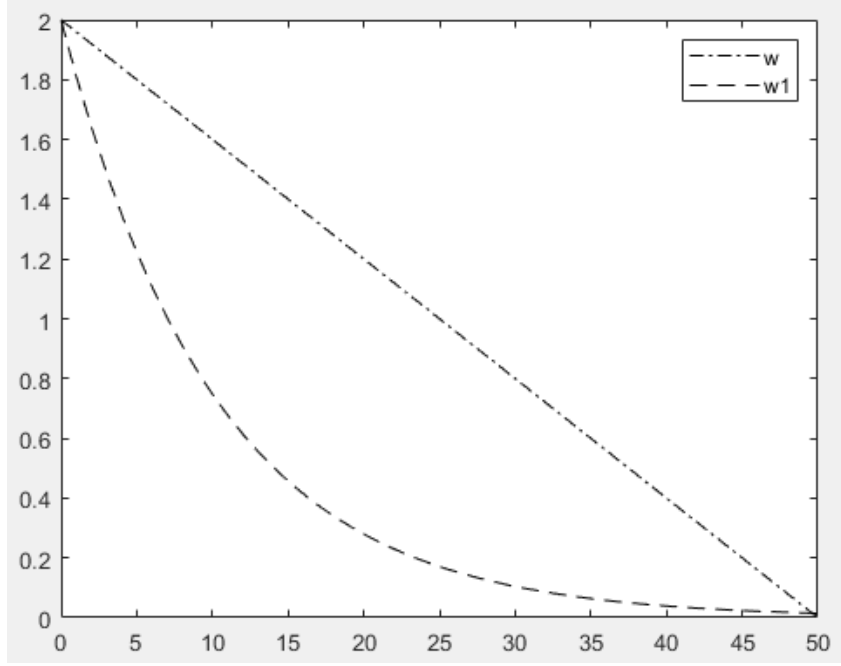


Figure 1. Change curve of density factor

3) Fusion Cauchy disturbance

The later stage of the re-algorithm will fall into the local optimal, which is not conducive to the global optimal value, so the algorithm in this paper introduces the Cauchy mutation operator. The Cauchy distribution is a continuous probability distribution common in probability theory, where the probability density is high in the middle and low in the two ends. Therefore, we introduce the Cauchy mutation operator, use Cauchy mutation to perturbate and mutate new positions in the updated locations, expand the search range of the Dragonfly optimization algorithm, and further improve the global optimal ability of the Dragonfly algorithm.

$$x_{new} = x_{best} + x_{best} * Cauchy(0,1) \quad (15)$$

$Cauchy(0,1)$ is the standard Cauchy distribution function, As shown in the formula.

$$f(x) = \frac{1}{\pi(x^2 + 1)} \quad (16)$$

The Cauchy mutation operator can effectively maintain the diversity of the population, improve the global search ability of the algorithm, and avoid falling into the local optimal.

4. TDOA LOCALIZATION BASED ON IMPROVED DRAGONFLY ALGORITHM

(1) The improved dragonfly algorithm is applied to TDOA localization.

Step 1: Population initialization is performed using chaos function according to the upper and lower bounds of the population.

Step 2: Calculate the fitness function through the fitness formula.

Step 3: Position update is carried out by the improved Dragonfly algorithm in this paper.

Step 4: Determine whether the iteration conditions are met. If they are not met, iterate through Steps 3 and 4. When satisfied, the current optimal position is output as the initial value of Taylor's technique expansion.

Step 5: The final positioning target position is obtained through continuous iteration of Taylor series expansion algorithm.

(2) Contrast simulation experiment

In this paper, the performance simulation test of the multi-strategy improved Honey Badger algorithm and the combined Taylor algorithm is carried out in the Matlab2021b environment. In the simulation scenario, eight base stations are set within the range of 10m*10m, and the base stations are respectively (0,0), (0,5), (5,0), (10,0), (10, 0),

and (10, 10) Search upper bound is [10,10], search lower bound is [0,0], as shown in Figure 5, T0 is the positioning target, BS is the positioning base station.

Root Mean Square Error (RMSE) is chosen as the evaluation index of this paper.

$$RMSE = \sqrt{\frac{1}{m} \sum_{m=1}^m (x_i - x'_i)^2 + (y_i - y'_i)^2} \quad (17)$$

The population size was 50 and the number of base stations was 8. Considering the accuracy of the indoor positioning system, the distance noise error was set within 0.1 to 1m for the experiment. Simulation diagrams of positioning accuracy under different distance noise standards are obtained. As shown in the figure, the RMSE of different algorithms increases with the increase of the standard deviation of distance noise, and the proposed algorithm achieves the lowest positioning error in the environment of different distance noise standards.

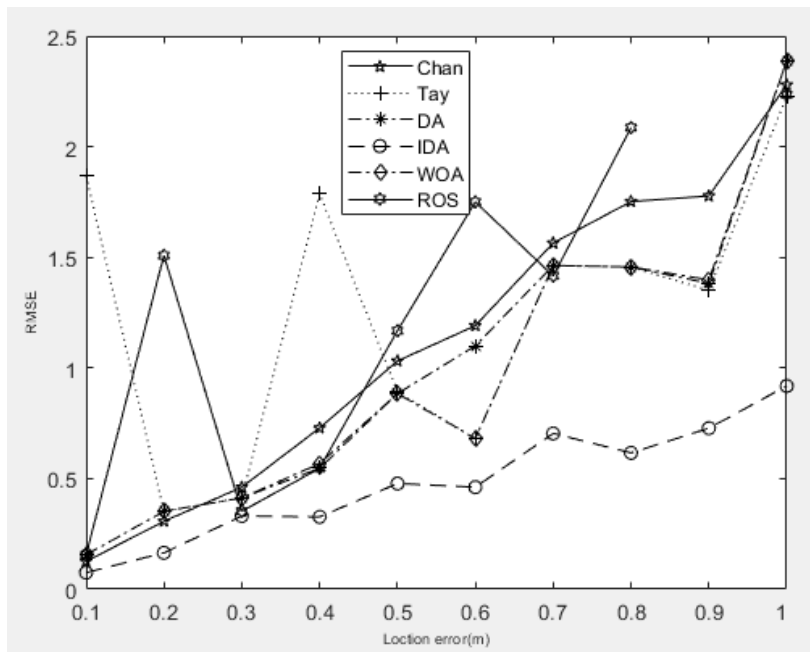


Figure 2. Comparison of positioning accuracy under different noises

It can be seen from Figure 2 that the dragonfly optimization algorithm improved in this paper has lower positioning errors under different positioning error standards compared with Chan algorithm, Taylor algorithm, Whale algorithm (WOA) and rat colony algorithm (ROS) algorithm.

5. CONCLUSION

In this paper, the dragonfly optimization algorithm is improved through multi-strategy. Through simulation, the improved optimization algorithm has lower positioning error compared with the traditional Chan algorithm, Taylor algorithm and the new start-element optimization algorithm.

REFERENCES

- [1] Mirjalili S. Dragonfly algorithm: a new meta-heuristic optimization technique for solving single-objective, discrete, and multi-objective problems[J]. *Neural Computing and Applications*, 2016, 27(4): 1053-1073.
- [2] Yi S. Exploring and evaluating the key technologies in TDOA-Based indoor positioning[J]. *Highlights in Science, Engineering and Technology*, 2023, 68: 108-114.
- [3] Dong J, Lian Z, Xu J, Yue Z. An improved adaptive sparrow search algorithm for TDOA-Based localization[J]. *ISPRS International Journal of Geo-Information*, 2023, 12(8): 334.
- [4] Liu H, Chen D, Lin F, et al. Wind Power Short-Term Forecasting Based on LSTM Neural Network with Dragonfly Algorithm[J]. *Journal of Physics Conference Series*, 2021,1748(3):032015.
- [5] Dongsheng Y, Mingling W, et al. Dynamic opposite learning enhanced dragonfly algorithm for solving large-scale flexible job shop scheduling problem[J]. *Knowledge-Based Systems*, 2022,238(28):107815.