Research on Positioning of Carrier-Based Aircraft Based on Particle Filter Optimized by Firefly Algorithm

Yajie Du*, Jianjun Zhao and Yi Wang
Naval Aeronautical University, Yantai 264001, China

*Corresponding author e-mail: 2219753715@qq.com

Abstract. In order to improve the positioning accuracy of the landing guidance radar, particle filter algorithm is used to estimate the positioning of the carrier aircraft. In order to solve the problem of the degradation of particle weight and the decrease of filter precision caused by the dilution of sampling particles in the conventional particle filter algorithm, an intelligent optimization particle filter algorithm based on firefly algorithm is proposed. By introducing the optimization mechanism of firefly algorithm into the conventional particle filter algorithm, particles can move to the high likelihood region, improve the overall quality of particle swarm, avoid particle degradation, and improve the filtering accuracy. The experimental results show that the algorithm is more effective and practical than the conventional particle filter algorithm.

1. Introduction
When landing, the carrier aircraft cannot do without the precise guidance of landing guidance radar. Because of the complex marine environment, and the aircraft carrier is a non-linear moving platform, the positioning error is large. It is very important to improve the positioning accuracy of carrier aircraft for landing safely. Particle filter algorithm as the main method to solve the nonlinear [1], non-Gaussian system parameter estimation and state estimation, particle filter algorithm as the core positioning method is widely used [2]. However, due to the particle degradation and sample dilution of particle filter, the positioning accuracy is not ideal [3].

In recent years, with the development of intelligent group optimization algorithm, more and more scholars apply intelligent optimization algorithm to particle filter (PF) [4-6]. At present, domestic and foreign scholars have successfully applied intelligent optimization algorithms such as genetic algorithm, particle swarm algorithm, ant colony algorithm and fish swarm algorithm to PF optimization [7-10]. It can effectively improve the lack of particle diversity and improve the estimation accuracy to a certain extent, but the calculation is large and the real-time performance is poor.

Based on the above analysis, this paper proposes a firefly algorithm to optimize particle filter. The combination of firefly algorithm and particle filter, using the optimization mechanism of firefly algorithm, improves the effectiveness of particles, avoids the phenomenon of particle degradation, and greatly improves the filtering accuracy of particle filter algorithm. Finally, through the simulation experiment, the improved particle filter algorithm is applied to the positioning of shipboard aircraft, and the effectiveness of the algorithm is verified.
2. Particle filter algorithm

Particle filter is an approximate estimation algorithm based on the large number theorem of Bayesian theory, which combines Monte Carlo and Bayesian theory. The basic idea is to find a group of random samples in the state space, estimate the posterior probability density function approximately according to the updated particle weights and position information pairs, and replace the integral operation with the sample mean value to obtain the process of the minimum variance estimation of the state \([11]\). The description of the assumed dynamic nonlinear system is as follows:

\[
\begin{align*}
    x_k &= f(x_{k-1}, w_{k-1}) \\
    z_k &= h(x_k, v_k)
\end{align*}
\]

Where: \(x_k \in \mathbb{R}^n\) and \(z_k \in \mathbb{R}^n\) are the state vector and observation vector of the system at \(k\) time; \(f(\cdot)\) and \(h(\cdot)\) are the nonlinear state transfer function and observation function of the system; \(w_k \in \mathbb{R}^n q_k, Q_k\) and \(v_k \in \mathbb{R}^n r_k, R_k\) are the process noise and observation noise of the system respectively.

If the initial probability density is \(p(x_0 | y_{i_0}) = p(x_0)\), the prediction and update process is as follows:

\[
\begin{align*}
    p(x_k | z_{k-1}) &= \int p(x_k, x_{k-1}) p(x_{k-1} | z_{k-1}) d x_{k-1} \\
    p(x_k | z_{k:k}) &= \frac{p(z_k, x_k) p(x_k | z_{k-1})}{\int p(z_k, x_k) p(x_k | z_{k-1}) d x_k}
\end{align*}
\]

The specific implementation steps are summarized as follows:

1) Initialization

\(N\) particles are sampled from the distribution \(p(x_0)\), and the weight of each particle is set to \(1/N\).

2) Sequential importance sampling

① Important density function:

\[
q(x'_k | x_{k-1}, z^k) = p(x'_k | x_{k-1})
\]

Extract \(N\) prediction particles from important density function \(\{x'_i, i = 1, 2, \cdots, N\}\).

② Calculate particle weight:

\[
\omega'_i = \omega'_i \frac{p(z_k | x'_i, Z^k) p(x'_i | x_{k-1})}{q(x'_i | x_{k-1}, Z^k)}
\]

③ Normalized weight

\[
\hat{\omega}_i = \frac{\omega'_i}{\sum_{i=1}^{N} \omega'_i}
\]

3) Resampling

① Calculate the effective particle capacity

\[
N_{\text{eff}} \approx \frac{1}{\sum_{i=1}^{N} (\hat{\omega}_i)^2}
\]

② If \(N_{\text{eff}} < N_{\text{threshold}}\), resample and map the weighted particles \(\{x'_{0:k}, \hat{\omega}'_k\}^N_{i=1}\) to equal weight particles \(\{x'_{0:k}, 1/N\}^N_{i=1}\).

4) State and variance estimates
It can be seen from the above process that from the moment \( k = 0 \), the particle filter system first initializes the samples. The system determines the prior probability of the target state and gives each particle the corresponding initial value. At the next moment, the system first carries out state transition, each particle propagates its own state according to the set state transition equation, then carries on the system observation, obtains the observation value, calculates the weight of all particles. Finally, particle weighting is used to get the output of posterior probability. At the same time, the samples are resampled and continue to transfer the system state, forming a loop tracking system.

The standard particle filter has the problems of particle weight degradation and particle diversity dilution, and selects the prior distribution as the importance density function, ignores the measurement information at the current time, resulting in low accuracy of state estimation, so this paper will improve the particle filter algorithm for these problems.

3. Firefly algorithm

Firefly algorithm (FA) is a kind of group intelligent bionic optimization algorithm constructed by imitating the group behavior of fireflies in nature, which has the advantages of few parameters, easy implementation and strong optimization ability [12]. The basic idea is to achieve the goal optimization based on the constant update of brightness and attractiveness, in which the different brightness of firefly individuals determines the direction of movement, and the attractiveness determine the distance of movement. Fireflies are attracted by fireflies that are brighter than themselves. The attraction of fireflies is directly proportional to the brightness, and the brightness decreases with the increase of distance. If there is no firefly that is brighter than itself, it will move randomly.

The optimization principle of firefly algorithm is described by mathematical formula as follows [13]:

1) The relative brightness of firefly \( i \) and firefly \( j \) is defined as

\[
I = I_0 \times e^{-r_{ij}^\gamma}
\]

Where: \( I_0 \) is the maximum brightness of firefly; \( \gamma \) is the absorption coefficient of light intensity; \( r_{ij} \) is the space distance between firefly \( i \) and firefly \( j \).

2) The attraction function of firefly is

\[
\beta = \beta_0 \times e^{-r_{ij}^\gamma}
\]

Where: \( \beta_0 \) is the maximum attraction of firefly.

3) The updated formula for the position of the firefly \( j \) attracted by the firefly \( i \) and moved to it is

\[
x_j = x_i + \beta \times (x_i - x_j) + \alpha \times (\text{rand} - 0.5)
\]

Where: \( x_i, x_j \) are the space position of the firefly \( i \) and the firefly \( j \) respectively; \( \beta \) is the attraction degree of the firefly; \( \alpha \in [0,1] \) is the step factor; rand is a random number of uniform distribution subject to the mean value of 0 and the variance of 1; \( \alpha \times (\text{rand} - 0.5) \) is interference terms, which is used to overcome the local extreme value phenomenon and avoid falling into premature convergence.

In the algorithm of firefly, the low brightness firefly moves around the high brightness firefly, but with the decrease of distance, the search ability of firefly is relatively reduced, and the phenomenon of
local extremum is easy to occur. Although the interference term $\alpha \times (\text{rand} - 0.5)$ is added to the above position update formula to alleviate the phenomenon of local extremum, the effect is not very ideal. Tian Mengchu and others use the global optimal value to replace the interaction information between particles [14]. Particles only need to compare with the global optimal value to reduce the complexity of operation. And in the firefly algorithm, the Firefly with a larger target function value has a higher corresponding brightness. Assuming that the maximum value of the target function is the global optimal value, the improved position update formula is

$$x'_k = x'_k + \beta \times (\text{gbest}_k - x'_k) + \alpha \times (\text{rand} - 0.5)$$

Where: $x'_k$ is the state value of the particle $i$ at the time $k$; $\beta$ is the attraction of firefly; $\alpha$ is the step factor; rand is a random number of $[0, 1]$; and gbest is the global optimal value. In formula, the global optimal value is used to guide the particle moving process, improve the global optimization ability of firefly optimization, and reduce the probability of local extreme value.

4. Particle filter based on Intelligent Optimization of firefly algorithm (FA-PF)

The traditional resampling method of particle filter avoids the phenomenon of particle shortage by deleting the set of small weight particles, but after many iterations, it will lead to the problem of particle dilution [15]. To solve the above problems, this study proposes the idea of using firefly algorithm to optimize particle filter. Firefly algorithm first randomly distributes the firefly population in the solution space, and each firefly individual is in the solution space according to equation, the updated position is calculated. After multiple moves, all individuals will gather at the position of the firefly with the highest brightness, so as to realize the final optimization.

Algorithm implementation and steps are as follows:

Step 1. At the initial time, $N$ particles are sampled as the initial particles of the algorithm $\{x'_0, i = 1, 2, \ldots, N\}$. The importance density function is expressed as

$$x'_k \sim q(x'_k | x'_{k-1}, z^t) = p(x'_k | z^t)$$

Step 2. Simulate the attraction and movement behavior of firefly optimization idea.

1) Calculate the attractivity between the particle $i$ and the global optimal value

$$\beta = \beta_0 \times e^{-r_i^2}$$

Where, $\beta_0$ is the maximum attraction of firefly, $\gamma$ is the absorption coefficient of light intensity, $r_i$ is the space distance between the particle $i$ and the global optimal value $\text{gbest}_k$.

2) Update particle position according to attraction

$$x'_k = x'_k + \beta \times (\text{gbest}_k - x'_k) + \alpha \times (\text{rand} - 0.5)$$

Step 3. Calculate and compare the fluorescent brightness value to update the global optimal value

$$\text{gbest}_k \in \{x'_1, x'_2, \ldots, x'_N | I(x)\}$$

$$= \max \{I(x'_1), I(x'_2), \ldots, I(x'_N)\}$$

Step 4. From the fluorescence calculation formula, it can be seen that the fluorescence value changes in the opposite direction with the difference between the predicted observation value and the real observation value. In this paper, the iteration termination threshold is set as 0.01. When the fluorescence function value is greater than 0.01, the algorithm stops iterating, otherwise it continues iterating to the maximum number of iterations. When the algorithm meets the set threshold, it means that the particles have been distributed near the real value. Or when the maximum number of iterations is reached, the optimization is stopped. Otherwise, go to step 2.

Step 5. Weight compensation and update. The core idea of the combination of firefly algorithm and PF is to perform the firefly iterative optimization operation on each particle in PF, so that the particle
moves along the direction of higher posterior density value of the target state, and improve the accuracy of state estimation. However, the firefly algorithm changes the position of each particle in the state space, at this time, the distribution density represented by the particle set is changed. The function \( p(x_k | z_{k-1}) \) is no longer, so the theoretical basis of Bayesian filtering will be lost. Therefore, this paper uses the idea of literature [16] for reference, and compensates and updates the weight at the same time of updating the particle position. The specific way is as follows:

\[
 w^*_i = \frac{p(x_k = s'_k | z_{k-1})}{q(s'_k)} p(z_k | x_k = s'_k)
\]

Where, \( s'_k \) is the particle \( i \) at time \( k \), \( q(*) \) is the importance function. After the weight compensation, the particle set before and after the optimization at least obeys the same distribution in theory, \( p(x_k | y_{1:k-1}) \). It keeps the theoretical basis of Bayesian filtering.

Step 6. Normalize

\[
 \hat{\omega}^*_i = \omega^*_i \sqrt{\sum_{i=1}^{N} \omega^*_i}
\]

Step 7. State output

\[
 \hat{x}_k = \sum_{i=1}^{N} \hat{\omega}^*_i x^*_k
\]

The above idea makes full use of the effective information of the whole particle set, which is conducive to the particle jumping out of the local extremum, reducing the number of iterations wasted in the situation where the state value change is not obvious, making the improved algorithm stop optimization more because it reaches the threshold value set initially, reducing the probability that the algorithm stops iterating to the maximum number of iterations, thus further improving the operation speed. In terms of the number of effective particle samples, the above methods can increase the diversity of particles and improve the quality of particle samples.

Due to the strong convergence ability of firefly algorithm, we set the maximum number of iterations and the termination threshold in FA-PF to limit the number of iterations of the algorithm, making the particle swarm move to the real area as a whole, but avoiding the final convergence. The specific reasons are as follows:

1) The core idea of FA-PF is to make the particle set have a more reasonable coverage of the high likelihood area, which should be wide. If all particles are concentrated near the real value, it will reduce the diversity of particles.

2) If the number of iterations is too many, the operation complexity of FA-PF will be significantly higher than that of PF, which will reduce the real-time performance of FA-PF.

5. Simulation experiment

The representative single variable non-static growth model in the literature is selected as the experimental simulation model, and its process model and measurement model are as follows:

**Process model:**

\[
 x_k = 0.5x_{k-1} + \frac{25x_{k-1}}{1 + x_{k-1}^2} + 8 \cos \left[ 1.2(k-1) \right] + w_k
\]

**Measurement model:**

\[
 z_k = \frac{x_k^2}{20} + v_k
\]
Where, $w_k \sim N(0,1)$, $v_k \sim N(0,1)$ are the Gaussian white noise with zero mean value. In the firefly algorithm, the parameters are usually set according to the existing literature, the maximum attraction is 1, the step factor is 0.3, and the maximum light intensity absorption coefficient is 1.

The evaluation standard of filtering accuracy adopts root mean square error, and its calculation formula is

$$RMSE = \left[ \frac{1}{M} \sum_{t=1}^{M} (x_t^c - \hat{x}_t^c)^2 \right]^{1/2}$$

Where: $M$ is the number of Monte Carlo simulations, set to 100.

When $N = 100$, the simulation results of state estimation and estimation error of the model are shown in Figure 1 and Figure 2.

![Figure 1. State estimation of filter (N=100).](image1.png)

![Figure 2. Absolute value of filter error (N=100).](image2.png)

It can be seen from Figure 1 and Figure 2 that the FA-PF algorithm has higher filtering accuracy and smaller estimation error than the ordinary PF algorithm. It can more accurately express the state of real particles, which further shows that the filtering effect of the improved algorithm is improved.
6. Conclusion
In this paper, an improved particle filter algorithm for firefly optimization is proposed, which can be used to estimate the location of shipboard aircraft and improve the location accuracy. The lack of particle weight and diversity in traditional particle filter algorithm seriously affects the performance and accuracy of filter. In this paper, based on the global optimal value instead of the interactive information between fireflies, the improvement of firefly algorithm is realized, and the improved firefly algorithm is used to optimize the sampling process of particle filter. At the same time, the incomplete resampling method is used to alleviate particle degradation and better protect the performance the experimental results show that the algorithm proposed in this paper improves the filtering accuracy compared with PF algorithm, and effectively improves the lack of particle diversity.

References
[1] Zhu Zhiyu. Particle filter algorithm and its application [M]. Beijing: Science Press, 2010.
[2] Wang Fasheng, Lu Yuming, Zhao Qingjie, et al. Particle filter algorithm [J]. Journal of Computer Science, 2014, 37 (8): 1679-1693.
[3] Li Juan, Liu Xiaolong, Lu Changgang, et al. Improved resampling algorithm of particle filter [J]. Journal of Jilin University (Engineering Edition), 2015, 45 (6): 2069-2074.
[4] YANG X S. Firefly algorithm stochastic test functions and design optimization [J]. International Journal of Bio-Inspired Computation, 2010, 2 (2): 78-84.
[5] Cheng Meiyin, Ni Zhiwei, Zhu Xuhui. A review of optimization algorithm theory of firefly [J]. Computer science, 2015, 42 (4): 19-24.
[6] Yu Shuhao. Research and application of firefly optimization algorithm [D]. Hefei: Hefei University of Technology, 2015.
[7] FARAHANI S M, ABSHOURI A A, NASIRI B, et al. A Gaussian firefly algorithm [J]. International Journal of Machine Learning & Computing, 2011, 1 (5): 448-453.
[8] Liu Changming, Ye Chunming. A novel intelligent optimization algorithm of bionic group: Firefly algorithm [J]. Computer application research, 2011, 28 (9): 3295-3297.
[9] Tian Mengchu, Bo Yuming, ChenZhimin, et al. Intelligent optimization of particle filter based on firefly algorithm [J]. Acta automatica Sinica, 2016, 42 (1): 89-97.
[10] Wang Ershen, Pang Tao, Qu Pingping, et al. Improved particle swarm optimization particle filter algorithm based on Chaos [J]. Journal of Beijing University of Aeronautics and Astronautics, 2016 (5): 885-890.
[11] MAROULAS V, STINIS P. Improved particle filters for multi-target tracking [J]. Journal of Computational Physics, 2012, 231 (2): 602-611.
[12] Xing Jiali, Yang Yifei. Application of particle filter algorithm based on particle swarm optimization [J]. Science Technology and Engineering, 2012, 64 (4): 936-939.
[13] Zhang Min, Jia Haitao, Shen Zhen. Improved resampling procedure based on genetic algorithm in particle filter [J]. Journal of University of Electronic Science and Technology of China, 2015, 44 (3): 344-349.
[14] ChenZhimin, Bo Yuming, Wu Panlong, et al. Application of new particle filter based on Adaptive Particle Swarm Optimization in target tracking [J]. Control and Decision, 2013, 28 (2): 193-200.
[15] Du Taihang, Li Jingqiu, Jiang Chundong. Improved firefly algorithm to optimize the signal source location of particle filter [J]. China test, 2017, 43 (11): 96-101.
[16] Zhu Chao, Liu Yi-an, Xue Song. Research on improved particle filter algorithm of firefly based on Chaos [J]. Transducer and Microsystem Technologies, 2017, 36 (9): 106-109.