Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 187 (2021) 206–211

Procedia Computer Science

www.elsevier.com/locate/procedia

International Conference on Identification, Information and Knowledge in the internet of Things, 2020

Particle filter algorithm optimized by genetic algorithm combined with particle swarm optimization

Jin Yang[a], Xuerong Cui[b,*] , Juan Li[a,*], Shibao Li[b], Jianhang Liu[a], Haihua Chen[a]

[a]*College of computer science and technology, China University of Petroleum(East China), Qingdao 266580, China*
[b]*College of Oceanography and Space Informatics, China University of Petroleum(East China), Qingdao 266580, China*

Abstract

The standard particle filter (PF) algorithm has the issue of particle diversity loss caused by particle degradation and resampling, which makes it impossible for particle samples to accurately represent the true distribution of state probability density function. Particle swarm optimization (PSO) algorithm can effectively improve the particle degradation problem of particle filter namely, PSO-PF, but its fitness function is greatly affected by the variance of measurement noise, and is easy to fall into local optimal, which greatly limits the filtering accuracy. Therefore, this paper proposes an algorithm that combines genetic algorithm (GA) and PSO algorithm to improve particle filtering, namely, GA-PSO-PF. This algorithm combines the fast convergence speed of particle swarm optimization with the strong global searching ability of genetic algorithm to increase the diversity of particles while ensuring the effectiveness of superior particles, and improve the speed and accuracy of finding the optimal solution. Experimental results show that the filtering performance of the proposed algorithm is better than PF and PSO-PF, and the positioning and tracking accuracy is improved by 54.44% compared with PF and 27.20% compared with PSO-PF.

Keywords: particle filter algorithm; particle swarm optimization; genetic algorithm; target tracking and location

1. Introduction

Particle filtering (PF), also known as the Sequential Monte Carlo method, the core idea of which is to approximate

* Corresponding author.
E-mail address: cuixuerong@163.com；lijuanlijuan@sina.com

10.1016/j.procs.2021.04.052

the probability density function of the system with some discrete random sampling points to obtain the minimum variance estimation of the state[1]. Not limited by the linear system model and Gaussian distribution hypothesis, it has been widely used in computer vision [2], image processing [3] and radar target tracking [4] fields. The schematic diagram of the algorithm is shown in Figure 1. However, the standard particle filter algorithm has the problem of particle degradation [5], some researchers have proposed a series of improvement methods. Rudolph et al. [6] introduced the Unscented Kalman Filter (UKF) method into Particle filtering and proposed the Unscented Particle Filter (UPF). Many researchers have also used some intelligent algorithms to optimize the particle filter algorithm[7][8][9].

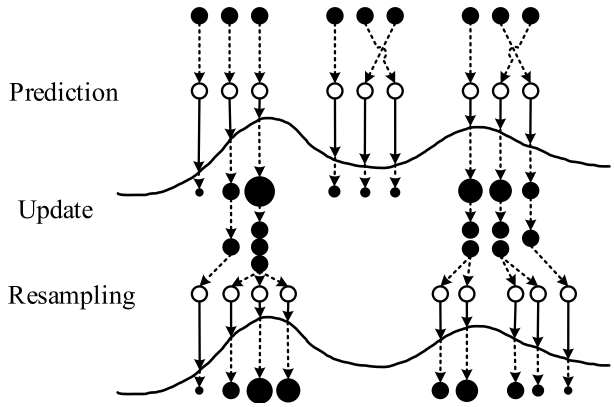

Fig. 1. Schematic diagram of particle filter algorithm.

Particle swarm optimization (PSO) is an intelligent optimization algorithm, the idea of which comes from the study on the predation behavior of birds [10]. It optimizes particle group through cooperation and competition between particles, so the movement of the whole group can evolve from disorder to order in the solution space of the problem, then the optimal solution of the problem can be obtained [11][12]. Therefore, in order to improve the performance of particle filtering, PSO is introduced in this paper to optimize the resampling particle set of PF (PSO-PF). However, this algorithm takes the difference between the current measured value and the predicted value as the evaluation criterion. When the noise variance is high, it directly affects the selection of the optimal particle, resulting in the serious deterioration of the filtering performance [13]. To solve the problem of particle degradation, in this paper, an adaptive genetic algorithm combined with particle swarm optimization is proposed to improve particle filtering (GA-PSO-PF).

2. Standard particle filter

Particle filtering uses weighted discrete particles to represent state probability distribution. It is a non-parametric approximation technique (density estimation) designed to solve nonlinear filtering problem and is applicable to any nonlinear system that can be described by state space model [14].

In target tracking problems, the state space model of the system can be described as:

$$x_k = f(x_{k-1}, u_k), \tag{1}$$

$$z_k = h(x_k, v_k), \tag{2}$$

where *f(.)* and *h(.)* are the state transfer equation and observation equation respectively. x_k is the system state at time k, z_k is the observed value, u_k is the process noise, and v_k is the observation noise.

The steps of the standard particle filter algorithm are as follows:

(1) Initialization: to generate particle set x_0^i(i=1,2,…,N) from prior distribution, and the weight of each particle is 1/N.

(2) Importance sampling: to predict the states of all particles, calculate the observed estimates of particles,

$$\hat{x}_{k+1} = f(x_k, u_{k+1}), \tag{3}$$

$$\hat{z}_k = h(\hat{x}_k, v_k), \tag{4}$$

update the weights of particles:

$$w_k^i = w_{k-1}^i \frac{p\left(z_k|x_k^i\right)p(x_k^i|x_{k-1}^i)}{q(x_k^i|x_{k-1}^i,z_k)}, \quad (5)$$

and normalize the weights:

$$\widetilde{w}_k^i = \frac{w_k^i}{\sum_{i=1}^N w_k^i}. \quad (6)$$

(3) Resampling: to copy or discard the sample according to w_k^i in order to get a new particle set, with the total number of particles remaining unchanged after resampling and w_k^i equalling $1/N$, $i = 1, 2,...,N$.

(4) State estimation: to calculate the target state estimation based on the weights and states of all particles,

$$\tilde{x}_k = \sum_{i=1}^N \hat{x}_k^i \cdot \widetilde{w}_k^i. \quad (7)$$

(5) Return to step (2) and continue filtering until the filtering is finished.

Through the above steps, it is not difficult to find that the resampling process of particle filter will reduce the diversity of particles, leading a large error in target state estimation.

3. PSO-PF

Particle swarm optimization algorithm can be expressed as: randomly initializing a particle swarm with population size m in n-dimensional space, where the position of the i^{th} particle in n-dimensional space is expressed as $X_i = (x_{i1}, x_{i2},\ldots,x_{in})$, velocity $V_i = (v_{i1}, v_{i2},\ldots,v_{in})$. For each iteration, the velocity and position of particles are updated according to Equations (8) and (9) by referring to the individual best $Pbest_i = (p_{i1}, p_{i2}..., p_{i2})$ and the global best $Gbest = (g_1, g_2,...,g_n)$.

$$v_{id}^{k+1} = w \times v_{id}^k + c_1 \times r_1 \times \left(p_{id} - x_{id}^k\right) + c_2 \times r_2 \times \left(p_{gd} - x_{id}^k\right), \quad (8)$$

$$x_{id}^{k+1} = x_{id}^k + v_{id}^{k+1}, \quad (9)$$

where, v_{id}^k is the d^{th} component of velocity v_i of the i^{th} particle in the k^{th} iteration, i =1,2..., N, w is inertia coefficient, c_1 and c_2 are acceleration coefficients, r_1 and r_2 are random numbers obeying U(0,1) distribution.

In order to overcome the shortcomings of particle filtering, in literature [15], particle swarm optimization was introduced into particle filtering to improve the sampling process, reduce particle degradation, and improve the accuracy of state estimation and the convergence of the algorithm. PSO-PF algorithm has the following steps:

(1) Introduce the latest observations into the sampling process and calculate the fitness of each particle,

$$fitness = \exp\left[-\frac{1}{2R_k}\left(z_k - \hat{z}_{k|k-1}^i\right)^2\right], \quad (10)$$

among them, z_k is the latest observation value, $\hat{z}_{k|k-1}^i$ is the predicted observation value, R_k is the variance of observed noise.

(2) Take the prior probability of particles as the important density function and initialize the particle swarm,

$$x_k \sim q\left(x_k^i\middle|x_{k-1}^i, z_k\right) = p(x_k|x_{k-1}). \quad (11)$$

(3) Calculate the weights of particles and normalize them using Equation (6),

$$w_k^i = w_{k-1}^i p(z_k|x_{k-1}^i), \quad (12)$$

update the velocity and position of each particle respectively using Equations (8) and (9).

(4) Resample the particles.

(5) State estimate and output the result.

Although PSO-PF algorithm improves the particle degradation problem to some extent, however, the basic particle swarm optimization algorithm itself has poor global searching ability. Such as, it is easy to fall into local optimal, is easy to scatter, and is greatly affected by measurement noise, which will affect the filtering accuracy. Therefore, in view of the inherent shortcomings of PSO, our previous genetic algorithm [16] is introduced to improve the mutation ability of particle swarm optimization, so as to improve the global optimization ability of the algorithm and the experimental accuracy.

4. GA-PSO-PF

In this paper, adaptive genetic algorithm and particle swarm optimization algorithm are combined to improve the particle degradation of particle filter, enhance the diversity of particles, and avoid the algorithm falling into local optimal. The steps of GA-PSO are as follows:

(1) Initialize the population and set the relevant parameters of the population.
(2) Calculate the fitness of each particle and record the optimal position.
(3) Update the velocity and position of each particle.
(4) The roulette algorithm [17] is adopted to select individuals for crossover and mutation operations of the genetic algorithm. If the evolution of a better individual is achieved, the $Pbest_i$ or the *Gbest* will be updated.
(5) Determine whether the maximum number of iterations is reached. If the conditions are met, the algorithm will stop, if not, step 4 will continue.

The flow chart of GA-PSO-PF algorithm is shown in Figure 2.

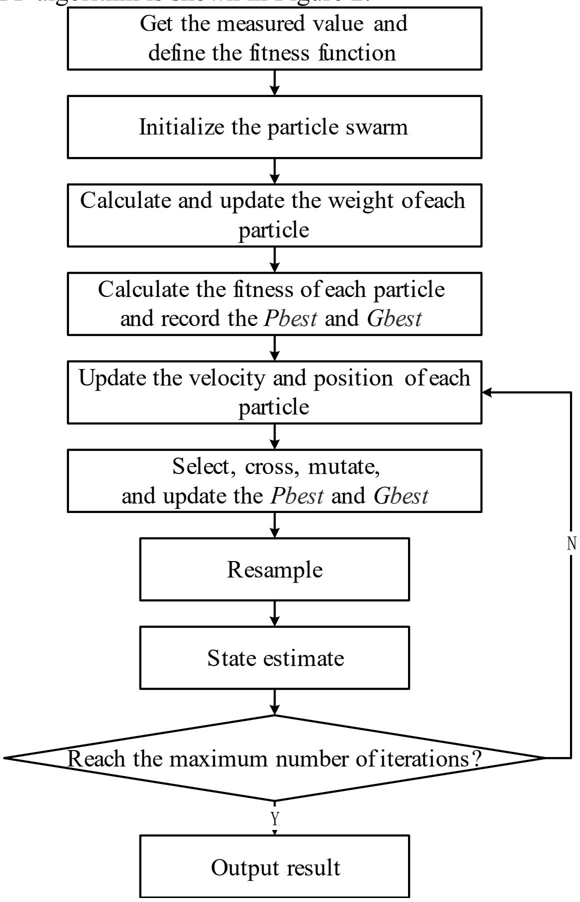

Fig. 2. Flow chart of GA-PSO-PF algorithm

5. Algorithm verification and analysis

5.1. Experimental simulation environment

In the two-dimensional space, the initial position, state equation and sensor measurement data of the car are set. The car is moved according to the motion equation which is superimposed with the control noise, and the particle filter method is used to locate the object.

5.2. Environmental parameter

(1) Total number of particles N=200;
(2) Movement time T=15 seconds, and assume one action per second;
(3) The size of the sports field is 100×100 m^2;

(4) Set the motion equation of the car, let the car move along a curve (including noise), and set the noise as 5 m;
(5) The initial position of the car is set at the position of (50,20) in the 100×100 m^2;

In the entire 100×100 m^2 simulation environment, 200 particles are uniformly distributed, and the geometric center of the particle set in the initial state is shown in Figure 3. The car moves according to the given motion equation superimposed with noise. After the car moves to the next position, the actual position of the car, the positions of the particles after resampling, and the geometric center position of the particles with PF, PSO-PF, GA-PSO-PF algorithms are shown in Figure 4, and their trajectories are shown in Figure 5. It can be found that the tracking performance of particle filter improved by PSO algorithm is better and closer to the real trajectory, while GA-PSO-PF algorithm is better than PSO-PF algorithm.

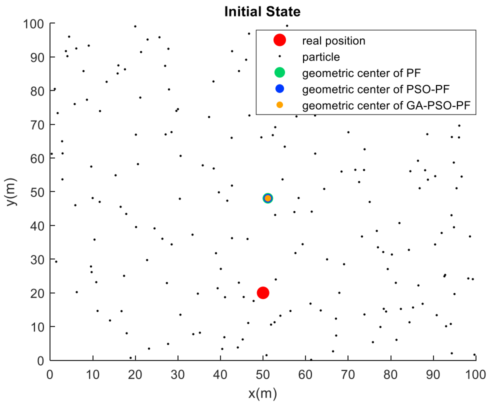

Fig. 3. Initial state

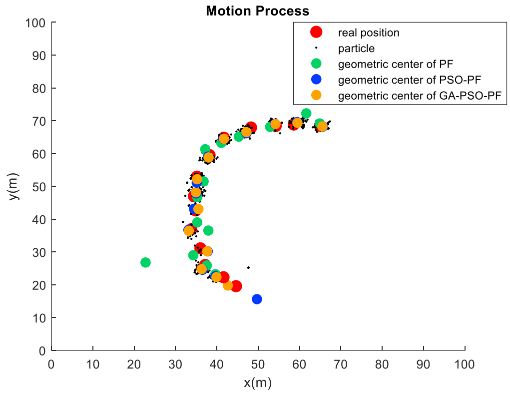

Fig. 4. Movement process

Figure 6 shows the error curves of these algorithms. It is found that the tracking performance of particle filter improved by PSO is better, while GA-PSO-PF algorithm converges faster and is more stable than PSO-PF.

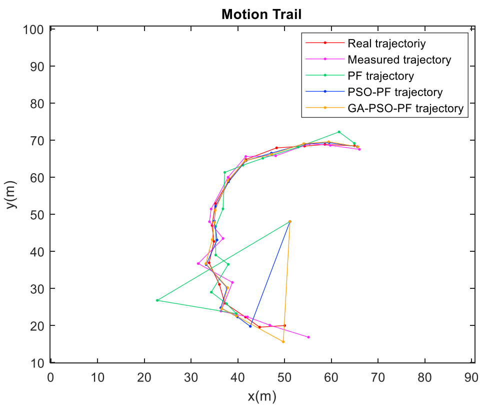

Fig. 5. Trajectories of each algorithm

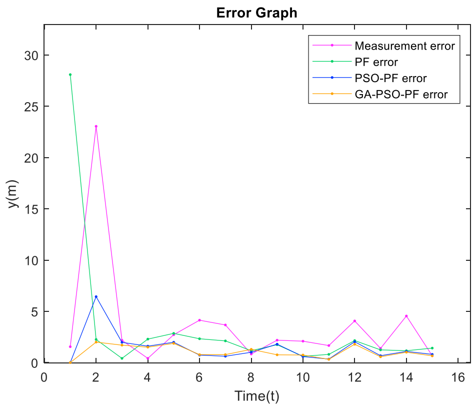

Fig. 6. Error curve of algorithms

Table 1. Comparison of mean square error of different algorithms.

Algorithms	Mean Square Error (m)	Variance (m)
PF	3.376	44.1031
PSO-PF	1.4476	2.1657
GA-PSO-PF	1.0538	0.3453

Table 1 shows the mean square error (MSE) and variance statistics of each algorithm after 15 independent experiments. It can be seen from the table that the PF algorithm has the highest MSE, followed by PSO-PF algorithm, while GA-PSO-PF algorithm is the lowest. The algorithm proposed in this paper has the smallest MSE, and the positioning and tracking accuracy is 54.44% higher than PF and 27.20% higher than PSO-PF. Meanwhile, the proposed algorithm has the smallest variance and is more stable.

6. Conclusion

In this paper, an improved particle filter tracking algorithm based on intelligent algorithm is proposed to solve the problem of particle degradation and scarcity. This algorithm combines the fast convergence speed of PSO with the strong global search ability of GA to optimize the particle set before resampling, increases the diversity of particles while ensuring the effectiveness of superior particles, and improves the speed and accuracy when finding the optimal solution. The simulation results show that compared with the traditional particle filter algorithm and the particle filter algorithm improved by particle swarm optimization. The GA-PSO-PF algorithm can better track the target and reduce the offset. Experiments show that the algorithm proposed in this paper is better than PF and PSO-PF in stability and accuracy and has stronger anti-interference and higher tracking accuracy which is 54.44% higher than PF and 27.20% higher than PSO-PF. In our future work, we will consider the fusion of Wi-Fi location and smartphone sensor location using the GA-PSO-PF algorithm. In addition, we will consider a combination of particle filtering and reinforcement learning methods to provide robustness for location failure.

Acknowledgements

This work was supported by National Natural Science Foundation of China (No. 61671482, 61902431, 91938204 and 61972417), the Key Research and Development Program of Shandong Province with No. 2019GGX101048 an d Fundamental Research Funds for the Central Universities (No.19CX05003A-9, 19CX05003A-4 and 18CX02136A).

References

[1] Shiqiang H and Zhongliang J. (2010) “Principle and application of particle filter”, *Science Press*.

[2] Sherrah J, Ristic B, and Redding N J. (2011) “Particle filter to track multiple people for visual surveillance”, *IET Computer Vision* **5** (**4**): 192-200.

[3] Varcheie P D Z, Bilodeau G A. (2011) “Adaptive fuzzy particle filter tracker for a PTZ camera in IP surveillance system”, *IEEE Transaction Instrumentation and Measurement*, **60** (**2**) :354-371.

[4] Wang X Z, Musicki D, Ellem R, et al. (2009) “Efficient and enhanced multi-target tracking with doppler measurements”, IEEE *Transaction Aerospace and Electronic Systems*, **45** (**4**) :1400-1417.

[5] Xiaoping H, Yan W and Pengcheng M. (2017) “Principle and application of particle filter”, *Publishing House of Electronics Industry*.

[6] Van M R and Doucet A. (2000) “The unscented particle filter”, *Cambridge: Cambridge University*.

[7] Zhimin C, Panlong W, Yuming B et al. (2018) “Adaptive Control Bat Algorithm Intelligent Optimization Particle Filter for Maneuvering Target Tracking”, *Chinese Journal of Electronics*, **46** (**4**): 886-894.

[8] Mengchu T, Yuming B, Zhimin C et al. (2016) “Firefly Algorithm Intelligence Optimized Particle Filter”, *ACTA AUTOMATICA SINICA*, **42** (**1**): 89-97.

[9] Jianchun Z, Fengju K, Hongtao L, Hao X. (2017) “Research on Chicken Swarm Optimization-based Particle Filter”, *Journal of System Simulation*, **29** (**2**): 295-300+308.

[10] Kennedy J, Eberhart R C. (1995) “Particle swam optimization”, *Proceedings of the 1995 IEEE International Conference on Neural Networks*, **4**: 1942－1948.

[11] Qingjian N, Hancheng X, Zhizheng Z et al. (2007) “Survey of Particle Swarm Optimization Algorithm”, *Pattern Recognition and Artificial Intelligence*, **20** (**3**): 349-357.

[12] Zhang G Y, Cheng Y M, Yang F et al. (2008) “Particle filter based on PSO”, *International Conference on Intelligent Computation Technology & Automation*. IEEE Computer Society.

[13] Chongyang L, Shouyong W, Yang W et al. (2013) “An improved particle filter algorithm based on particle swarm optimization”, *Journal of Air Force Early Warning Academy*, **27** (**3**):209-212+216.

[14] Kun H, He Z. (2018) “Improved Particle Filter Based on Fruit Fly Optimization and Its Application in Target Tracking”, *Journal of Hunan University (Natural Sciences)*, **45** (**10**):130-138.

[15] Zheng F, Guofeng T, Xinhe X. (2007) “Particle swarm optimized particle filter”, *Control and Decision*, **22** (**3**): 273-277.

[16] Xuerong C, Jin Y, Juan Li, Chunlei W. (2020). “Improved Genetic Algorithm to Optimize the Wi-Fi Indoor Positioning Based on Artificial Neural Network”, *IEEE Access*.

[17] Yonglin L, Chunming Y, Changping L. (2014). “Roulette selection self adaptive harmony search algorithm”, *Application Research of Computers*, **31** (**6**): 1665-1668.