Improved Resampling Procedure Based on Genetic Algorithm in Particle Filter

ZHANG Min[1], JIA Hai-tao[2], and SHEN Zhen[2]

(1. Library, University of Electronic Science and Technology of China　Chengdu　611731;

2. Research Institute of Electronic Science and Technology, University of Electronic Science and Technology of China　Chengdu　611731)

Abstract　Particle filtering is a nonlinear and non-Gaussian dynamical filtering system. It has found widespread applications in detection, navigation, and tracking problems. The strong maneuverability of target tracking brings heavy impact on particle attributes in resampling process of particle filters, such as, particle state, particle weights, and so on. This paper proposes a new particle filter algorithm based on genetic algorithm optimization. This algorithm combines the hereditability and aberrance of the genetic algorithm into the resampling procedure of particle filter to improve the adaptability of maneuvering target tracking.

Key words　genetic algorithm;　maneuvering target tracking;　nonlinear filtering;　particle filter;　particle resampling

基于遗传算法改进的粒子滤波重采样模型

张　民[1]，贾海涛[2]，沈　震[2]

(1. 电子科技大学图书馆　成都　611731；2. 电子科技大学电子科学技术研究院　成都　611731)

【摘要】提出一种基于遗传算法改进的新粒子滤波算法，该算法对于每次迭代计算出的最差粒子并未简单地进行丢弃，而是将这些最差粒子利用生物遗传中的遗传性和变异性将其进行修正。该算法利用最差粒子数据与种群中特殊数据进行交叉变异方法来增强粒子种群中的多样性，从而有利于粒子滤波对机动目标的跟踪；同时保留部分粒子在未来进行唤醒也体现了多样性。该算法更有利于实现粒子滤波在机动目标跟踪的适应性，提高其跟踪效果。

关　键　词　遗传算法；　机动目标跟踪；　非线性滤波器；　粒子滤波；　重采样

中图分类号　TN953　　　　　文献标志码　A　　　doi:10.3969/j.issn.1001-0548.2015.03.005

Target tracking is used in many application areas, such as the defense system, radar system, sonar system, aeronautical system, satellite system, and autonomous robots[1]. For anyone target tracking system, it must solve two basic tasks. One is the accurate estimation procedure, which needs to infer the accurate estimation of target position from noise measurement data. The other is predicting the target position in the next time that can be used to control the tracker to catch the target [2]. So the kernel problem of target tracking is to estimate the states of the moving target, such as position, velocity, and acceleration. Most target tracking algorithms belong to Bayesian theoretic, such as Kalman filter and particle filter, which they are popular Bayesian filters for target tracking because of their probabilistic nature[3].

The particle filter algorithm was first proposed in Ref.[4]. In recent years, the particle filter (PF), as an effective estimator for the nonlinear filtering problem, has been widely used in many fields, including signal processing, biostatistics, economics, and engineering.

The kernel of particle filters is designing the posterior probability density functions based on a sample (or particle). This allows the filter to handle the nonlinearity of system, as well as the non-Gaussian nature of noise processes.

PF can sufficiently estimate the system states when the number of particles (i.e., estimations of the

state vectors that evolve in parallel) is large. In Ref.[5], a particle filtering algorithm for tracking was introduced, which focuses on geometric properties of the sensor network configuration, and the algorithm was derived from geometry. In Ref.[6], a tracking method was proposed that first estimated the positions of a target in its most recent past and then fit them with a piece-wise trajectory. In Ref.[7], another method for distributed tracking in binary sensor networks was developed, which was derived by using hidden state estimation and the Viterbi algorithm.

Up to today, particle is one of the most success non-liner filters. But there are some optimization researches in maneuvering target tracking. During tracking strong maneuvering targets, continuous or strong motor will take heavy impacts on the particle filter re-sampling process. Concisely to say, this strong mobility or continuous motor will make the particle weights that can correctly estimate target state turn smaller, or even be abandoned, and the particle weights which have not contribution even be increased infinitely, and this group of particles cannot correctly estimate the target state. So this paper proposes a new particle filter algorithm based on genetic algorithm optimization. This algorithm takes the descendiblity and aberrance of the genetic algorithm into particle filter resampling procedure, which seems to more adapt the maneuvering target tracking.

1　Particle Filter Theory

Considering a single target tracking problem, x_k is the target motion state vector at time k:

$$\begin{cases} x_k = f(x_{k-1}) + v_k \\ z_k = h(x_{k-1}) + \omega_k \end{cases} \quad (1)$$

where $v_k \sim N(\mu, \Sigma)$ and $N(\mu, \Sigma)$ are noise sequence which have the Gaussian distribution with mean μ and covariance matrix Σ, ω_k is observation noise sequence and is independent of v_k, $f(\cdot)$ is nonlinear state function, z_k is observation vector obtained at time k, and $h(\cdot)$ is nonlinear measurement function. Let $X_{0:k} = \{x_0, x_1, \cdots, x_k\}$ and $Z_{0:k} = \{z_0, z_1, \cdots, z_k\}$ denote the vectors of the motion states and the observation states of target from beginning to time k.

In the following discussion, the notation $p(\cdot)$ is used to denote a probability density function (PDF), for example $x_n^{(i)} \sim p(x_n)$ indicates that the particles are distributed according to the pdf of the true state[9].

Using statistics theoretics, the system is completely described as follows [10]:

1) $p(x_n | x_{n-1})$. The state transition probability density function. It desribes the evolution of the system from time $n-1$ to n. Alternatively, the same could be described with a state transition model of the form $x_n = \Phi(x_{n-1}, v_n)$, where v_n is a noise process.

2) $p(z_n | x_n)$. Observation likelihood density, describing the conditional likelihood of observation given state. As before, this relationship could be in the form of an observation model $z_n = h(x_n) + n_n$, where n_n is a noise process which is independent of v_n.

3) $p(x_n)$. The prior state probability at t time.

It is assumed that the $X_{0:n} = \{x_0, x_1, \cdots, x_n\}$ is a homogeneous Markov chain, the conditional probability density of x_n given by the past states $x_{0:n-1} = (x_0, x_1, \cdots, x_{n-1})$ only depends on x_{n-1}, through the transition density $p(x_n | x_{n-1})$, and the conditional probability density of z_n given by the states $x_{0:t}$ and the past observations $z_{0:n-1}$ only depends on x_t through the conditional likelihood $p(z_n | x_n)$[11].

The objective of filtering is to estimate the posterior density of the state given by the past observations $p(x_n | z_{1:n})$. As new observations arrive, a recursive update of the posterior density is given by the recursive Bayesian filter. It is defined as[12]:

$$p(x_n | z_{1:n-1}) = \int p(x_n | x_{n-1}) p(x_{n-1} | z_{1:n-1}) \mathrm{d}x_{n-1} \quad (2)$$

$$p(x_n | z_{1:n}) = \frac{p(z_n | x_n) p(x_n | z_{1:n-1})}{p(z_n | z_{1:n-1})} \quad (3)$$

In most applications, the posterior density of the state vector $p(x_n | z_{1:n})$ is interested. In particle filtering, densities are approximated by a set of samples (particles) $\{x_n^{(i)}\}_{i=1}^{N_p}$. In the state space, their associated normalized probability weights satisfy $\sum_{i=1}^{N_p} \{W_n^{(i)}\} = 1$. Then the posterior density of the state vector can be approximated as[13]:

$$p(x_n | z_{1:n}) \approx \sum_{i=1}^{N_p} w_k^i \delta(x_n - x_n^i) \quad (4)$$

where $\delta(\cdot)$ is the Dirac Delta function centered at x_n. The set $W_n^{(i)}$ is the weights of particle set that represent

the posterior density at time n, and is estimated recursively from $W_{n-1}^{(i)}$. The initial particle set $W_0^{(i)}$ is obtained from sampling the prior density $\pi_0 = p(x_0)$ [14].

In general, it is difficult to sample directly from the full posterior density. To overcome this difficulty, the method of importance sampling is used. The particles $x_n^{(i)}$ are drawn from an easy sampling function $q(\cdot)$ called importance density. So the normalized weights is written as[15]:

$$w_t^i = \frac{p(x_{0:t}^i \mid z_{0:t})}{q(x_{0:t}^i \mid z_{0:t})} \qquad (5)$$

The importance density is factorized as follows[16]:

$$q(x_{0:t} \mid z_{0:t}) = q(x_t \mid x_{t-1}, z_t) q(x_{0:t-1} \mid y_{1:t-1}) \qquad (6)$$

So the weights can be updated sequentially as[17]:

$$w_t^i = \frac{p(z_t \mid x_t^i) p(x_t^i \mid x_{t-1}^i)}{q(x_t^i \mid x_{t-1}^i, z_t)} \qquad (7)$$

One of the most common particle-filtering algorithms is the sampling importance resampling (SIR) filter. It updates the sample sets that represent the posterior about the map and the trajectory of the vehicle. The process can be summarized by the following four steps [18].

1) Sampling: The next generation of particles $\{x_n^{(i)}\}$ is obtained from the generation $\{x_{n-1}^{(i)}\}$ by sampling from the proposal distribution π. Usually, a probabilistic motion model is used as the proposal distribution.

2) Importance weighting: According to the importance sampling principle, an individual importance weight $w_n^{(i)}$ is assigned to each particle. The weights account for the fact that the proposal distribution π is, in general, not equal to the target distribution of successor states.

3) Resampling: Particles are drawn with the replacement proportional to their importance weight. This step is necessary, since only a finite number of particles are used to approximate a continuous distribution. Furthermore, resampling allows us to apply a particle filter in situations in which the target distribution differs from the proposal. After resampling, all the particles have the same weight[19].

4) Map estimation: For each particle, based on the trajectory $x_{0:n}^{(i)}$ of that sample and the history of observations $z_{0:n}$, the corresponding map estimation $p(x_n \mid z_{1:n})$ is computed [20].

2 Improved Algorithm

Without the resampling step, the basic particle filter would suffer from the sample depletion. This means that all particles not a few will have negligible weights after a while. The resampling step resolves the reduction of the effects of degeneracy. The basic idea of resampling is to drop particles that have small weights and to concentrate on those which have large weights. A new set of samples is generated by resampling the set of samples and taking out the particles that have small weights. N_r samples from the current set, proportionally to their weights. In this new set, for instance, the samples with the lowest probabilities will disappear. Next, the weights associated with the samples are scaled in order to represent the probability associated with each sample. In fact the resulting set of samples is an independency density sample from the discrete posterior probability function $p(x_t \mid z_{0:t})$. Therefore, the weights can now be reset as $w_t^i = \dfrac{1}{N_r}$. Fig. 1 shows the resampling procedure.
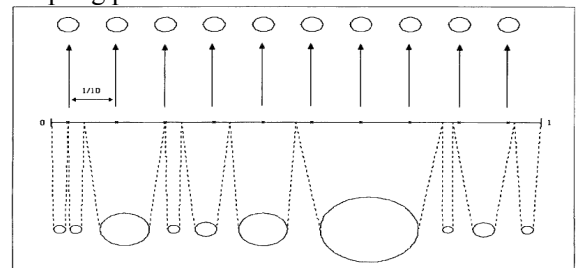

Fig. 1　The resampling procedure

The common resampling procedure would duplicate the old ones that have high weights, which might lead to a loss of diversity (named sample impoverishment). It is very severe and in a poor way that all particles may collapse at a single point with a few iterations if the process noise is very small. Especially there is some mutation in target state when it maneuvers. Without considering this mutation, the particles will not represent the posterior density of target state, and would generate degeneracy of particles.

Aiming at the resampling step, this paper proposes an improved algorithm whose main idea is generating

some aberrance particles in the resampling procedure. Once the target state has been maneuvered, some aberrance particles would work well and have high weights, which will increase the diversity and make the particle filter suit to maneuvering target tracking. Figure 2 shows the algorithm flow.

The improved algorithm detail step is:

1) Sorting particles into three types: the normal particle, aberrance particle, and best particle. The best particles would maintain its state in the resampling procedure, while the aberrance particle would randomly alter its state.

2) Evaluating the maneuvering. Here a maneuvering parameter is used to evaluate the target maneuvering. The function of maneuvering parameter is:

$$\beta_m = \frac{\sum\limits_{i=1}^{N}(w_{n-1}^i - \overline{w}_{n-1})^2}{\sum\limits_{i=1}^{N}(w_n^i - \overline{w}_n)^2} \qquad (8)$$

where β_m is maneuvering parameter and $\overline{w}_{n-1}$ is the mean value of $n-1$ iterative loop weights.

3) If the maneuvering parameter is beyond the evaluation threshold, there might have some target maneuvering. The resampling procedure would augment the weights of the aberrance particles. Whereas, resampling procedure would debase the weights of the aberrance particles.

4) Recording the best particle's state in every iterative loop. The resampling procedure would eliminate the best particle which is no longer the best in three iterative loops.

5) If the maneuvering parameter is beyond the evaluation threshold, the resampling procedure would revive the best particle record in three iterative loops. The main idea of this improved algorithm is taking the aberrance and descendiblity of the genetic algorithm to increase the diversity of the resampling procedure in order to adapt to maneuvering target tracking.
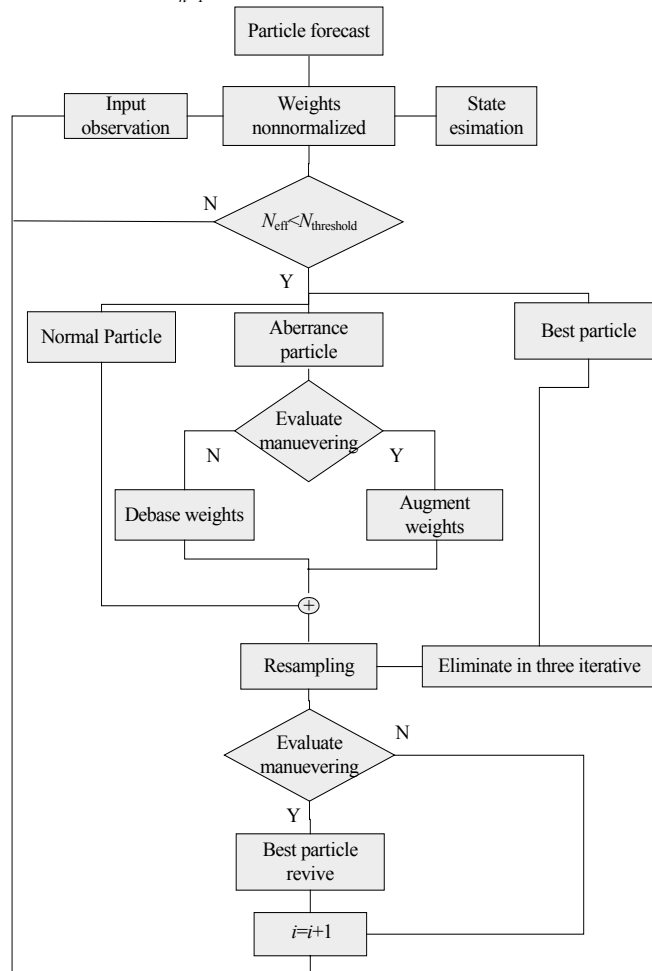

Fig. 2　Improved algorithm flow

3 Simulation

For validating the proposed algorithm, this paper takes a filter simulation for a maneuvering target tracking.

Set the nonlinear state function as:

$$x(n) = 0.5x(n-1) + \frac{25x(n-1)}{1+x(n-1)^2} +$$

$$8\cos\{1.2(n-1)\} + v(n) \qquad (9)$$

and the observation function:

$$z(n) = \frac{x(n)^2}{20} + \omega(n) \qquad (10)$$

where $v(n), \omega(n)$ is Gaussian distribution with mean 0 and variance 4, 0.01. The standard particle filter, weights choice resampling (WCR filter), linear optimal resampling (LOR filter), and the improved filter are used to estimate the target state. Fig. 3 shows the simulation results.
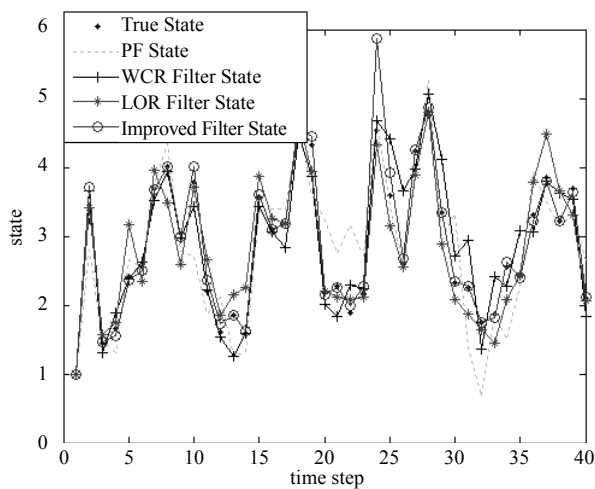

Fig. 3 Simulation results for four filters
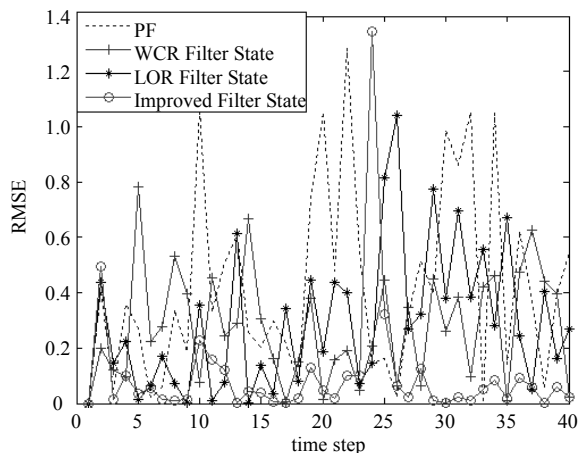
The filter estimation error is shown as Fig. 4.

Fig. 4 Estimation Errors for four filters

Table 1 Filtering performance for three filters

Algorithm	Converge NCE time	MSE	Mean of error	Average valid particle number
Standard particle filter	0.221 4	0.026 7	0.635 1	12.843
Weights choice resampling	0.157 85	0.019 62	0.331 63	16.237
Linear optimal resampling	0.168 74	0.027 74	0.312 53	17.357
Improved particle filter	0.144 62	0.011 52	0.254 06	18.412

At the same time, the Kalman filter is simulated. The performances of three filters are shown in Table 1.

4 Conclusion

This paper presents a new particle filter algorithm based on genetic algorithm optimization. This algorithm takes the descendiblity and aberrance of the genetic algorithm into particle filter resampling procedure, which seems to be more adaptive to the maneuvering target tracking. The simulation proves that the improved algorithm suits maneuvering target tracking.

Reference

[1] ARORA A. A line in the sand: a wireless sensor network for target detection, classification, and tracking[J]. Comput Netw, 2004, 46(5): 605-634.

[2] BUGALLO M F, LU T, DJURI´C P M. Target tracking by multiple particle filtering[C]//Proceedings of IEEE Aerospace Conference. Big Sky, MO, USA: IEEE, 2007: 153-156.

[3] DJURI´C P M, LU T, BUGALLO F. Multiple particle filtering[C]//Proceedings of the IEEE 32nd International Conference on Acoustics, Speech and Signal Processing (ICASSP'2007). Honolulu, Hawaii, USA: IEEE, 2007: 1181-1184.

[4] ISARD M, BLAKE A. Condensation-conditional density propagation for visual tracking[J]. IJCV, 1998, 29(1): 5-28.

[5] ASLAM J, BUTLER Z, CONSTANTIN F V, et al. Tracking a moving object with a binary sensor network[C]//Proc 1st Int Conf Embedded Networked Sensor Syst. Los Angeles, CA, USA: [s.n.], 2003:150-161.

[6] KIM W, MECHITOV K, CHOI J Y, et al. On target tracking with binary proximity sensors[C]//Proc 4th Int Symp Inf Process Sensor Netw. Los Angeles, CA, USA: IPSN, 2005.

[7] OH S, SASTRY S. Tracking on a graph[C]//Proc 4th Int Symp Inf Process Sensor Networks. Los Angeles, CA, USA: IPSN, 2005.

[8] GRISETTI G, STACHNISS C, BURGARD W. Improved

techniques for grid mapping with rao-blackwellized particle filters[J]. IEEE Transactions on Robotics, 2007, 23(1): 34-46.

[9] LASKA B N M, BOLIC M, GOUBRAN R A. Particle filter enhancement of speech spectral amplitudes[J]. IEEE Transactions on Audio, Speech, and Language Processing, 2010,18(8): 2155- 2167.

[10] GUSTAFSSON F. Particle filter theory and practice with positioning applications[J]. IEEE A&E Systems Magazine. Part 2: Tutorlals-Gustafsson, 2010, 25(7): 53-81.

[11] MARTINEZ-ESPLA J J, MARTINEZ-MARIN T, LOPEZ-SANCHEZ J M. A particle filter approach for insar phase filtering and unwrapping[J]. IEEE Transactions on Geoscience and Remote Sensing, 2009, 47(4): 1197-1211.

[12] SANKARANARAYANAN A C, SRIVASTAVA A, CHELLAPPA R. Algorithmic and architectural optimizations for computationally efficient particle filtering[J]. IEEE Transactions on Image Processing, 2008, 17(5): 737-748.

[13] CHENG Qi. An efficient two-stage sampling method in particle filter[J]. IEEE Transactions on Aerospace and Electronic Systems, 2012, 48(3): 2666-2672.

[14] WANG Ya-feng, ZHANG You-an, LIU Hua-ping, et al. Central difference particle filter applied to transfer alignment for sins on missiles[J]. IEEE Transactions on Aerospace and Electronic Systems, 2012, 48(1): 375-387.

[15] CLOSAS P, BUGALLO M F. Improving accuracy by iterated multiple particle filtering[J]. IEEE Signal Processing Letters, 2012, 19(8): 531-534.

[16] HU Xiao-li, SCHON T B, LJUNG L. A basic convergence result for particle filtering[J]. IEEE Transactions on Signal Processing, 2008, 56(4): 1337-1348.

[17] NICOLI M, MORELLI C, RAMPA V. A jump markov particle filter for localization of moving terminals in multipath indoor scenarios[J]. IEEE Transactions on Signal Processing, 2008, 56(8): 3801-3809.

[18] SEONG-HOON P W, WAEL W M, FARID G. A kalman/particle filter-based position and orientation estimation method using a position sensor/ inertial measurement unit hybrid system[J]. IEEE Transactions on Industrial Electronics, 2010, 57(5): 1787-1798.

[19] SUTHARSAN S, KIRUBARAJAN T, LANG T, et al. An optimization-based parallel particle filter for multitarget tracking[J]. IEEE Transactions on Aerospace and Electronic Systems, 2012, 48(2): 1601-1618.

[20] BRANKO R, SANJEEV A. Bernoulli particle filter with observer control for bearings-only tracking in clutter[J]. IEEE Transactions on Aerospace and Electronic Systems, 2012, 48(3): 2405- 2415.

编 辑 黄 莘