# The Application of Support Vector Regression in Particle Filtering

Xingzi Qiang
*School of Electronic and Information Engineering*
*Beihang University*
Beijing China
qiangxingzi@163.com

Rui Xue
*School of Electronic and Information Engineering*
*Beihang University*
Beijing China
xuerui@buaa.edu.cn

Yanbo Zhu
*School of Electronic and Information Engineering*
*Beihang University*
Beijing Chian
zhuyanbo@buaa.edu.cn

2021 IEEE International Conference on Information Communication and Software Engineering (ICICSE) | 978-1-6654-1919-2/21/$31.00 ©2021 IEEE | DOI: 10.1109/ICICSE52190.2021.9404107

***Abstract*—In this paper, we propose a resampling method for particle filtering (PF) based on the support vector regression (SVR). The SVR is introduced to fit the posteriori probability density function of the current state in the procedure of the filtering. Particles was resetting in the high-probability region, and the weights of these particles are calculated according the fit function. Simulation shows that the proposed resampling method is effective for non-linear filter problem.**

***Keywords*—prartilce filter, resamping method, support vector regression, nonlinear filter**

## I. Introduction

Sequential importance sampling (SIS) method was first used by Hammersley in the 1950s to solve statistical problems [1]. In the late 1960s, Handschin and Mayne applied it to solve related problems in the field of automatic control [2]. However, due to the limitation of computing power, the degeneracy and impoverishment problem, its development was very slow for a long time. In the 1990s, Simth and Gordon et al introduced Resampling into SIS method, which greatly alleviated the degeneracy and impoverishment problem, and formed SIR (Sampling importance Resampling, or called particle Filtering), which sparked a research boom. Notestar is a typical application example of particle filtering (PF) in US Navy underwater monitoring system. In the 21st century, with the increase of computer computing speed and the urgent needs of nonlinear filtering applications, PF has achieved great success in many fields, including visual tracking, target positioning, navigation, tracking, communication, signal processing, image processing, robot, satellite, remote sensing, nuclear medicine, imaging, chemical industry, finance, and economy.

The procedure of PF is sampling particles base on a proposal distribution. After that, the weights of these particles are adjusted by introducing the likelihood distribution of observation, and these weighted particles is used to represent the posteriori distribution of the state. Finally, the state is estimated by the summation of these weighted particles.

In this procedure, the selection of the proposal distribution is very important, which might affect the performance of the state estimation sensitively. Simply, when the proposal distribution is similar as the posteriori distribution, most of the particles are concentrated in the high-probability region. At this time, the state estimation of PF is approximate the Bayesian filtering. However, when the proposal distribution is far from the posteriori distribution, most of particles are concentrated in the low-probability region, which lead to the low effect of the particles and the poor performance of the state estimation.

In the past two decades, many researchers have devoted themselves to improve the proposal distribution. Extended Particle filter (EPF) and Unscented Particle filter (UPF) are two kinds of improved algorithms, which was first proposed to improve the performance of PF [3]-[5]. The core idea of EPF and UPF are to optimize the proposed distribution to make them closer to the real posterior probability distribution by using Extended Kalman Filter (EKF) [6] and Unscented Kalman Filter (UKF) [7] respectively. To a large extent, these two improved PF algorithms have achieved good results, meanwhile, they are also widely used in practical engineering [8]. On these bases, Wang [9] has proposed two-stage particle filter to further optimize the selection of the proposed distribution. Beyond that, a plenty of improved PF algorithms were proposed based on this idea, such as Iterated Extended Particle Filter (IEPF) [10], Iterated Unscented Particle Filter (IUPF) [11], Feedback Particle Filter (FPF) [12], Mixture Kalman Particle Filter (MKPF) [13], Intelligent Particle Filter (IPF) [14] and so on.

A resampling method based on the support vector regression (SVR) is proposed in this paper. the SVR is introduced to improve the distribution of the particles. The weighted particles obtained by SIS is used as samples of the SVR method. and we can calculate the parameters of fit function by the SVR method. Then the new particles are sampling in the high-probability region uniformly, and the corresponding weight is calculating according to the fit function. Finally, the posteriori distribution of the state is described by these uniform weighted particles and complete the state estimation.

## II. Support Vector Regression

SVR is widely concerned due to its excellent performance in small sample parameter estimations. In this section, we construct the model of support vector regression to estimate the probability density function.

978-0-7381-3150-4/21/$31.00 ©2021 IEEE

*A. The Theory of SVR*

Given a set of samples $\{(\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2), \cdots, (\boldsymbol{x}_m, y_m)\}$, where $\boldsymbol{x}_i \in R^d$, $y_i \in R$, the objective function of SVR is as follow :

$$\min_{w,b,\xi,\xi^*} \frac{1}{2} w^T w + C_1 \sum_i \xi_i + C_2 \sum_i \xi_i^* \tag{1}$$

And the constraint condition is:

$$\begin{aligned} & w^T \varphi(x_i) - y_i \le \varepsilon + \xi_i \\ & y_i - w^T \varphi(x_i) \le \varepsilon + \xi_i^* \\ & \xi_i^*, \xi_i \ge 0 \end{aligned} \tag{2}$$

where, $i = 1, 2, \cdots, m$ ; $\frac{1}{2} w^T w$ is structural risk, $C_1, C_2$ is penalty coefficient. The decision function can be obtained by optimizing the objective function under constraint conditions:

$$f(x) = \boldsymbol{w}^T \varphi(x) \tag{3}$$

*B. Modeling and Solving of Density Function Based on SVR*

Suppose $\{\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_m\}$ is a set of samples from the probability density function $p(\boldsymbol{x})$ and $\boldsymbol{x}_i \in R^d$. Here $p(\boldsymbol{x})$ is unknown. Then we can construct the empirical distribution function of it as follow:

$$F_m(\boldsymbol{x}) = \frac{1}{md} \sum_{i=1}^{d} \sum_{j=1}^{m} \theta\left(x^i - x_j^i\right) \tag{4}$$

where,

$$\theta(x) = \begin{cases} 1 & x > 0 \\ 0 & otherwise \end{cases} \tag{5}$$

Then the support vector regression method is used to fit the data as follow:

$$\{(\boldsymbol{x}_1, F_m(\boldsymbol{x}_1)), (\boldsymbol{x}_2, F_m(\boldsymbol{x}_2)), \cdots, (\boldsymbol{x}_m, F_m(\boldsymbol{x}_m))\}$$

The difference with general support vector regression is that:

(1) The decision function needs to ensure the definition of density function

$$p(\boldsymbol{x}) \ge 0 \text{ and } \int_{-\infty}^{+\infty} p(\boldsymbol{x}) d\boldsymbol{x} = 1 \tag{6}$$

(2) The output target is inconsistent with the that of the regression function. In the traditional support vector regression, the empirical distribution of the sample $F_m(\boldsymbol{x}_i)$ is used as the output target. However, in the problem of density function estimation, the approximation of the probability density function $p(\boldsymbol{x})$ is the output target.

To overcome the two-above problem, we reconstructed the objective function of the traditional support vector regression:

$$\min \quad F(\xi, \xi^*) = \sum_{i=1}^{m} \sum_{j=1}^{m} \beta_i \beta_j k(x_i, x_j) + C_1 \sum_{i=1}^{m} \xi + C_2 \sum_{i=1}^{m} \xi^* \tag{7}$$

$$\begin{aligned} s.t \quad & y_j - \sum_{i=1}^{m} \beta_i K(x_j, x_i) \le \varepsilon + \xi_j^* \\ & \sum_{i=1}^{m} \beta_i K(x_j, x_i) - y_i \le \varepsilon + \xi_j \\ & \sum_{i=1}^{m} \beta_i = 1 \\ & \xi_j^*, \xi \ge 0, \beta_i \ge 0 \end{aligned} \tag{8}$$

where $k(x_i, x_j)$ is the kernel function. $K(x_j, x_i)$ denotes the indefinite integral of the kernel function. $\beta_i$ is the weight of each sample. In the solution of support vector regression problem, most of the $\beta$ is similar to zero, and only a little $\beta$ is meaningful. After solving the above objective function, the form of nonparametric probability density estimation is obtained as follow:

$$p(x, \beta) = \sum_{i=1}^{m} \beta_i k(x, x_i) \tag{9}$$

When the kernel satisfies

$$\int k(x, x_i) dx = 1 \tag{10}$$

Because of the constraints $\sum_{i=1}^{m} \beta_i = 1$, The integral of the probability density function is guaranteed to be one. In the constraint $\sum_{i=1}^{m} \beta_i K(x_j, x_i)$ , In fact, we use the kernel integral to find the fitting cumulative probability of the *jth* sample. The constraints require that the fitting cumulative probability be as close as possible to the real cumulative probability $y_j$. $\xi_j^*, \xi$ Represents the deviation of the tolerance error for upward and downward respectively. $C_1$ and $C_2$ are is penalty coefficients for upward and downward.

## III. Particle Filter

The objective of recursive Bayesian filtering is to obtain the posterior PDF at the current time step and to achieve the state estimation by calculating the expectation of the posterior PDF. Recursive Bayesian filtering, which is shown in Fig. 1, can be divided into two steps: *prediction* and *update*. The prediction step infers the prior distribution at time step $k$ using the state transition distribution and posterior probability distribution at time step $k-1$:

$$p(\boldsymbol{x}_k | \boldsymbol{y}_{1:k-1}) = \int p(\boldsymbol{x}_k | \boldsymbol{x}_{k-1}) p(\boldsymbol{x}_{k-1} | \boldsymbol{y}_{1:k-1}) d\boldsymbol{x}_{k-1} \tag{11}$$

And the update infers the posterior distribution at time step $k$ using the prior distribution at time step $k$ and the likelihood distribution at time step $k$ :

$$p(\boldsymbol{x}_k | \boldsymbol{y}_{1:k}) = \frac{p(\boldsymbol{y}_k | \boldsymbol{x}_k) p(\boldsymbol{x}_k | \boldsymbol{y}_{1:k-1})}{p(\boldsymbol{y}_k | \boldsymbol{y}_{1:k-1})} \tag{12}$$

For the linear Gaussian environment, the KF is a special case of Bayesian filtering, and its analytical solution is the optimal solution under the minimum mean square error (MMSE) criterion. For a nonlinear/non-Gaussian environment, it may not be possible to obtain the analytical solution of (3) directly, which limits the direct application of recursive Bayesian filtering.

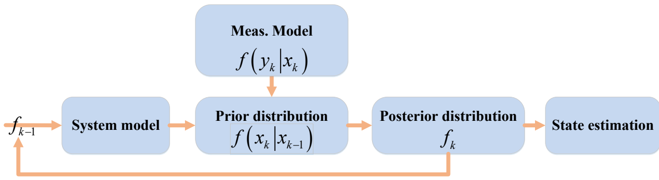


Fig. 1. Bayesian filtering framework.

To avoid difficult integral operation in the nonlinear and non-Gaussian system, PF, an approximate Bayesian filtering method, was presented based on the principle of Monte Carlo simulation.

Replacement of calculating the priori filtering density by using integral operation directly, GPF simulate the integral operation with the propagation of particles $(x_t^i)_{i=1,\cdots,N}$ and their weights $(w_t^i)_{i=1,\cdots,N}$, where $N$ denotes the number of the particles. The brief procedures of PF are as follows:

(1) Initialization: Begin by sampling particles $(x_0^i)_{i=1,\cdots,N}$ from $p(x_0)$. Setting the weights of the initial particles: $(w_0^i)_{i=1,\cdots,N} = N^{-1}$. For convenience, particles and their weights can be expressed as $\{x_0^i, w_0^i\}$.

(2) Prediction (step $t$): Calculate $\hat{x}_t^i$ according to the state model and $\{x_{t-1}^i, w_{t-1}^i\}$. After this stage, particles and their weights can be expressed as $\{\hat{x}_t^i, w_{t-1}^i\}$.

(3) Update: Calculate likelihood weights $\tilde{w}_t^i$ according to the observation model and $y_t$, particles and their weights can be expressed as $\{\hat{x}_t^i, \tilde{w}_t^i\}$.

(4) mixed weights and normalization:

$$\hat{w}_t^i = \frac{\tilde{w}_t^i w_{t-1}^i}{\sum_{i=1}^{N} \tilde{w}_t^i w_{t-1}^i} \tag{13}$$

After this stage, particles and their weights can be expressed as $\{\hat{x}_t^i, \hat{w}_t^i\}$.

(5) State estimation:

$$\bar{x}_t = \sum_{i=1}^{N} \hat{w}_t^i \hat{x}_t^i \tag{14}$$

(6) Resampling: obtain new particles and their weights $\{x_t^i, w_t^i\}$, where $w_t^i = N^{-1}$.

(7) Loop step 2-6 till the end.

The correspondence between PF and Bayesian filtering can be understood as follows:

During the execution of the PF, $\{x_0^i, w_0^i\}$ describe the initial state distribution; $\{\hat{x}_t^i, w_{t-1}^i\}$ describe the prior filtering density; $\{\hat{x}_t^i, \tilde{w}_t^i\}$ describe the likelihood density refer to $p(y_t|x_t)$; $\{\hat{x}_t^i, \hat{w}_t^i\}$ describe the posterior filtering density; The resampling step is used to mitigate the degeneracy problem. As same as $\{\hat{x}_t^i, \hat{w}_t^i\}$, $\{x_t^i, w_t^i\}$ is also used to describe the posterior filtering density. Comparing with $\{\hat{x}_t^i, \hat{w}_t^i\}$, $\{x_t^i, w_t^i\}$ remove the particles with low weights and repeat the particles with high weights, which avoid plenty of invalid calculations.

## IV. The Resampling Method Based on SVR

The resampling step is to allow the weighted samples to describe the posterior distribution more consistently. Traditional resampling methods are to duplicate high-weighted particles and omit the low-weighted particles, which always must rely on the quality of the weighted particles. When the quality of the weighted particles is not good, then, Resampling is bound to exacerbate this disadvantage, and lead to the bad result of state estimation. In this paper, we use the SVR technique to estimate the probability density function of the posteriori distribution, and the choice of the particles is not depending on the original particles, which avoid the occurrence of this disadvantage. The flow of the resampling method based on the SVR is shown in Fig. 2.

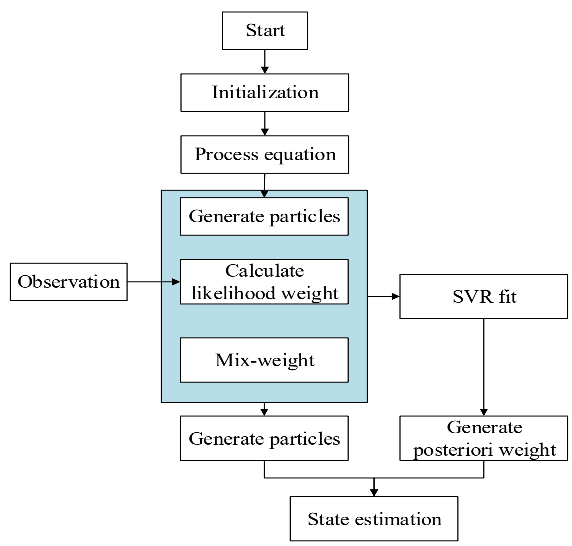


Fig. 2. The flow of the resampling method based on the SVR.

After the step (4) in section III, we obtain $\{\hat{x}_t^i, \hat{w}_t^i\}$ to describe the posteriori distribution. To improve the effect of these particles, the SVR technique is used to fit the mathematical expression of the posteriori distribution. Then, we set particles uniformly in the high-probability region of the current state space, and get a new set of particles $\{x_t^i\}_N$, and the corresponding posteriori weight $\{w_t^i\}_N$ is calculating according to (3). Finally, $\{\boldsymbol{x}_t^i, w_t^i\}$ can also be used to describe the posteriori distribution.

## V. SIMULATIONS

To demonstrate the performance of the proposed algorithm in the multi-dimensioned state space, tracking and positioning of aircraft linear motion was simulated in this experiment. Meanwhile, EKF, UKF, EPF, UPF and GPF were used to compare with the proposed algorithm. The state space model can be represented as

$$\boldsymbol{X}_t = \begin{bmatrix} \boldsymbol{F}_1 & 0 & 0 \\ 0 & \boldsymbol{F}_2 & 0 \\ 0 & 0 & \boldsymbol{F}_3 \end{bmatrix} \boldsymbol{X}_{t-1} + \boldsymbol{u}_{t-1} \tag{15}$$

$$y_{t,i} = \sqrt{\left(P_{x,t,i}^{sat} - P_{x,t}\right)^2 + \left(P_{y,t,i}^{sat} - P_{y,t}\right)^2 + \left(P_{z,t,i}^{sat} - P_{z,t}\right)^2} + v_{t,i} \qquad i = 1,2,\cdots L \tag{16}$$

where $\boldsymbol{X}_t = \left[P_{x,t}, V_{x,t}, P_{y,t}, V_{y,t}, P_{z,t}, V_{z,t}\right]^T$,

$$\boldsymbol{F}_i = \begin{bmatrix} 1 & T_0 \\ 0 & 1 \end{bmatrix},\ i = 1,2,3 \qquad \boldsymbol{u} \sim N(0,\ 10).$$

$P^{sat}$ denotes the position of satellite. $P$ and $V$ denotes the position and velocity of the aircraft in different directions at time step $t$ respectively. $T_0$ is sampling period. $L$ denotes the number of visible satellites.

$\boldsymbol{u}_{t-1}$ and $v_t$ represents the processing and observation noise respectively. where the covariance matrix of processing noise can be described as

$$\boldsymbol{Q} = \sigma^2 \begin{bmatrix} \boldsymbol{q}_1 & 0 & 0 \\ 0 & \boldsymbol{q}_2 & 0 \\ 0 & 0 & \boldsymbol{q}_3 \end{bmatrix} \text{ and } \boldsymbol{q}_i = \begin{bmatrix} \frac{1}{4}T_0^4 & \frac{1}{2}T_0^3 \\ \frac{1}{2}T_0^3 & T_0^2 \end{bmatrix} \quad i = 1,2,3. \tag{17}$$

$$v_t \sim La\left(0, \sqrt{5}\right) \tag{18}$$

In this experiment, the trajectory of 24 GPS satellites was simulated by MATLAB. Satellite be visible when the elevation of it is greater than 5 degrees. Earth-Centered Earth-Fixed (ECEF) coordinate system was used in this scenario. The number of particles is 1500. The initial state is as follow

$$\boldsymbol{X}_0 = [4153646.92867716,100 \quad 424811.754823606,-50 \quad 4831776.39999292,-10]^T \tag{19}$$

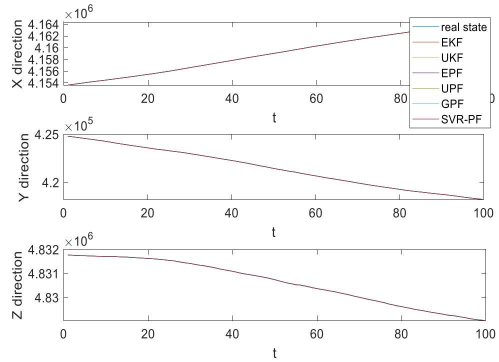


Fig. 3. State estimating results.

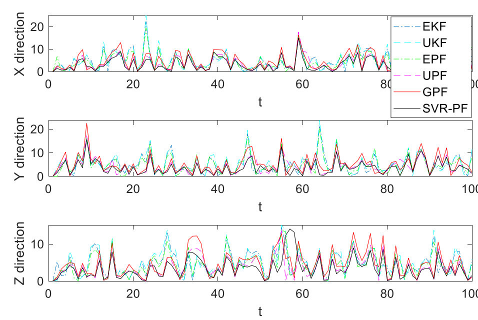


Fig. 4. Absolute errors of state estimation.

The results of the position estimation and the absolute errors of estimation can be seen in Fig. 3 and Fig. 4. The mean and variance of the errors in different directions can be seen in Table I. They show that the performance of the proposed algorithm is better than other filtering algorithms.

TABLE I. THE MEAN AND VARIANCE OF THE ERRORS IN DIFFERENT DIRECTIONS

| | | EKF | UKF | EPF | UPF | GPF | SVR-PF |
|---|---|---|---|---|---|---|---|
| mean | X | -0.8100 | -0.7840 | -08386 | -1.1784 | 1.1905 | -0.0583 |
| | Y | -1.1463 | -1.2063 | -1.0649 | -0.7151 | 0.5886 | -0.0490 |
| | Z | 0.1031 | 0.1928 | 0.0369 | -0.4641 | 0.7991 | -0.0944 |
| variance | X | 39.4263 | 44.6685 | 36.4056 | 27.1014 | 40.2219 | 19.7659 |
| | Y | 41.9933 | 47.7648 | 39.8822 | 25.3336 | 40.9435 | 22.1459 |
| | Z | 33.4736 | 38.0891 | 29.6838 | 21.7620 | 33.3973 | 18.9052 |

## VI. CONCULSIONS

In this paper, the SVR technique was introduced to improve the effect of weighted particles in the PF algorithm. The traditional resampling method was replaced by SVR fit. Simulation shows that the SVR resampling method can improve the filtering performance in the application of aircraft integrated navigation.

## Acknowledgment

This work was supported in part by the National Key Research and Development Program of China under Grant 2017YFB0503400 and in part by the National Natural Science Foundation of China under Grant U2033215, U1833125.


## References


[1] H. L. Frisch, S. B. Gordon, V. A. Vyssotsky and J. M. Hammersley, "Monte Carlo solution of bond percolation processes in various crystal lattices," in The Bell System Technical Journal, vol. 41, no. 3, pp. 909-920, May 1962

[2] F. Galiana, E. Handschin and A. Fiechter, "Identification of stochastic electric load models from physical data," in IEEE Transactions on Automatic Control, vol. 19, no. 6, pp. 887-893, December 1974

[3] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," *IEEE Trans. Signal Process.*, vol. 50, no. 2, pp. 174–188, Feb. 2002.

[4] Van Der Merwe, Rudolph, et al. "The unscented particle filter." *Advances in neural information processing systems*. 2001.

[5] S. J. Julier and J. K. Ulhmann, "Unscented filtering and nonlinear estimation," *Proc. IEEE*, vol. 92, no. 3, pp. 401–422, Mar. 2004.

[6] S. Julier and J. Uhlmann, "A New Extension of the Kalman Filter to Nonlinear Systems," *Proc. SPIE*, vol. 3068, pp. 182-193, 1997.

[7] N. Gordon, D. Salmond, and A. F. M. Smith, "Novel approach to nonlinear/non-gaussian Bayesian state estimation," *IEE Proc. -F Radar, Sonar Navig*., vol. 140, pp. 107–113, 1993

[8] Zhou, Ning, Da Meng, and Shuai Lu. "Estimation of the dynamic states of synchronous machines using an extended particle filter." *IEEE Transactions on Power Systems* 28.4 (2013): 4152-4161.

[9] F. Wang, J. Zhang, B. Lin, and X. Li, ''Two stage particle filter for nonlinear Bayesian estimation,'' IEEE Access, vol. 6, pp. 13803–13809, 2018.

[10] L. Li, H. Ji, and J. Luo, ``The iterated extended Kalman particle filter," in *Proc. ISCIT*, Beijing, China, 2005, pp. 1213_1216.

[11] F. Wang, X. Li, and M. Lu, ``Improving particle _lter with better proposal distribution for nonlinear filtering problems," in *Proc. WASA*, Zhangjiajie, China, 2013, pp. 1_12

[12] Zhang, Chi, Amirhossein Taghvaei, and Prashant G. Mehta. "Feedback Particle Filter on Riemannian Manifolds and Matrix Lie Groups." *IEEE Transactions on Automatic Control* 63.8 (2018): 2465-2480.

[13] F. Wang and Q. Zhao, ``A new particle _lter for nonlinear _ltering problems,"*Chin. J. Comput.*, vol. 31, no. 2, pp. 346_352, 2008.

[14] S. Yin, X. Zhu. "Intelligent particle filter and its application to fault detection of nonlinear system," *IEEE Trans. Ind. Electro*, vol. 62, no. 6, pp.3852-3861, 2015.