A Particle Bernoulli Filter Based on Gaussian Process Learning for Maneuvering Target Tracking

Zheng Hu
Key Lab of Information Fusion Technology
Northwestern Polytechnical University
Xi'an, China
zheng.hoo@mail.nwpu.edu.cn

Tiancheng Li
Key Lab of Information Fusion Technology
Northwestern Polytechnical University
Xi'an, China
t.c.li@nwpu.edu.cn

***Abstract*—In this work, the Bayes-optimal Bernoulli filter (BF) is studied for the target tracking where the target is randomly present or absent in the view field of the sensor while the sensor may provide imperfect measurement which contains miss detection and false alarm. To solve the issue that the dynamic model of the target is switching in an unknown mode, we employ the Gaussian process (GP) regression tool, which is a data-driven approach for learning the motion model online, to approximate the transitional density in the formulation of the BF. To deal with the nonlinear measurement model, the proposed GP-based BF is implemented using particles. In the simulation experiment, the proposed approach is performed on a maneuvering target tracking scenario and compared with the Bernoulli particle filters utilizing the full or partial model changing information.**

***Index Terms*—Gaussian process regression, Bernoulli filter, data-driven approach, particle filter.**

I. Introduction

The target tracking problem, referred to the estimation of the latent state of interest from the noise corrupted signal or imperfect measurement in discrete time, has been widely used in the engineering field. Classic methods for target tracking are model-dependent, in which the state-space models are modeled as a Markov process of the unobserved state, thereby inferring the state of interest through the Bayesian framework [1]. There are two challenges in real target tracking applications: the reliable dynamic models and statistic noises are time-varying and hard to be modeled due to the maneuvering of target; the targets are stochastically present or absent in the view field of sensor owing to the physical characteristics. s According to the formulation of standard Bayes filters [1], the inevitable components are the dynamics and measurement models, which are described by some parameters about the involved physical processes. Although these parametric models are convenient to be designed and computationally efficient, accurate parametric models are difficult to obtain, which in turn may limit the performance.

This work was partially supported by the National Natural Science Foundation of China (62071389), by the JWKJW Foundation (2021-xxxx-JJ-0897, 2020-xxxx-ZD-150-12), and by the Key Laboratory Foundation of National Defence Technology (JKWATR-210504).

Z. Hu, and T. Li are all with the Key Lab of Information Fusion Technology (Ministry of Education), School of Automation, Northwestern Polytechnical University, Xi'an 710129, China, e-mail: zheng.hoo@mail.nwpu.edu.cn, t.c.li@nwpu.edu.cn.

To overcome these limitations, the Gaussian process regression (GPR) provides a non-parametric method successfully applied in learning prediction and observation models for dynamic systems. Gaussian process (GP) [2] assumes that a distribution of the approximated function is specified by the mean and covariance functions. Due to the key feature that GPs calculate not only the mean of the predictive states but also the uncertainty of these states, GPs are employed to deal with the problem of system identification, [3], [4]. [5] integrates GP into various forms of Bayes filters and presents the GP-BayesFilters, so GPR models are combined with the extended Kalman filter (GP-EKF), the unscented KF (GP-UKF) and the particle filter (GP-PF). [6] incorporates GP into the cubature KF (CKF) and proposes consensus cubature filtering algorithm based on GP (CCF-GP) in a distributed sensor network. Gaussian process motion tracker (GPMT) is addressed in [7], [8], which is a data-driven approach for target tracking with online training and parameter learning.

Due to the background clutters and imperfect detection, a dynamic Bernoulli phenomenon that a target can enter and exit the view of sensors at random instances is taken into account in some target tracking applications. [9], [10] model this phenomenon as a Bernoulli process for the state transition and vague measurement respectively based on the finite set statistic (FISST) [11], which leads to the so-called Bernoulli filters (BFs). [10], [11] have illustrated that the BF is the optimal Bayes filter for a single dynamic system which can randomly switch on or off. The BF has received considerable attention in the field, which has been furthered extended to account for the point measurements with amplitude [12], for the distributed implementation [13], etc.

In order to deal with target motion model switching in an unknown manner, this paper proposes a Bernoulli particle filter based on GP which learns the prediction models from the posterior states and realizes the joint detection and tracking for intensity measurements by incorporating GP into Bernoulli particle filter. Moreover, The numerical simulation is designed to illustrate the feasibility and effectiveness of our method by comparing with these standard BFs utilizing the full or partial model changing information.

The remaining part of this article is structured as follows. Background knowledge and notation are given in Section II.

The proposed BF based on GP is introduced in Section III. The simulation results are presented in Section IV. Finally, Section V concludes this paper.

II. Preliminaries and Notation

A. Random Finite Sets

A random finite set (RFS) $\mathbf{X}$ is a random variable with respect to unordered finite sets, which is completely determined by its cardinality distribution $\rho(n) = P(|\mathbf{X}| = n)$, $n \in \mathbb{N}_0$ and a family of symmetric joint distribution $p_n(\mathbf{x}_1, \ldots, \mathbf{x}_n)$. According to the FISST [11], the FISST probability density function (PDF) $f(\mathbf{X})$ is expressed as:

$$f(\mathbf{X}) = f(\{\mathbf{x}_1, \ldots, \mathbf{x}_n\}) = n! \cdot \rho(n) \cdot p_n(\mathbf{x}_1, \ldots, \mathbf{x}_n). \quad (1)$$

The set integral is defined as

$$\int f(\mathbf{X})\delta\mathbf{X} = f(\emptyset) + \sum_{n=1}^{\infty} \frac{1}{n!} \int f(\{\mathbf{x}_1, \ldots, \mathbf{x}_n\})d\mathbf{x}_1 \ldots \mathbf{x}_n. \quad (2)$$

The Bernoulli RFS and the Poisson RFS are relevant for this paper since the intensity measurement for a single target is only taken into account. The Bernoulli RFS $\mathbf{X}$, whose cardinality distribution $\rho(n)$ is Bernoulli, can either be empty (with probability $1-q$) or have one element (with probability q) complying with the distribution $p(\mathbf{x})$. The FISST PDF $f(\mathbf{X})$ is given as:

$$f(\mathbf{X}) = \begin{cases} 1-q & \text{if } \mathbf{X} = \emptyset \\ q \cdot p(\mathbf{x}) & \text{if } \mathbf{X} = \{\mathbf{x}\} \end{cases} . \quad (3)$$

The Poisson RFS $\mathbf{X}$ is defined as its cardinality distribution is Poisson. The FISST PDF of Poisson RFS can be derived by Equation (1) as following

$$f(\mathbf{X}) = e^{-\lambda} \prod_{\mathbf{x} \in \mathbf{X}} \lambda p(\mathbf{x}), \quad (4)$$

where λ denotes the expected number of the elements.

B. Bernoulli Filter

Let $f_{k|k}(\mathbf{X}_k|\mathbf{Z}_{1:k})$ denote the posterior FISST PDF at time k. According to the structure of a Bernoulli RFS, like (3), it can be uniquely determined by two quantities: a) the posterior probability of target existence $q_{k|k}$; b) the posterior spatial PDF $s_{k|k}$. Therefor the BF only needs to propagate the pair $(q_{k|k}, s_{k|k})$ that contains these two quantities.

Keeping consistent with the framework of Bayes filter, the prediction equations of the BF, originally derived in [11] via the Chapman—Kolmogorov equation, are given by:

$$q_{k|k-1} = p_b(1 - q_{k-1|k-1}) + p_s q_{k-1|k-1}, \quad (5)$$

$$s_{k|k-1}(\mathbf{x}) = \frac{p_b(1 - q_{k-1|k-1})b_{k|k-1}(\mathbf{x})}{q_{k|k-1}} + \frac{p_s q_{k-1|k-1} \int \pi_{k|k-1}(\mathbf{x}|\mathbf{x}')s_{k-1|k-1}(\mathbf{x}')d\mathbf{x}'}{q_{k|k-1}}, \quad (6)$$

in which p_b is the probability of target born during the sampling interval with the birth density $b_{k|k-1}(\mathbf{x}_k)$; p_s is the probability of target survival from time $k-1$ to k with the transitional density $\pi_{k|k-1}(\mathbf{x}_k|\mathbf{x}_{k-1})$.

In this paper, we only focus on the intensity measurement model. Suppose the sensor contains $n \geq 1$ intensity measuring elements reporting simultaneously a measurement value $z_k^{(s)}$, $s = 1, \ldots, n$ at time k on the known locations. All n measurements are collected into a single measurement RFS $\mathbf{z}_k = [z_k^{(1)}, \ldots, z_k^{(n)}]^{\mathrm{T}}$. The intensity contribution of the element s can be expressed as:

$$z_k^{(s)} = \begin{cases} h_k^{(s)}(\mathbf{x}_k) + w_k^{(s)} & \text{if } \mathbf{X}_k = \{\mathbf{x}_k\} \\ w_k^{(s)} & \text{if } \mathbf{X}_k = \emptyset \end{cases} , \quad (7)$$

where $h_k^{(s)}(\mathbf{x}_k)$ is the intensity contribution in element s according to the measurement likelihood $g_1^{(s)}$, and $w_k^{(s)}$ is the background noise subjecting to the PDF $g_0^{(s)}$. Then the likelihood function of the measurement vector $\mathbf{z}_k$ can be calculated as:

$$\varphi_k(\mathbf{z}_k|\mathbf{X}_k) = \begin{cases} \prod_{s=1}^{n} g_1^{(s)}(z_k^{(s)}|\mathbf{x}) & \text{if } \mathbf{X}_k = \{\mathbf{x}\} \\ \prod_{s=1}^{n} g_0^{(s)}(z_k^{(s)}) & \text{if } \mathbf{X}_k = \emptyset \end{cases} . \quad (8)$$

According to the update step of Bayes filter, the update equations of the BF, referred to [10], are given as:

$$q_{k|k} = \frac{q_{k|k-1} \int \ell_k(\mathbf{z}_k|\mathbf{x})s_{k|k-1}(\mathbf{x})d\mathbf{x}}{1 - q_{k|k-1} + q_{k|k-1} \int \ell_k(\mathbf{z}_k|\mathbf{x})s_{k|k-1}(\mathbf{x})d\mathbf{x}}, \quad (9)$$

$$s_{k|k}(\mathbf{x}) = \frac{\varphi_k(\mathbf{z}_k|\{\mathbf{x}\})s_{k|k-1}(\mathbf{x})}{\int \varphi_k(\mathbf{z}_k|\{\mathbf{x}\})s_{k|k-1}(\mathbf{x})d\mathbf{x}} \quad (10)$$

with the measurement likelihood ratio:

$$\ell_k(\mathbf{z}_k|\mathbf{x}) = \frac{\varphi_k(\mathbf{z}_k|\{\mathbf{x}\})}{\varphi_k(\mathbf{z}_k|\emptyset)} = \prod_{s=1}^{n} \frac{g_1^{(s)}(z_k^{(s)}|\mathbf{x})}{g_0^{(s)}(z_k^{(s)})}. \quad (11)$$

C. Gaussian Process

A GP is a collection of random variables, any finite number of which have a joint Gaussian distribution [2]. Suppose there is a training data set $\mathcal{D} = \{(\mathrm{x}_i, y_i)|i = 1, \ldots, l\}$ containing l training points (x_i, y_i), in which x_i is the d-dimensional input feature vector and y_i is a scalar training output corresponding to x_i. One assumes that these training points are derived from the noisy process $y_i = f(\mathrm{x}_i) + \epsilon, i = 1, \ldots, l$, where $f(\cdot)$ is the unknown function to be learned by a GP, ϵ denotes a additive noise following an IID Gaussian distribution with zero mean and variance σ_l^2. The GP will can be simply written as $f(\mathrm{x}) \sim \mathcal{GP}(m(\mathrm{x}), k(\mathrm{x}, \mathrm{x}'))$, in which $m(\mathrm{x})$ and $k(\mathrm{x}, \mathrm{x}')$ are the mean function and the covariance (kernel) function of a real process $f(\mathrm{x})$, respectively.

The covariance function is the crucial component of a GP, whose choice lies on the characteristics of the process. The most widely used form is the squared-exponential (SE) covariance function [14]:

$$k(\mathrm{x}, \mathrm{x}') = \sigma_f^2 \exp\{-\frac{1}{2}(\mathrm{x} - \mathrm{x}')W(\mathrm{x} - \mathrm{x}')^{\mathrm{T}}\} + \sigma_l^2\delta, \quad (12)$$

where σ_f^2 and σ_l^2 are the signal variance and noise variance, respectively. $\bullet^{\mathrm{T}}$ denotes the transpose operator. The diagonal matrix $W = \mathrm{diag}(1/L_1^2, 1/L_2^2, \ldots, 1/L_l^2)$ is the length-scales matrix. x and x$'$ are either training or testing vector. δ represents the Kronecker-delta function which is one if $\mathrm{x} = \mathrm{x}'$ or zero otherwise.

The free parameters of the covariance function $\theta = [W, \sigma_f, \sigma_l]$ are referred to as Hyperparameters. θ can be determined by $\theta_{\max} = \arg\max_{\theta} \log(p(\mathbf{y}|X, \theta))$, and the log marginal likelihood $\log(p(\mathbf{y}|X, \theta))$ is given in [2]. The solution of Hyperparameters can be obtained via some numerical optimization techniques [15].

Given the test input x_*, the predictive mean and variance of Gaussian distribution over the output y_* are respectively given by [2]:

$$\mathrm{GP}_\mu(\mathrm{x}_*, \mathcal{D}) = \mathbf{k}_*^{\mathrm{T}} K^{-1} \mathbf{y}, \tag{13}$$

$$\mathrm{GP}_\Sigma(\mathrm{x}_*, \mathcal{D}) = k(\mathbf{x}_*, \mathbf{x}_*) - \mathbf{k}_*^{\mathrm{T}} K^{-1} \mathbf{k}_*, \tag{14}$$

in which K is the l by l kernel matrix and $K[i,j] = k(\mathrm{x}_i, \mathrm{x}_j)$, $\mathrm{GP}_\mu(\mathrm{x}_*, \mathcal{D})$, $\mathbf{k}_*$ is a vector whose elements between $\mathbf{x}_*$ and X are calculated by the covariance function.

III. Proposed Gaussian Process Bernoulli Filter

In this section, we will first design a GPR method to learn the prediction models from the time-series posterior states. Then this method is incorporated with the BF in the sequential Monte Carlo framework to realize the joint detection and tracking for the intensity measurement model.

A. GPR Model for Recursive Prediction

The GPR model for recursive prediction assumes that each dimension of the target state is not correlated in the predictive time instance, but these dimensional states may be correlated in the previous moments. Suppose that the posterior estimated states in time-series $\{\hat{\mathbf{x}}_1, \hat{\mathbf{x}}_2, \ldots, \hat{\mathbf{x}}_k\}$ are available, in which $\hat{\mathbf{x}}_t^i$ is the i-th dimension of the posterior state $\hat{\mathbf{x}}_t$ at time instance t. The predictive state is denoted as $\hat{\mathbf{x}}_{k+1|k}$, each dimension of variable $\hat{\mathbf{x}}_{k+1|k}^i$ is nonlinearly dependent on the function of time f^i with some uncertainties, which is modeled by a GP, i.e.,

$$\hat{\mathbf{x}}_{k+1|k}^i = f^i(t) + \epsilon_i, i = 1, 2, \ldots, n_x, \tag{15}$$

$$f^i(\mathbf{t}) \sim \mathcal{GP}^i(m(\mathbf{t}), k(\mathbf{t}, \mathbf{t}')). \tag{16}$$

We note that the formulation (15) resembles the trajectory of function of time [16], [17], which is free of Markov modeling. By this, we are able to learn the trajectory of the target from the data rather than from an assumed Markov model. The corresponding training data set $\mathcal{D}^i = (\mathbf{t}_k, \mathbf{X}^i)$ is selected from the d most recent posterior estimated states, where $\mathbf{t}_k = [k - d + 1, k - d + 2, \ldots, k]^{\mathrm{T}}$ represents the the d most recent time instances and $\mathbf{X}^i = [\hat{\mathbf{x}}_{k-d+1}^i, \hat{\mathbf{x}}_{k-d+2}^i, \ldots, \hat{\mathbf{x}}_k^i]^{\mathrm{T}}$ represents the training output consisting of the i-th dimensional variables of these corresponding posterior states. d is also called the depth of the training. The predictive mean and variance of the state for i-th dimension at time $\mathbf{t}_* = k + 1$ are respectively given by:

$$\mathrm{GP}_\mu^i(\mathbf{t}_*, \mathcal{D}^i) = \mathbf{k}_*^{\mathrm{T}} K^{-1} \mathbf{y}^i, \tag{17}$$

$$\mathrm{GP}_\Sigma^i(\mathbf{t}_*, \mathcal{D}^i) = k(\mathbf{t}_*, \mathbf{t}_*) - \mathbf{k}_*^{\mathrm{T}} K^{-1} \mathbf{k}_*. \tag{18}$$

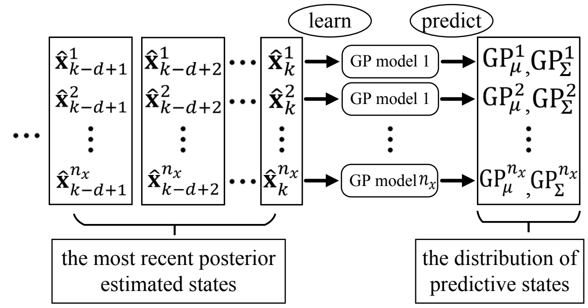

Fig. 1: Learning and prediction process for GP model

According to the learning and testing processes shown in Fig. 1, the distributions of variables of each dimension are predicted separately by GPR. Assuming that the variables in each dimension are independent and obey Gaussian distribution, the probabilistic distribution of the predictive state $\hat{\mathbf{x}}_{k+1|k}$ can be expressed as:

$$P_{\mathrm{GP}}(\hat{\mathbf{x}}_{k+1|k}) = \mathcal{N}(\mathrm{GP}_\mu, \mathrm{GP}_\Sigma), \tag{19}$$

in which $\mathrm{GP}_\mu = [\mathrm{GP}_\mu^1(\mathbf{t}_*, \mathcal{D}^1), \ldots, \mathrm{GP}_\mu^{n_x}(\mathbf{t}_*, \mathcal{D}^{n_x})]^{\mathrm{T}}$ and $\mathrm{GP}_\Sigma = \mathrm{diag}\{[\mathrm{GP}_\Sigma^1(\mathbf{t}_*, \mathcal{D}^1), \ldots, \mathrm{GP}_\Sigma^{n_x}(\mathbf{t}_*, \mathcal{D}^{n_x})]\}$ are the mean and covariance of the Gaussian predictive distribution.

Unlike the GMPT proposed in [7], [8], our presented learning and testing processes do not employ the position information transferred by the measurements, but the posterior states estimated by the designed filter. The proposed Bernoulli particle filter Based on GP predicts the state through the most recent posterior states and corrects the predictive state by the last measurement.

B. Bernoulli Particle Filter Based on GP

Since the recursions of the BF have no analytic solution in general, we usually resort to some approximate treatments, e.g. particles or Gaussian mixture [10], [18]. The Bernoulli particle filter employs a series of particles $\{w_k^{(i)}, \mathbf{x}_k^{(i)}\}_{i=1}^N$ to approximate the spatial PDF $s_{k|k}(\mathbf{x})$ by the state of the art sequential Monte Carlo method [19]–[22], in which $\mathbf{x}_k^{(i)}$ and $w_k^{(i)}$ denote the state of particle i and its corresponding normalized weight, i.e., $\sum_{i=1}^N w_k^{(i)} = 1$, respectively.

Given the probability of existence is $q_{k-1|k-1}$ at time $k-1$, and based on the core of the sequential Monte Carlo method, the posterior spatial PDF can be expressed by

$$\hat{s}_{k-1|k-1}(\mathbf{x}) = \sum_{i=1}^N w_{k-1}^{(i)} \delta_{\hat{\mathbf{x}}_{k-1}^{(i)}}(\mathbf{x}). \tag{20}$$

According to the expect a posterior (EAP), the posterior point estimated state $\hat{\mathbf{x}}_{k-1}$ is given as: $\hat{\mathbf{x}}_{k-1} = \sum_{i=1}^N w_{k-1}^{(i)} \mathbf{x}_{k-1}^{(i)}$.

For the prediction step of the BF, the predictive probability of existence is directly calculated by (5). The predictive spatial PDF $\hat{s}_{k|k-1}(\mathbf{x})$ is also approximated by a group of particles, i.e.,

$$\hat{s}_{k|k-1}(\mathbf{x}) = \sum_{i=1}^{N+B} w_{k|k-1}^{(i)} \delta_{\hat{\mathbf{x}}_{k|k-1}^{(i)}}(\mathbf{x}), \tag{21}$$

in which B is the number target-birth particles drawn from the proposal distribution β_k, discussed in the standard particle filter [19], [23], [24]. The particles are generated from two proposal distributions

$$\mathbf{x}_{k|k-1}^{(i)} \sim \begin{cases} \rho_k(\mathbf{x}_k|\mathbf{x}_k^{(i)}, \mathbf{Z}_k) & i = 1, \ldots, N \\ \beta_k(\mathbf{x}_k|\mathbf{Z}_k) & i = N+1, \ldots, N+B \end{cases} \tag{22}$$

with the weights

$$w_{k|k-1}^{(i)} = \begin{cases} \dfrac{p_s q_{k-1|k-1}}{q_{k|k-1}} \dfrac{P_{\mathrm{GP}}(\mathbf{x}_{k|k-1}^{(i)}) w_{k-1}^{(i)}}{\rho_k(\mathbf{x}_k|\mathbf{x}_k^{(i)}, \mathbf{Z}_k)}, \\ i = 1, \ldots, N, \\ \dfrac{p_s(1-q_{k-1|k-1})}{q_{k|k-1}} \dfrac{b_{k|k-1}(\mathbf{x}_{k|k-1}^{(i)})}{\beta_k(\mathbf{x}_k|\mathbf{Z}_k)}, \\ i = N+1, \ldots, N+B. \end{cases} \tag{23}$$

As for the update equations (9) and (10), the likelihood ratios for each particle can be straightforwardly computed and the integral in (9) is approximated as

$$I_k = \int \ell_k(\mathbf{z}_k|\mathbf{x}) s_{k|k-1}(\mathbf{x}) d\mathbf{x} \approx \sum_{i=1}^{N+B} \ell_k(\mathbf{z}|\mathbf{x}_{k|k-1}^{(i)}) w_{k|k-1}^{(i)}. \tag{24}$$

Except the prediction steps which employ the GP to forecast the distribution of the state at next time instance, the process of the particle BF based on GP is similar to the standard Bernoulli particle filter, whose pseudo-code is detailed as the Algorithm 1 in [10].

IV. Numerical Simulations

In numerical simulation, we consider a scene where the target turns around on the x-y coordinates plane, which the target switches the motion model between constant velocity (CV) and constant turn (CT) models. The sensor located at the origin of coordinate measures the bearing-range data. The state variable of the target $\mathbf{x}_k = [p_{x,k}, \dot{p}_{x,k}, p_{y,k}, \dot{p}_{y,k}]^{\mathrm{T}}$ comprises the position and velocity for each coordinate. The state transition model is $\mathbf{x}_k = F_{CV/CT}\mathbf{x}_{k-1} + G\mathbf{w}_{k-1}$, where F_{CT} and F_{CV} are the state transitional function of CT and CV, respectively. F_{CT}, F_{CV} and G can be referred to [25]. Sampling interval $T = 1$ s, process noise $\mathbf{w}_k \sim \mathcal{N}(0, \mathrm{diag}([\sigma_x^2, \sigma_y^2]))$ with $\sigma_x = \sigma_y = 5$ ms^{-2}, and the turn rate $\omega = 1.5\pi/180$ rad.

The measurement variables of sensor $\mathbf{z}_k = [\theta, \gamma]^{\mathrm{T}}$ can be described as $\mathbf{z}_k = [\arctan(p_{y,k}/p_{x,k}), \sqrt{p_{x,k}^2 + p_{y,k}^2}]^{\mathrm{T}} + \mathbf{v}_k$, where the measurement noise $\mathbf{v}_k \sim \mathcal{N}(0, \mathrm{diag}([\sigma_\theta^2, \sigma_\gamma^2]))$ with $\sigma_\theta = \pi/180$ rad and $\sigma_\gamma = 20$ m. The observation region is the half disc $[0, \pi]$ rad by $[0, 2000]$ m. Background clutter subjects to a Poisson RFS with a mean rate of 20 return per scan and a uniform spatial distribution on the observation region.

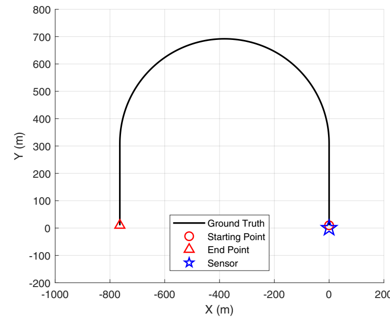

Fig. 2: Target trajectory and sensor position

The target appears at time $k = 10$ s, follows the CV model in the time interval $[11, 40]$, the CV model in $[41, 160]$ and the CV model in $[161, 190]$ successively, and then disappears after time $k = 191$ s. The target trajectory is shown as Fig. 2, the red circle and triangle respectively denote the start and end point of the trajectory, respectively, and the blue star indicates the position of sensor.

In the setting of all filters, the probability of survival is $p_s = 0.9$, the probability of detection is $p_d = 0.85$, and the probability of target birth is $p_b = 0.1$ with the birth density $b_{k|k-1}(\mathbf{x}) = \mathcal{N}(\mathbf{x}; m_b, Q_b)$ where $m_b = [-20, -2, 20, 10]$ and $Q_b = \mathrm{diag}([100, 10, 100, 10]^2)$. The number of particles sampled from the proposal prediction distribution is $N = 1500$ and the number of particles drawn directly from the birth density is $B = 500$ at each time step. The depth of training is $d = 8$ for the BF based on GP, denoted as GPBF. The standard Bernoulli particle filter with perfect Fdynamic models, i.e., switch ideally between CV and CT, is denoted as ideal BF in the following figures and tables. In addition, the Bernoulli filters which only employ a single CT or CV model are denoted as BF (CT) and BF (CV), respectively. Except the ideal Bernoulli particle filter, all filters do not have the information about motion models switching. The tracking results of the BF based on GP is given in Fig. 3.

Although the estimation of the proposed method initiates and terminates the true track with some time steps delay, it can be found that the estimates of both x-coordinate and y-coordinate positions can concentrate on the true values of target track in the presence of clutters. Therefore the proposed approach adopts to the changing motion model scenarios through learning the GP models from the posterior estimated states and is feasible for the scene exists randomly on/off switching and imperfect detection.

To evaluate the performance of these filters, the optimal sub-pattern assignment (OSPA) distance, referred to as [26], is used. The cutoff parameter of OSPA is $c = 100$ and order $p = 2$. The OSPA distance for 100 Monte Carlo runs of four

TABLE I: Processing Time in Milliseconds

Method	Ideal BF	BF(CT)	BF(CV)	GPBF
Time (msec)	2.6802	2.6803	2.6980	23.1472

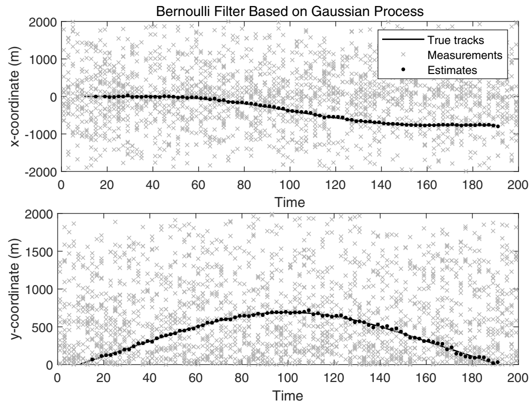

Fig. 3: Measurements and position estimates

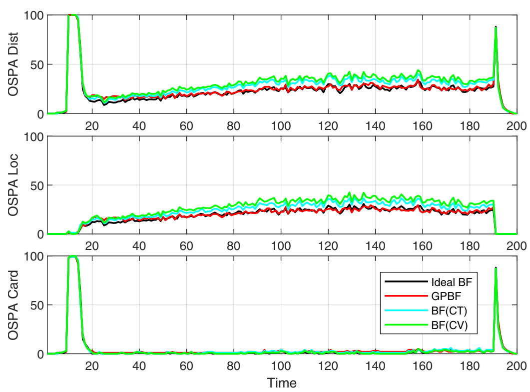

Fig. 4: OSPA distance for Bernoulli particle filters

BFs are given in Fig. 4, which shows the total OSPA, along with its localization and cardinality components. From the Fig. 4, it can be observed that the OSPA distance for the proposed method is closer to the ideal Bernoulli particle filter than the others. The BF which utilizes a single CT or CV model tracks the target with a poor performance than the others.

Moreover, the average processing time of each filter iteration is given in Table I. The processing time of Bernoulli particle filter based on GP is much longer than the other three methods whose processing times are nearly equal. This is owing to that the proposed method requires learning the GP model and prediction for particles at each iteration, which makes it computationally costly. It is therefore our future work to speed up the GP learning speed.

V. Conclusions

In this paper, we propose the Bernoulli particle filter based on GP. Unlike the model-based approaches that need a-priori, exact knowledge about the the target dynamics, the proposed approach employ GP to learn the prediction models from the time-series posterior states for recursive prediction in the BF. Simulation results illustrate the proposed method is feasible and effective for the target maneuvering and imperfect detection scenario. However, the GP learning efficiency remains a concern.

References

[1] S. Särkkä, *Bayesian Filtering and Smoothing*. New York, NY, USA: Cambridge Univ. Press, 2013.

[2] C. K. Williams and C. E. Rasmussen, *Gaussian Processes for Machine Learning*. MIT press Cambridge, MA, 2006, vol. 2, no. 3.

[3] D. B. Grimes, R. Chalodhorn, and R. P. Rao, "Dynamic imitation in a humanoid robot through nonparametric probabilistic inference." in *Robotics: science and systems*. Cambridge, MA, 2006, pp. 199–206.

[4] F. Tobar, P. M. Djurić, and D. P. Mandic, "Unsupervised state-space modeling using reproducing kernels," *IEEE Trans. Signal Process.*, vol. 63, no. 19, pp. 5210–5221, 2015.

[5] J. Ko and D. Fox, "GP-Bayesfilters: Bayesian filtering using Gaussian process prediction and observation models," *Auton. Robot.*, vol. 27, no. 1, pp. 75–90, 2009.

[6] L. Song, Z. Jing, P. Dong, R. Zou, and S. Chen, "Consensus cubature filtering based on Gaussian process for distributed sensor network with model uncertainty," *Signal Process.*, vol. 190, p. 108347, 2022.

[7] W. Aftab and L. Mihaylova, "A Gaussian process regression approach for point target tracking," in *2019 22th International Conference on Information Fusion (FUSION)*. IEEE, 2019, pp. 1–8.

[8] ——, "A learning Gaussian process approach for maneuvering target tracking and smoothing," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 57, no. 1, pp. 278–292, 2020.

[9] B.-T. Vo, D. Clark, B.-N. Vo, and B. Ristic, "Bernoulli forward-backward smoothing for joint target detection and tracking," *IEEE Trans. Signal Process.*, vol. 59, no. 9, pp. 4473–4477, 2011.

[10] B. Ristic, B.-T. Vo, B.-N. Vo, and A. Farina, "A tutorial on Bernoulli filters: theory, implementation and applications," *IEEE Trans. Signal Process.*, vol. 61, no. 13, pp. 3406–3430, 2013.

[11] R. P. Mahler, *Statistical Multisource-Multitarget Information Fusion*. Norwood, MA, USA: Artech House, 2007.

[12] B. Ristic, L. Rosenberg, D. Y. Kim, and R. Guan, "Bernoulli filter for tracking maritime targets using point measurements with amplitude," *Signal Process.*, vol. 181, p. 107919, 2021.

[13] T. Li, Z. Liu, and Q. Pan, "Distributed bernoulli filtering for target detection and tracking based on arithmetic average fusion," *IEEE Signal Process. Lett.*, vol. 26, no. 12, pp. 1812–1816, 2019.

[14] I. Steinwart, "On the influence of the kernel on the consistency of support vector machines." *J. Mach. Learn. Res.*, 2002.

[15] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2004.

[16] T. Li, H. Chen, S. Sun, and J. M. Corchado, "Joint smoothing and tracking based on continuous-time target trajectory function fitting," *IEEE Trans. Autom. Sci. Eng.*, vol. 16, no. 3, pp. 1476–1483, 2018.

[17] T. Li and H. Fan, "A computationally efficient approach to non-cooperative target detection and tracking with almost no a-priori information," *arXiv preprint arXiv:2104.11122*, 2021.

[18] B. T. Vo, C. M. See, N. Ma, and W. T. Ng, "Multi-sensor joint detection and tracking with the Bernoulli filter," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 48, no. 2, pp. 1385–1402, 2012.

[19] A. Doucet, N. De Freitas, N. J. Gordon *et al.*, *Sequential Monte Carlo methods in practice*. New York, NY, USA: Springer, 2001.

[20] T. Li, M. Bolic, and P. M. Djuric, "Resampling methods for particle filtering: Classification, implementation, and strategies," *IEEE Signal Process. Mag.*, vol. 32, no. 3, pp. 70–86, 2015.

[21] N. Chopin, P. E. Jacob, and O. Papaspiliopoulos, "SMC2: an efficient algorithm for sequential analysis of state space models," *J. R. Stat. Soc. Ser. B-Stat. Methodol.*, vol. 75, no. 3, pp. 397–426, 2013.

[22] L. Martino, J. Read, V. Elvira, and F. Louzada, "Cooperative parallel particle filters for online model selection and applications to urban mobility," *Digit. Signal Prog.*, vol. 60, pp. 172–185, 2017.

[23] B. Ristic, S. Arulampalam, and N. Gordon, *Beyond the Kalman Filter-Particle Filters for Tracking Applications*. Norwood, MA, USA: Artech House, 2004.

[24] A. Doucet and A. M. Johansen, "A tutorial on particle filtering and smoothing : Fifteen years later," Dept. Statist., Univ. British Columbia, Tech. Rep. 2008.

[25] X. R. Li and V. P. Jilkov, "Survey of maneuvering target tracking. part i. dynamic models," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 39, no. 4, pp. 1333–1364, 2004.

[26] D. Schuhmacher, B.-T. Vo, and B.-N. Vo, "A consistent metric for performance evaluation of multi-object filters," *IEEE Trans. Signal Process.*, vol. 56, no. 8, pp. 3447–3457, 2008.