Abstract—This paper studies the optimal state estimation for a dynamic system, whose transfer function can be nonlinear and the input noise can be of arbitrary distribution. Our algorithm differs from the conventional extended Kalman filter (EKF) and the particle filter (PF) in that it estimates not only the state vector but also the Cramér-Rao bound (CRB), which serves as an accuracy indicator. Combining the state estimation, the CRB, and the incoming new measurement, the algorithm updates the state estimation according to the maximum likelihood (ML) criterion. To illustrate the effectiveness of the proposed method for autonomous driving, we apply it to estimate the position and velocity of a vehicle based on the noisy measurements of distance and Doppler offset. Simulation results show that the proposed algorithm can achieve estimation significantly more accurate than the standard EKF and the PF.

Index Terms—Bayesian filter; Nonlinear/Non-Gaussian; Doppler shift; MLE; Sequential Monte Carlo;

I. INTRODUCTION

Many scientific problems necessitate noisy measurements on the system in order to assess the status of the system over time. We will focus on discrete-time nonlinear filtering to model the system in this study, which is frequently used in applications such as adaptive analysis and non-stationary time series prediction [1].

For analyzing a dynamic system, we need require two models at least: First, the system model, which describes the development of the state in time frame; second, the measurement model, which relates the noisy measurements to the state. The probabilistic state-space formulation and the requirement for the updating of information on receipt of new measurements are suited for the Bayesian approach. This gives a generic framework for estimating dynamic states.

In a Bayesian approach to dynamic state estimation, we try to construct a posterior probability density function (pdf) of the states based on all available information. For many problems, we need to make an estimations when each time measurements are received. In this case, we can use a recursive filter to solve the problem. Such filters basically consist of two phases: prediction and update. The prediction phase uses the system model to predict the state pdf from one measurement point in the current time instant to the next. Since the state is usually subject to disturbances, the update phase uses the latest measurement to modify the predicted pdf.

The most commonly used type of state estimator is the Kalman filter [2]. It is the optimal estimator for linear systems, but few systems in the real world are linear. A common approach to overcome this problem is to linearize the system using a first-order Taylor series expansion to obtain the standard extended Kalman filter [3]. However, the standard extended Kalman filter can limit the estimation results for systems with a high degree of nonlinearity or that do not obey a Gaussian distribution [4]. In recent years, particle filter algorithm are widely used [5], [6], and it can not only solve the nonlinear and non-Gaussian problems better, but also has robustness. However, the problem of particle degradation is common in the basic particle filtering algorithm, because the variance of particle weights will keep increasing with time iteration. And degeneracy is inevitable, after several iterations, the weights of all but a few particles will be small enough to be negligible.

In dynamic estimation, Cramer-Rao Bound (CRB) is used to determine the performance limit of the recursive Bayesian estimator with uncertain target states [7]. For nonlinear filtering problems, after [8] proposed a recursive formula for calculating the lower bound of the theory, the technology is widely used in prediction [9], tracking [10], [11] and other fields. Previous studies mainly compare the algorithm results with CRB to verify the effectiveness of the algorithm. We propose a new algorithm, which uses CRB as an intermediary to realize the transmission of information, thereby completing dynamic estimation and update. As an application, we combine this algorithm with the joint estimation of target position and velocity based on distance and Doppler shift, including both Gaussian and non-Gaussian scenarios.

In this paper, a novel adaptive model-based optimal state estimation is proposed for a system whose state function can be nonlinear and whose input noise can be arbitrarily distributed. Unlike the traditional algorithm, the proposed algorithm combines the system state function with the observed quantities measured at the current time instant and uses maximum likelihood estimation to estimate the system state at the current time instant, and also calculates its corresponding CRB based on the estimation result. After that, the estimated system state and CRB are used as the mean and variance of the current time instant state, respectively, and substituted into the state function at the next time instant to complete the system state update.

We apply the proposed algorithm to an autonomous driving application scenario, where we estimate the vehicle position and velocity based on noise measurements of distance and Doppler offset. To demonstrate the effectiveness of the proposed algorithm, we first model the problem as a system state obeying a Gaussian distribution. The result shows that the estimation performance of the proposed algorithm is comparable to that of conventional estimation algorithms. After that, we modify the system model so that it no longer obeys a Gaussian distribution, and the results show that the
proposed algorithm outperforms the standard EKF and PF.

The notations in this article are as follows. Bold uppercase and lowercase letters indicate matrices and column vectors, respectively. \( L \) is n-dimensional identity matrix. \( \| A \| \) is the norm of matrix \( A \). \( A^T \) is the transpose of matrix \( A \). \( \otimes \) represents Kronecker product. \( x \propto y \) represents that \( x \) is directly proportional to \( y \). \( \mathbb{R}^n \) is an n-dimensional subspace.

II. Problem Formulation and Preliminaries

In the process of dynamic estimation, the system obtain observations by the measurement at every time instant \( t_i \). But after the measurement, the observations are discarded, which leads to the loss of information. We need to consider integrating the measurement results of completed estimates into the subsequent estimation process. Consider a general state-space model expressed by a state transition function and a observation function

\[
\begin{align*}
\theta_i &= f_i(\theta_{i-1}, \omega_i) \\
\beta_i &= h_i(\theta_i, \nu_i)
\end{align*}
\]

where \( \theta_i \in \mathbb{R}^k \) and \( \beta_i \in \mathbb{R}^p \) are respectively the state and the observation of the system at time instant \( t_i \), \( \omega_i \) and \( \nu_i \) are independent noise, \( f_i \) and \( h_i \) are nonlinear functions, which are assumed known at time instant \( t_i \) and may be time-varying.

Let us first take a brief review of the classic Kalman Filter as follows.

A. Kalman Filter

For the linear system as a special case of (1), the most widely known is Kalman filter, which solves the general problem of estimating the state \( \theta_i \) of a discrete-time system constructed by a linear stochastic difference equation

\[
\theta_i = F_i \theta_{i-1} + \omega_i
\]

with a observation \( \beta_i \) that is

\[
\beta_i = H_i \theta_i + \nu_i
\]

where similar to (1) and (2), \( F_i \), \( H_i \), \( \omega_i \) and \( \nu_i \) are respectively state transition matrix and observation matrix, \( \omega_i \sim N(0, Q_i) \) and \( \nu_i \sim N(0, R_i) \) represent the process noise and measurement noise, which all obey independent Gaussian distribution.

Kalman filter estimates the system state by using a form of feedback control, where the filter estimates the state \( \theta_i \) of the system at a time instant and then obtains feedback in the form of the noisy observation \( \beta_i \). Thus, the equations of the Kalman filter are divided into two categories: the prediction equation and the update equation. The prediction equation is responsible for predicting the current state \( \hat{\theta}_{i|i-1} \) and the error covariance \( P_{i|i-1} \) to obtain a priori estimates for the next time instant \( t_i \)

\[
\hat{\theta}_{i|i-1} = F_i \hat{\theta}_{i-1|i-1}
\]

\[
P_{i|i-1} = F_i P_{i-1|i-1} F_i^T + Q_i.
\]

The update equation is responsible for incorporating the observations into the prior estimate in order to obtain an improved posterior estimate \( \hat{\theta}_{i|i} \) and corresponding error covariance \( P_{i|i} \). The recursive formulations are

\[
\hat{\theta}_{i|i} = \hat{\theta}_{i|i-1} + K_i(\beta_i - H_i \hat{\theta}_{i|i-1})
\]

\[
P_{i|i} = (I - K_i H_i) P_{i|i-1}
\]

\[
K_i = P_{i|i-1} H_i^T (H_i P_{i|i-1} H_i^T + R_i)^{-1}
\]

But Kalman filter is theoretically optimal for linear Gaussian model. In order to solve nonlinear problems, Extended Kalman filter (EKF) is proposed. Firstly, EKF linearizes the observation function from (2) to (4) with first-order Taylor expansion. Secondly, EKF filters the signal using the Kalman filtering framework. Thus it is a suboptimal filtering.

Although for most filtering problems, Gaussian approximation (e.g., Kalman filter and extended Kalman filter methods) in state transition function can be solved well. However, for such filtering problems with multiple modes or discrete state quantities, Gaussian approximation is not suitable; and for nonlinear problems, extended Kalman filter also loses some accuracy when doing local linearization, and the stronger the nonlinearity of the model, the more it is lost. Unscented Kalman filter (UKF) can solve this problem.

UKF extracts a series of representative points from the original Gaussian distribution, including the mean point, and puts the representative points into the nonlinear equation to approximate around these points, so as to obtain better approximate results. Although UKF does not require a Jacobian matrix, it actually has the same level of complexity, and may even be a little slower to approximate the regular distribution. The difference between UKF and EKF is sometimes small.

In addition, Particle Filter (PF) based on the sequential importance sampling method is an effective alternative to obtain the solution of the posterior distribution in the Bayesian filtering problem by means of Monte Carlo approximation for the purpose of filtering or state estimation.

B. Cramer-Rao Bound

In estimation theory and statistics, Cramér Rao Bound (CRB) denotes the lower bound on the variance of an unbiased estimate \( \hat{\theta} \) of a deterministic (fixed but unknown) parameter \( \beta \), and the variance of any such estimate is at least as high as the inverse of the Fisher information matrix \( J \).

Denote \( p(\beta, \theta) \) is probability density function of \( \beta \) and \( \theta \), \( g(\beta) \) is a function of \( \beta \), which is an estimate of \( \theta \). A general CRB of the estimation is

\[
P = \text{E} \left\{ [g(\beta) - \theta][g(\beta) - \theta]^T \right\} \geq J^{-1},
\]

where \( J \) has the following form

\[
J = \text{E} \left\{ -\frac{\partial^2 \log p(\beta, \theta)}{\partial \beta \partial \theta^T} \right\}.
\]
Among all unbiased methods, the solution achieves the smallest possible mean square error and is therefore a minimum variance unbiased (MVU) estimate.

Based on CRB, we propose a novel algorithm to perform dynamic estimation, which we will introduce in the next section in detail.

III. THE PROPOSED CKF SCHEME

From Bayesian Estimation perspective, dynamic problem is to recursively calculate the degree of belief in the state \( \theta_i \) at time instant \( t_i \) with the measurements \( \beta_{1:i-1} \). Furthermore, we assume that the probability density function (pdf) \( p(\theta_0) \) of the initial state \( \theta_0 \) is available as the prior. The state can be estimated by the state transition function and the observation function mentioned above.

Since equations (1) and (2) are independent from each other, the observations \( \beta_i \) at the current time instant are not affected by observations at other time instants, i.e., the observations at different time instants are independent from each other, and the pdf \( p(\beta_i) \) is a constant value. According to Baye’s rule, we can convert the conditional pdf \( p(\theta_i|\beta_{1:i}) \) as follows.

\[
p(\theta_i|\beta_{1:i}) = \frac{p(\theta_i, \beta_{1:i})}{p(\beta_{1:i})} = \frac{p(\theta_i|\beta_{1:i-1})p(\beta_{1:i-1})}{p(\beta_i|\beta_{1:i-1})} = \frac{p(\theta_i|\beta_{1:i-1})p(\beta_i|\theta_i)}{p(\beta_i)} \propto p(\theta_i|\beta_{1:i-1})p(\beta_i|\theta_i),
\]

where the prior function is defined by the state transition function (1), the likelihood function \( p(\theta_i|\beta_{1:i-1}) \) is defined by the observation function (2). From (12), we can transform the general Bayesian filtering problem into an optimization problem with the state and observation functions.

In order to accurately estimate the system state, we will use maximum likelihood estimation (MLE) to solve the above problems. The estimation of \( \theta_i \) can be written as

\[
\hat{\theta}_i = \arg \max_{\theta_i} \{ \log p(\theta_i|\beta_{1:i-1}) + \log p(\beta_i|\theta_i) \}. \tag{13}
\]

The objective function is

\[
L_i(\theta_i) = \log p(\theta_i|\beta_{1:i-1}) + \log p(\beta_i|\theta_i). \tag{14}
\]

After obtaining the estimation \( \hat{\theta}_i \), we need to consider how to apply the estimation results of the current time instant to the state function of the next time instant. The principle of asymptotic normality proofs that an estimator of MLE will obey an approximately Gaussian distribution as the sample size gets infinitely large, where the mean of the estimator is the true parameter of the estimator, the variance of estimator is Cramer-Rao Bound \( C_i \), which is the inverse of the corresponding Fisher Matrix \( J_i \). Thus we can consider the state \( \theta_i \) as approximately following a Gaussian distribution, whose mean is the estimation \( \hat{\theta}_i \) and variance is corresponding CRB \( C_i \), i.e.,

\[
\theta_i \sim N(\hat{\theta}_i, C_i). \tag{15}
\]

In order to obtain \( C_i \), we first calculate the Fisher Matrix \( J_i \). From (14), we transform the Fisher Matrix into

\[
J_i = J_{i1} + J_{i2} = \mathbb{E} \left\{ -\frac{\partial^2 \log p(\theta_i|\beta_{1:i-1})}{\partial \theta_i \partial \theta_i^T} \right\} + \mathbb{E} \left\{ -\frac{\partial^2 \log p(\beta_i|\theta_i)}{\partial \theta_i \partial \theta_i^T} \right\}. \tag{16}
\]

With the approximated distribution of the estimated result \( \hat{\theta}_i \) of the current time instant \( t_i \), we can substitute \( \hat{\theta}_i \) into the state function to update of the system state at the next time instant \( t_{i+1} \), which will make \( \theta_{i+1} \) obeys

\[
\theta_{i+1} = f_i(\hat{\theta}_i, C_i, \omega_i). \tag{17}
\]

After the estimation, the process will be repeated with the previous a posteriori estimates used to project or predict the new a priori estimates. Therefore, the proposed algorithm recursively conditions the current estimate on all of the past measurements. The following table shows the specific tracking process.

**Algorithm 1 Nonlinear Bayesian Estimation**

1. Initialize the state \( \hat{\theta}_0 \), the variance \( \bar{R}_0 \) and the pdf \( p(\theta_0) \).
2. for \( i = 1:N \) do
3. \( \text{Obtain } p(\theta_i|\beta_{1:i-1}) \text{ from (1) and } p(\beta_i|\theta_i) \text{ from (2)} \).
4. \( \text{Estimate the state } \hat{\theta}_i \text{ at the time instant } t_i \).
5. \( \text{Output the estimation } \hat{\theta}_i \).
6. \( \text{Calculate the CRB } C_i \text{ of } \theta_i \).
7. \( \text{Let } \theta_i \sim N(\hat{\theta}_i, C_i) \).
8. \( \text{Put } \theta_i \text{ into (11) at the next time instant } t_{i+1} \).
9. end for

IV. APPLICATION OF CKF TO TARGET TRACKING

In this section, we present a collection of applications that profile the proposed algorithm. Consider a moving target with variable acceleration in a network of location-known sensors. At every time instant \( t_i \), a sensor with position information \( p_i \) measures the target to obtain the distance \( d_i \) and the Doppler Shift \( \Delta f_i \) between the sensor and the target. We will estimate the actual position \( \tilde{x}_i \) and velocity \( \tilde{y}_i \) of the target by the above information.

We denote \( \tilde{x}_i = x_i - p_i \). The relative distance \( d_i \) and the Doppler Shift \( \Delta f_i \) between the target and the sensor satisfy

\[
d_i = \| \tilde{x}_i \| + n_i, \tag{18}
\]

\[
\Delta f_i = \frac{f_c}{c} \frac{\tilde{x}_i^T \tilde{y}_i}{\| \tilde{x}_i \|} + z_i, \tag{19}
\]

where \( f_c \) is carrier frequency, \( c \) is wave velocity, \( n_i \sim N(0, \sigma_n^2) \) and \( z_i \sim N(0, \tilde{\sigma}_n^2) \) are observation noise. Since both \( f_c \) and \( c \) are constant in the same scenario, let \( \xi_i = z_i c / f_c \).

We can obtain radial velocity of the target with the sensor from (19), i.e.,

\[
v_i = \frac{c}{f_c} \Delta f_i = \frac{\tilde{x}_i^T \tilde{y}_i}{\| \tilde{x}_i \|} + \xi_i. \tag{20}
\]
For simplicity, we form \( \hat{\beta}_i = \left[ \frac{x_i}{\|x_i\|} \right]^T \) from (18) and (20). The observation is
\[
\beta_i = \hat{\beta}_i + \nu_i, \tag{21}
\]
where \( \nu_i = \left[ \eta_i \right] \sim N(0, Q_i), \ Q_i = \begin{bmatrix} \sigma_i^2 & 0 \\ 0 & (c_i^2/f)^2 \end{bmatrix}, \) i.e.,
\[
\beta_i \sim N(\hat{\beta}_i, Q_i). \tag{22}
\]

In the subsections, we will introduce the applications in two different scenarios from a simple linear state model with Gaussian noise to a non-Gaussian system.

A. Gaussian System

We integrate the variables that require estimation at time instant \( t_i \) as a state vector \( \theta_i = [\alpha_i^T \ y_i^T \ x_i^T]^T, \) which contains the acceleration \( \alpha_i \in \mathbb{R}^3, \) the velocity \( y_i \in \mathbb{R}^3 \) and the location \( x_i \in \mathbb{R}^3 \) of the target. Let the acceleration \( \alpha_i \) of target modeled by Auto-Regressive (AR) model obey the recursion:
\[
\alpha_i = \lambda_i \alpha_{i-1} + \eta_i, \tag{23}
\]
where \( \lambda_i \) is a instantaneous autoregressive coefficients at time instant \( t_i, \) \( \eta_i \in \mathbb{R}^{3 \times 1} \) is Gaussian noise with mean 0 and variance \( \Sigma_i. \) The time interval between time instants \( t_{i-1} \) and \( t_i = \Delta t_i = t_i - t_{i-1} \). We assume the acceleration of the target remains constant between two measurements. According to the principle of uniform speed linear motion, we can formulate the state model
\[
\theta_i = A_i \theta_{i-1} + \omega_i, \tag{24}
\]
where \( \omega_i = [\eta_i^T \ \alpha_i^T \ \theta_i^T]^T \sim N(0, R_i). \) The state transition matrix \( A_i \in \mathbb{R}^{9 \times 9} \) has the following form
\[
A_i = \begin{bmatrix}
\lambda_i \\
\Delta t_i/2 \\
\Delta t_i/2 \\
\end{bmatrix} \times I_3. \tag{25}
\]

After modelling the system, We substitute the proposed algorithm in Section II into this scenario, so as to estimate and update \( \theta_i. \) According to the proposed algorithm, we can approximate the state \( \theta_i \) obeys
\[
\theta_i \sim N(A_i \hat{\theta}_{i-1}, \hat{R}_i) \tag{26}
\]
where \( \hat{R}_i \) satisfies \( \hat{R}_i = A_i C_{i-1} A_i^T + R_i, \) \( \hat{\theta}_{i-1} \) is the estimation of the target state \( \theta_{i-1} \) at time instant \( t_{i-1}, \) \( C_{i-1} \) is the corresponding CRB. With the state and the observation function, we can obtain the objective function as follows based on (14), (22) and (26).

\[
L_i(\theta_i) = L_i^{(1)} + L_i^{(2)} = \frac{1}{2} \left[ \begin{array}{c}
\theta_i - \beta_i \\
\hat{\theta}_i - \hat{\beta}_i \\
\end{array} \right]^T Q_i^{-1} \left[ \begin{array}{c}
\theta_i - \beta_i \\
\hat{\theta}_i - \hat{\beta}_i \\
\end{array} \right] + \frac{1}{2} \left[ \begin{array}{c}
\hat{\theta}_i - A_i \hat{\theta}_{i-1} \\
\end{array} \right]^T \hat{R}_i^{-1} \left[ \begin{array}{c}
\hat{\theta}_i - A_i \hat{\theta}_{i-1} \\
\end{array} \right] \tag{27}
\]

In order to estimate the target state in the above MLE problem, we choose sequential convex programming (SCP) algorithm in the article. Denote the first and the second order derivatives of the objective function (27) for the estimated state as \( \nabla L_i \) and \( \nabla^2 L_i. \) Consequently, the Newton direction is
\[
u_i = -\left( \nabla^2 L_i \right)^{-1} \nabla L_i. \tag{28}
\]
The first derivative of the objective function is
\[
\nabla L_i = \frac{\partial L_i}{\partial \beta_i} + \frac{\partial L_i}{\partial \hat{\beta}_i} = -2h_i Q_i^{-1} (\beta_i - \hat{\beta}_i) + 2 \hat{R}_i^{-1} (\theta_i - A_i \hat{\theta}_{i-1}), \tag{29}
\]
where
\[
h_i = \frac{1}{\|x_i\|^2} \begin{bmatrix} 0 \\
\hat{x}_i \\
\hat{y}_i \end{bmatrix}. \tag{30}
\]

The second derivative is
\[
\nabla^2 L_i = \frac{\partial^2 L_i}{\partial \beta_i} + \frac{\partial^2 L_i}{\partial \hat{\beta}_i} = \begin{bmatrix} 0 & 0 & 0 \\
0 & 0 & 0 \\
0 & 0 & 0 \end{bmatrix} + 2 \hat{R}_i^{-1}. \tag{31}
\]
where the calculation of \( \frac{\partial^2 L_i}{\partial \beta_i} \) will be described in detail in the appendix. After calculating the Newton direction, we use the backtracking line search algorithm to obtain a step size \( s, \) so as to update \( \theta_i \leftarrow \theta_i + su_i. \) With several iterations, we will find the optimal estimation \( \hat{\theta}_i \) at time instant \( t_i \) when the objective function (27) converges.

In order to transfer the information of the current time instant \( t_i \) to the next time instant \( t_{i+1}, \) we need to calculate the corresponding CRB \( C_i \) after estimating \( \hat{\theta}_i. \) Since (22) and (26) both obey Gaussian distribution, which can be integrated into
\[
\left[ \begin{array}{c}
\theta_i \\
\hat{\theta}_i \\
\end{array} \right] \sim N(\mu(\theta_i), \Psi(\theta_i)), \tag{32}
\]
where
\[
\mu(\theta_i) = \begin{bmatrix} \hat{A}_i \hat{\theta}_{i-1} \\
\hat{\beta}_i \\
\end{bmatrix}, \quad \Psi(\theta_i) = \begin{bmatrix} \hat{R}_i & 0 \\
0 & Q_i \end{bmatrix}. \tag{33}
\]

The corresponding Fisher matrix is
\[
J_i = \frac{\partial \mu(\theta_i)^T}{\partial \theta_i} \Psi^{-1}(\theta_i) \frac{\partial \mu(\theta_i)}{\partial \theta_i}, \tag{34}
\]
where the first-order derivative of \( \mu(\theta_i) \) has the following form
\[
\frac{\partial \mu(\theta_i)}{\partial \theta_i} = \begin{bmatrix} I_3 \\
-h_i^T \end{bmatrix}. \tag{35}
\]
Because \( A_i, \hat{\theta}_{i-1} \) and \( \hat{\beta}_i \) are independent of each other, simplifying (35), the Fisher matrix is
\[
J_i = J_{i_1} + J_{i_2} = \hat{R}_i^{-1} + h_i Q_i^{-1} h_i^T. \tag{36}
\]
Inverting the Fisher matrix \( J_i, \) we obtain the CRB \( C_i, \) i.e., \( C_i = J_i^{-1}. \) According to asymptotic normality, we approximate the distribution of the state \( \theta_i \) obeys
\[
\theta_i \sim N(\hat{\theta}_i, C_i). \tag{37}
\]
Substitute $\theta_i$ from (37) into the state transition function to update the object at the next time instant $t_i+1$, which will make $\theta_{i+1}$ obeys
\[
\theta_{i+1} = A_{i+1}\theta_i + \omega_{i+1},
\]
which updates the target state for the next time instant $t_{i+1}$.

In summary, we first use the state transition function (24) and observation function (21) to estimate the state of target by maximum likelihood estimation, and then substitute the estimation result $\hat{\theta}_i$ and its corresponding CRB $C_i$ into the state transition function (38) of the next time instant, so as to complete the dynamic estimation process.

### B. Non-Gaussian System

In general driving process, the acceleration of vehicles does not always exist, but carries on the uniform motion at some time, and carries on the variable motion at some time. To be more realistic, we will modify the system. We still assume the acceleration of the target remains constant between two measurements. Compared with scenario 1, we use acceleration $a_i$ as control variable in the state function. The state function at the time instant $t_i$ can be expressed as
\[
\theta_i = \begin{bmatrix} y_i \\ x_i \end{bmatrix} = \begin{bmatrix} 1 \\ \Delta t_i \end{bmatrix} \begin{bmatrix} y_{i-1} \\ x_{i-1} \end{bmatrix} + \frac{\Delta t_i}{2} A_{i-1} a_{i-1},
\]
where $a_i, y_i, x_i \in \mathbb{R}^{3\times1}$ are respectively the acceleration, velocity and position of the estimated target. The acceleration $a_i$ obeys gaussian mixture distribution as follows
\[
a_i \sim \sum_{t=1}^{N_p} p_t N(\mu_{i,t}, \Sigma_{i,t}).
\]
where $N_p$ is the total number of mixture components, $p_t$ is the mixing coefficient of the $t$-th Gaussian component $N(\mu_{i,t}, \Sigma_{i,t})$. The mixture model parameters $p_t, \mu_{i,t}, \Sigma_{i,t}$, $t = 1, \cdots, N_p$, are assumed to be known. We make $N_p = 3$ to express the acceleration, uniformity and deceleration processes with different Gaussian distributions. From (39) and (40), we can know the estimated target state $\hat{\theta}_i$ satisfies
\[
\hat{\theta}_i \sim \sum_{t=1}^{N_p} p_t N(\hat{\rho}_{i,t}, \hat{\Sigma}_{i,t}).
\]
where
\[
\hat{\Sigma}_{i,t} = A_{i,t} C_{i-1} A_{i,t}^T + B_{i,t} \Sigma_{i-1,t} B_{i,t}^T
\]
\[
\hat{\rho}_{i,t} = A_{i,t} \hat{\theta}_{i-1} + B_{i,t} \mu_{i,t}.
\]
We can obtain the log-likelihood function of the state from (41) is
\[
\log p(\theta_i|\beta_{1:t-1}) = \log \sum_{t=1}^{3} M_i(t) e^{u_i(t)}
\]
where
\[
M_i(t) = \frac{p_t}{\sqrt{2\pi \sigma |\hat{\Sigma}_{i,t}|}}
\]
and
\[
u_i(t) = \frac{1}{2}(\theta_i - \rho_i(t))^{T}(\hat{\Sigma}_{i,t})^{-1}(\theta_i - \rho_i(t)).
\]
Comparing the two system model, we only change the state function. Since the state function and the observation function are independent of each other, i.e., the observation at the current time instant is not influenced by the target state at the previous moments, the observation function is still (21). Based on (41), (22) and (44), the objective function is
\[
L_i(\theta_i) = \log p(\theta_i|\beta_{1:t-1}) + \log p(\beta_i|\theta_i)
\]
\[
= \log \sum_{t=1}^{3} M_i(t) e^{u_i(t)} - \frac{1}{2}(\beta_i - \hat{\beta}_i)^T \Sigma_{i}^{-1}(\beta_i - \hat{\beta}_i).
\]
(47)

Note that since in the Gaussian model, both the state function and the observation function obey Gaussian distribution, the both pdfs can be combined, so that the original maximum likelihood estimation problem is simplified to do the minimization estimation of (27). In the non-Gaussian system model, the pdfs of the state function and the observation function are two different distributions, which is unable to simplify. Therefore, in contrast to the previous model, the objective function is (47) in the non-Gaussian system model. In optimization, we need maximize the estimate of (47), not minimizing.

With the objective function, we will optimize the problem. Similarly, we need to solve the first-order and second-order derivatives of the objective function in SCP algorithm. Compared with the Gaussian system model, we leave the velocity $v_i$ and position $x_i$ in the state vector $\theta_i$, removing the acceleration $a_i$. Since the observation function is still (21), whose derivative has been calculated in subsection IV-A, we can obtain the second-order of $\log p(\beta_i|\theta_i)$ is
\[
\frac{\partial^2 \log p(\beta_i|\theta_i)}{\partial \theta_i \partial \theta_i^T} = \begin{bmatrix} \frac{\partial^2 L_i(1)}{\partial v_i \partial v_i} & \frac{\partial^2 L_i(1)}{\partial v_i \partial \xi_i} \\ \frac{\partial^2 L_i(1)}{\partial \xi_i \partial v_i} & \frac{\partial^2 L_i(1)}{\partial \xi_i \partial \xi_i} \end{bmatrix}
\]
(48)
where $L_i(1)$ is same as $L(1)$ in (27), where the results of the calculation we have described in detail in the previous subsection. We only introduce the derivative of $\log p(\theta_i|\beta_{1:i-1})$ as follows. The first-order derivation of $\log p(\theta_i|\beta_{1:i-1})$ is
\[
\frac{\partial \log p(\theta_i|\beta_{1:i-1})}{\partial \theta_i} = -\sum_{t=1}^{3} M_i(t) e^{u_i(t)} (\hat{\Sigma}_{i,t})^{-1}(\theta_i - \rho_i(t))
\]
(49)

The second derivation of $\log p(\theta_i|\beta_{1:i-1})$ is
\[
\frac{\partial^2 \log p(\theta_i|\beta_{1:i-1})}{\partial \theta_i \partial \theta_i^T} = \Lambda_i - \zeta_i \zeta_i^T,
\]
(50)
where
\[
\Lambda_i = \sum_{t=1}^{3} M_i(t) e^{u_i(t)} (\hat{\Sigma}_{i,t})^{-1}(\theta_i - \rho_i(t)) \frac{(\theta_i - \rho_i(t))(\hat{\Sigma}_{i,t})^{-1} - I_3}{p(\theta_i|\beta_{1:i-1})},
\]
(51)
\[
\zeta_i = \sum_{t=1}^{3} M_i(t) e^{u_i(t)} (\hat{\Sigma}_{i,t})^{-1}(\theta_i - \rho_i(t)) p(\theta_i|\beta_{1:i-1})
\]
(52)
After obtaining the estimation \( \hat{\theta}_i \), we need to calculate the corresponding CRB. Since the observation function is consistent with the form in the Gaussian model, whose Fisher matrix \( \mathbf{J}_{i1} \) is shown in (56), we just need to calculate the CRB of the state function \( \mathbf{J}_{i1} \). From (41), we know the state function is a non-Gaussian function, which is difficult to compute the closed solution of the second order derivative of the mean in the log-likelihood function. We can obtain a numerical solution of the CRB \( \mathbf{C}_i \) by Monte Carlo method with defined formula (11). Generate samples \( \{ \theta^k_i, k = 1, \ldots, N_s \} \) satisfying (41), where \( N_s \) is the number of samples. The samples are independent identically distributed. The numerical approximation of Fisher matrix \( \mathbf{J}_{i1} \) is

\[
\mathbf{J}_{i1} = -\frac{1}{N_s} \sum_{k=1}^{N_s} \left\{ \frac{\partial^2 \log p(\theta^k_i | \theta_{i-1}^k)}{\partial \theta^k_i \partial \theta^k_i} \right\}.
\]  

(53)

Summing \( \mathbf{J}_{i1} \) in (53) and \( \mathbf{J}_{i2} \) in (56), we can obtain the whole Fisher matrix of \( \theta_i \) in the non-Gaussian model

\[
\mathbf{J}_i = \mathbf{J}_{i1} + \mathbf{J}_{i2},
\]

(54)

which is the inverse of the CRB \( \mathbf{C}_i \). On the basis of asymptotic normality, we make the distribution of the state \( \theta_i \) approximately obey

\[
\theta_i \sim \mathcal{N}(\tilde{\theta}_i, \mathbf{C}_i).
\]

(55)

Substitute \( \theta_i \) from (57) into the state transition function to update the object at the next time instant \( t_{i+1} \), which will make \( \theta_{i+1} \) obeys

\[
\theta_{i+1} = A_{i+1} \theta_i + \omega_{i+1}.
\]

(56)

Thereby we complete the update of the target state at the next time instant.

As a summary, we first make the system state model obey a Gaussian mixture model. At every time instant, the system receives an observation. Then we estimate the target state by MLE with its probability distribution, and substitute the estimation result \( \hat{\theta}_i \) and the corresponding CRB \( \mathbf{C}_i \) into the state transfer function at the next time instant, thus completing the dynamic estimation process.

V. NUMERICAL SIMULATIONS

In previous sections, we introduce a dynamic estimation algorithm and apply it to two different tracking and positioning scenarios. In this section, we will validate the result of the comparison using simulations to verify the effectiveness of the proposed algorithm.

We first compare RMSE of the estimation result with EKF and CRB in the Gaussian system model, based on the average of \( 10^3 \) Monte Carlo trials, where we set the variance of the acceleration noise in each dimension to be \( 1(m/s^2)^2 \), i.e., \( \Sigma_i = I_3 \), the variance of the noise of the relative distance is \( \sigma_i = 0.25 m^2 \), the variance of the noise of the radial phase velocity is \( \xi_i = 0.25(\text{m/s})^2 \), the AR model coefficient is \( \lambda = 0.3 \), and the time interval between two measurements is \( \Delta t_i = 0.1s \). Fig.1 shows that the estimation error of the proposed method is similar to that of EKF, and both algorithms are close to CRB.

In the non-Gaussian system model, we compare RMSE of the estimation result with EKF, Particle Filter (PF) and CRB, based on the average of \( 10^3 \) Monte Carlo trials. The acceleration follows a ternary mixed Gaussian model, in which the probability density distribution of each dimension is \( 0.3 \times N(1, 0.1) + 0.2 \times N(-1, 0.1) + 0.5 \times s \). The variance of the noise of the relative distance is \( \sigma_i = 1m^2 \). The variance of the noise of the radial phase velocity is \( \xi_i = 1(\text{m/s})^2 \). The time interval between two measurements is \( \Delta t_i = 0.1s \). The sample size of PF is 100. The sample number for CRB calculation in the proposed method is 100. Fig.2 shows that the estimation results of the three methods are similar in the estimation of target velocity, while for the estimation of target position, compared with EKF and PF, the proposed method has the highest estimation accuracy and the smallest dynamic error peak.

VI. CONCLUSION

In this paper, we propose a nonlinear Bayesian dynamic estimation algorithm, which uses the estimation result and its corresponding Cramer-Rao Bound as an intermediate propagation to perform updated estimation. We also apply the algorithm to the joint estimation of target position and velocity based on distance and Doppler shift in two different scenarios, and compare the results with the standard EKF and PF. The effectiveness of the algorithm is verified by simulations.

APPENDIX

In order to facilitate the calculation of the second derivative, we expand the first part of the objective function \( L^{(1)}_i \) into

\[
L^{(1)}_i = m_{i1} + m_{i2} = \frac{(d_i - \|\hat{x}_i\|)^2}{\sigma^2_i} + \frac{(v_i - \frac{\hat{x}_i^T \hat{x}_i}{\|\hat{x}_i\|^2})^2}{\hat{\sigma}^2_i}.
\]

(57)
Fig. 2: Comparison of the RMSE velocity and position estimation results between the proposed method, EKF and PF in Scenario 2.

Since the variables in $\mathbf{x}_i$ without $a_i$, we can just take the derivative of $x_i$ and $y_i$, which can be calculated in three steps.

Step 1: The second derivative $L_i^{(1)}$ for $x_i$ is

$$\frac{\partial^2 L_i^{(1)}}{\partial x_i \partial x_i^T} = \frac{\partial^2 m_{i_2}}{\partial x_i \partial x_i^T} + \frac{\partial^2 m_{i_2}}{\partial x_i \partial x_i^T}$$

(58)

where the second derivative of $m_{i_2}$ is

$$\frac{\partial^2 m_{i_2}}{\partial x_i \partial x_i^T} = -2 \left\{ \left( \frac{d_i}{\|x_i\|} - 1 \right) \mathbf{I}_3 - \frac{d_i \hat{x}_i x_i^T}{\|x_i\|^3} \right\}.$$  (59)

Since the second derivative of $m_{i_2}$ is complicated, we first calculate the first derivative of $m_{i_2}$ as follows.

$$\frac{\partial m_{i_2}}{\partial x_i} = \frac{2}{\sigma_i^2} \left( g_{i_2}^{(1)} + g_{i_2}^{(2)} + g_{i_2}^{(3)} + g_{i_2}^{(4)} \right)$$

$$- \frac{2}{\sigma_i^2} \left( \frac{v_i y_i}{\|x_i\|} \frac{x_i^T y_i}{\|x_i\|^2} - \frac{v_i x_i^T y_i x_i}{\|x_i\|^3} + \frac{\hat{x}_i^T y_i x_i^T}{\|x_i\|^4} \right).$$

(60)

According to (60), we need to compute the derivation of $g_{i_2}^{(1)}$, $g_{i_2}^{(2)}$, $g_{i_2}^{(3)}$ and $g_{i_2}^{(4)}$ as follows to obtain the second derivative of $m_{i_2}$,

$$\frac{\partial g_{i_2}^{(1)}}{\partial x_i} = -v_i \frac{y_i x_i^T}{\|x_i\|^3}$$

(61)

$$\frac{\partial g_{i_2}^{(2)}}{\partial x_i} = -\hat{x}_i^T y_i x_i^T + \frac{\hat{x}_i^T y_i x_i}{\|x_i\|^4} + 2 \frac{\hat{x}_i^T y_i x_i x_i^T}{\|x_i\|^4}$$

(62)

$$\frac{\partial g_{i_2}^{(3)}}{\partial x_i} = -v_i \frac{\hat{x}_i^T y_i x_i x_i^T}{\|x_i\|^3} - 3v_i \frac{\hat{x}_i^T y_i x_i x_i^T}{\|x_i\|^5}$$

(63)

$$\frac{\partial g_{i_2}^{(4)}}{\partial x_i} = 2 \frac{\hat{x}_i^T y_i x_i x_i^T}{\|x_i\|^4} + \frac{\hat{x}_i^T y_i x_i x_i^T}{\|x_i\|^4} - 4 \frac{\hat{x}_i^T y_i x_i x_i^T}{\|x_i\|^6}.$$  (64)

Based on the above, the second derivative of $m_{i_2}$ is

$$\frac{\partial^2 m_{i_2}}{\partial x_i \partial x_i^T} = -\frac{2}{\sigma_i^2} \left( \frac{\partial g_{i_2}^{(1)}}{\partial x_i^T} + \frac{\partial g_{i_2}^{(2)}}{\partial x_i^T} + \frac{\partial g_{i_2}^{(3)}}{\partial x_i^T} + \frac{\partial g_{i_2}^{(4)}}{\partial x_i^T} \right)$$

(65)

Step 2: The second derivative $L_i^{(1)}$ for $y_i$ is

$$\frac{\partial^2 L_i^{(1)}}{\partial y_i \partial y_i^T} = \frac{2}{\sigma_i^2} \left( \frac{\hat{x}_i x_i^T}{\|x_i\|^2} \right)$$

(66)

Step 3: The second derivative $L_i^{(1)}$ for $y_i$ and $x_i$ is

$$\frac{\partial^2 L_i^{(1)}}{\partial y_i \partial x_i^T} = -\frac{2}{\sigma_i^2} \left( \frac{v_i y_i}{\|x_i\|} \frac{y_i x_i^T}{\|x_i\|^2} + \frac{y_i^T \hat{x}_i x_i x_i^T}{\|x_i\|^4} \right)$$

(67)

and

$$\frac{\partial^2 L_i^{(1)}}{\partial x_i \partial x_i^T} = \left( \frac{\partial^2 L_i^{(1)}}{\partial y_i \partial x_i^T} \right)^T$$

(68)

REFERENCES

[1] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, “A tutorial on particle filters for online nonlinear/Gaussian bayesian tracking,” *IEEE Transactions on signal processing*, vol. 50, no. 2, pp. 174–188, 2002.

[2] G. Welch, G. Bishop, *et al.*, “An introduction to the Kalman filter,” 1995.

[3] M. I. Ribeiro, “Kalman and extended Kalman filters: Concept, derivation and properties,” *Institute for Systems and Robotics*, vol. 43, p. 46, 2004.

[4] Y. Huang, Y. Zhang, Z. Wu, N. Li, and J. Chambers, “A novel adaptive Kalman filter with inaccurate process and measurement noise covariance matrices,” *IEEE Transactions on Automatic Control*, vol. 63, no. 2, pp. 594–601, 2017.

[5] X. Ma, P. Karkus, D. Hsu, and W. S. Lee, “Particle filter recurrent neural networks,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, pp. 501–510, 2020.

[6] S. S. Moghaddasi and N. Faraji, “A hybrid algorithm based on particle filter and genetic algorithm for target tracking,” *Expert Systems with Applications*, vol. 147, p. 113188, 2020.

[7] H. L. V. Trees and K. L. Bell, *Bayesian bounds for parameter estimation and nonlinear filtering/tracking*. Wiley-IEEE press New York, 2007.

[8] P. Tichavsky, C. H. Muravchik, and A. Nehorai, “Posterior Cramér-Rao bounds for discrete-time nonlinear filtering,” *IEEE Transactions on signal processing*, vol. 46, no. 5, pp. 1366–1396, 1998.

[9] M. Šimandl, J. Královec, and P. Tichavský, “Filtering, predictive, and smoothing cramér–rao bounds for discrete-time nonlinear dynamic systems,” *Automatica*, vol. 37, no. 11, pp. 1703–1716, 2001.

[10] X. Zhang, P. Willett, and Y. Bar-Shalom, “Dynamic Cramer-Rao bound for target tracking in clutter,” *IEEE Transactions on Aerospace and Electronic Systems*, vol. 41, no. 4, pp. 1154–1167, 2005.

[11] M. L. Hernandez, A. Farina, and B. Ristic, “Perfil for tracking in cluttered environments: Measurement sequence conditioning approach.” *IEEE Transactions on Aerospace and Electronic systems*, vol. 42, no. 2, pp. 680–704, 2006.