information

MDPI

Article

Iterative Truncated Unscented Particle Filter

Yanbo Wang, Fasheng Wang *, Jianjun He and Fuming Sun

School of Information and Communication Engineering, Dalian Minzu University, Dalian 116620, China; viscomp@163.com (Y.W.); hejianjun@dlnu.edu.cn (J.H.); sunfuming@dlnu.edu.cn (F.S.)
* Correspondence: wangfasheng@dlnu.edu.cn

Received: 2 February 2020; Accepted: 13 April 2020; Published: 16 April 2020

Abstract: The particle filter method is a basic tool for inference on nonlinear partially observed Markov process models. Recently, it has been applied to solve constrained nonlinear filtering problems. Incorporating constraints could improve the state estimation performance compared to unconstrained state estimation. This paper introduces an iterative truncated unscented particle filter, which provides a state estimation method with inequality constraints. In this method, the proposal distribution is generated by an iterative unscented Kalman filter that is supplemented with a designed truncation method to satisfy the constraints. The detailed iterative unscented Kalman filter and truncation method is provided and incorporated into the particle filter framework. Experimental results show that the proposed algorithm is superior to other similar algorithms.

Keywords: state estimation; particle filter; iterative unscented Kalman filter; iterative truncated particle filter

1. Introduction

State estimation of a nonlinear dynamic system is of great importance in a variety of fields including computer vision, target tracking, machine learning, geophysics, bioengineering, control systems and econometrics [1–3]. The most popular approach for state estimation is the Bayesian filtering method, within which the posterior probability density function (PDF) is recursively calculated. However, it is difficult to obtain a closed-form solution for Bayesian filtering problems. Thus, researchers must search approximate solutions for these state estimation problems.

The particle filter (PF), or sequential Monte Carlo method, is an efficient solution to nonlinear filtering problems which has been applied in various fields including virtual reality [4], target tracking [5], robotics [6], econometrics [7], computer vision [8,9], etc. The basic idea of the PF is to calculate the posterior PDF using a finite set of particles and corresponding weights. The particle set defines the shape of the PDF of the state. A crucial issue of the PF is to design an appropriate proposal distribution for drawing particles. Much work has been done for designing proposal distributions. Doucet [10] used the extended Kalman filter (EKF) to build a Gaussian distribution for drawing particles. The mean and covariance estimates are obtained using the EKF prediction and update equations. As EKF is based on the linearization of nonlinear models, the estimation accuracy cannot be assured. In [10], the unscented Kalman filter (UKF) is used to build the proposal distribution, which avoids the linearization operation through propagating a set of deterministically chosen sample points. The resulting algorithm is named an unscented particle filter (UPF) which is successfully applied to visual tracking [9], and image Jacobian matrix estimation [6]. The iterated EKF is used for designing the proposal distribution in [11] providing better estimation accuracy compared to UPF. Following the idea of [11], the authors in [12] proposed to generate the proposal distribution using an iterative version of UKF (IUKF) which makes use of both statistical and analytical linearization techniques in different stages of the filtering process. In order to obtain better estimation accuracy, it is of great importance to make full use of the current observations which provide valuable information on the states. In [13], the authors combine the iterative EKF (IEKF) and the UKF to build the proposal distribution in the particle filtering framework. The proposed two-stage particle filter updates each particle sequentially using the IEKF and UKF, which makes full use of current observations.

Most of the above mentioned particle filters deal with unconstrained state estimation. The systems are described by the state and measurement equations accompanied by the PDFs of the uncertainties. However, in many practical applications, the system nonlinearity is always limited by the sub-regions of the state space when the system is subject to some constraints. Constrained state estimation arise from physical restrictions, natural phenomena, or elicited qualitative knowledge about the system of interest [14]. For example, in visual object tracking, when an object moves with moderate speed in a video sequence, its posotion is limited in the frame space. It is difficult to incorporate such constraints into the system models. In order to solve the constrained nonlinear state estimation problems, Fernandez et al. [15] devise a truncated unscented Kalman filter (TUKF) to approximate the first two moments of the posterior PDF. In [16], the authors proposed a mixture truncated unscented Kalman filter (MTUKF) which approximates the posterior PDF as a Gaussian mixture rather than a single Gaussian. However, KF-type filters may not be applicable when the state distribution becomes highly non-Gaussian due to truncated marginal probability density function.

Constrained particle filters [17] are known to provide an effective solution to constrained state estimation. Straka et al. [18] proposed to design the proposal distribution using the TUKF in the particle filter framework for generally nonlinear inequality-constrained filtering problems. Li et al. [19] proposed a novel auxiliary truncated particle filter (ATPF) for bearing-only tracking. In ATPF, the proposal distribution combines the prior PDF and a modified PDF which consider not only current observations but also spatio-temporal information. Zhang et al. [20] extends the ATPF to a multiple model particle filter for bearings-only target tracking thus exhibiting better performance. In [21], the authors present a truncated unscented particle filter which uses the TUKF to generate the proposal distribution in order to incorporate both observations and constraint information. Zhao et al. [22] proposed to impose constraints on prior particles, posterior particles and state estimates. The proposed constrained particle filter provides better physical interpretation and involves no restrictive assumptions on the distributions. In [23], the authors introduced a controlled particle filter in which the proposal distribution is generated using optimal control. The proposed method can achieve good estimation performance with fewer particles. Huang et al. [24] proposed a Rao-Blackwellized particle filter with optimal proposal distribution. This method could enforces the inequality constraints for all outcomes of the filtering distribution. It exploits the Gaussian linear sub-structure for analytic integration and a marginalization method to reduce the dimension of the state variables which ensure its computational efficiency.

In this paper, an iterative truncated particle filter is developed by using the iterative unscented Kalman filter (IUKF) to generate a proposal distribution for efficient sampling for state estimation with general inequality constraints. A truncation step is inserted into the iterated unscented Kalman filter in order to make the approximated PDF accommodate to the constraints. The proposed method is named as iterative truncated unscented particle filter (ITUPF).

The paper is organized as follows: Preliminary knowledge about constrained state estimation and particle filtering is presented in Section 2. In Section 3, the ITUPF is introduced after the details of the IUKF and the truncation method. Section 4 shows simulation results and the last section draws conclusion.

2. Preliminary Knowledge

2.1. State Estimation with Inequality Constraints

Let us consider a state-space model defined by the following equations:

$$x_k = f_k(x_{k-1}) + u_k \tag{1}$$

$$y_k = h_k(x_k) + v_k \tag{2}$$

where $x_k \in R^{n_x}$ represents the system state at time k, y_k denotes observation, f_k and h_k represent the system and measurement functions, respectively. The process noise is u_k and measurement noise is v_k. Both of them are described by known probability density functions $p(u_k)$ and $p(v_k)$. The initial state x_0 has the prior density $p(x_0)$.

The aim of Bayesian filtering is to find the state x_k based on the history of observations $Y^k = \{y_1, y_2, ..., y_k\}$ up to time k. According to the Bayesian approach, the posterior density $p(x_k|Y^k)$ provides the solution of the filtering problem.

In order to consider the information besides the system description given by (1) and (2), we use the following general constraint:

$$a_k \leq \Psi_k(x_k) \leq b_k \tag{3}$$

where Ψ_k is the constraint function at time k, and the inequality holds for all the elements of the vectors. The aim of the constrained state estimation is to find the PDF $p(x_k|Y^k)$ considering the system Equations (1) and (2), and constraint (3). We use the notation Ψ_k to represent the states that satisfy the inequality of (3), that is:

$$\Psi_k = \{x_k : a_k \leq \Psi_k(x_k) \leq b_k\} \tag{4}$$

Then, constrained state estimation is given by the constrained PDF $p(x_k|Y^k, x_k \in \Psi_k)$.

2.2. Particle Filter

In the Bayesian context, the PDF $p(x_k|Y^k)$ can be obtained through two steps, i.e., prediction and update steps. The aim of prediction step is calculating the predictive density $p(x_k|Y^{k-1})$ as follows:

$$p(x_k|Y^{k-1}) = \int p(x_k|x_{k-1}) p(x_{k-1}|Y^{k-1}) dx_{k-1} \tag{5}$$

The update step computes the required $p(x_k|Y^k)$ based on the predictive density and new observation y_k by:

$$p(x_k|Y^k) = \frac{p(y_k|x_k) p(x_k|Y^{k-1})}{\int p(y_k|x_k) p(x_k|Y^{k-1}) dx_k} \tag{6}$$

The above two equations constitute the solution of the Bayesian estimation problem, but the integrals are intractable for most nonlinear non-Gaussian models. We have to resort to approximation method to obtain a solution.

The PF method can approximate the posterior PDF using a set of particles $\{x_k^{(i)}\}_{i=1}^N$ with corresponding weights $\{w_k^{(i)}\}_{i=1}^N$ as

$$p(x_k|Y^k) = \sum_{i=1}^{N} w_k^{(i)} \delta(x_k - x_k^{(i)}). \tag{7}$$

In this equation, δ denotes the Dirac delta function and N the number of particles. The particles are intuitively drawn from the posterior PDF in order to better approximate it. However, it is impossible to directly draw particles from $p(x_k|Y^k)$. So, importance sampling is usually adopted to draw particles from a so-called importance or proposal distribution $q(x_k|x_{k-1}, y_k)$. The weight of each particle $x_k^{(i)}$ is given by:

$$w_k^{(i)} = w_{k-1}^{(i)} \frac{p(y_k|x_k^{(i)}).p(x_k^{(i)}|x_{k-1}^{(i)})}{q(x_k^{(i)}|x_{k-1}^{(i)}, y_k)} \tag{8}$$

3. Iterative Truncated Particle Filter

3.1. Iterative Truncated UKF

3.1.1. Iterative Unscented Kalman Filter

The iterative UKF is an improved UKF where the Newton-Raphson iteration is used at the update step. It is also rooted in the unscented transformation. A set of deterministically chosen sigma-points are used to compute the first two moments of a n_x-dimensional random variable x_k. The sigma-points χ_i with corresponding weights ϖ_i are selected based on the first two moments $\hat{x}_k$ and P_k as follows:

$$\chi_0 = \hat{x}_k, \varpi_0 = \frac{\lambda}{n_x + \lambda} \tag{9}$$

$$\chi_i = \hat{x}_k + (\sqrt{(n_x + \lambda)P_k})_i, \varpi_i = \frac{1}{2(n_x + \lambda)} \tag{10}$$

$$\chi_{n_x+i} = \hat{x}_k - (\sqrt{(n_x + \lambda)P_k})_i, \varpi_{n_x+i} = \frac{1}{2(n_x + \lambda)} \tag{11}$$

where $i = 1, 2, ..., n_x$, λ is the scaling parameter, $(\sqrt{(n_x + \lambda)P_k})_i$ represents the i-th column of the matrix square root of $(n_x + \lambda)P_k$.

At time $k-1$, suppose the PDF $p(x_{k-1}|Y^{k-1})$ is determined by the first two moments $\hat{x}_{k-1}$ and P_{k-1}. In order to obtain the moments estimation of time k, we first transfer the sigma-points through the system transition model.

$$\chi_{k|k-1} = f_{k-1}(\chi_{k-1}) \tag{12}$$

Then, the predicted mean is obtained based on the transferred sigma-points as follows:

$$\hat{x}_{k|k-1} = \sum_{i=0}^{2n_x} \varpi_{k-1}^{(i)} \chi_{k|k-1}^{(i)} \tag{13}$$

and the predicted covariance matrix is

$$P_{k|k-1} = \sum_{i=0}^{2n_x} \varpi_{k-1}^{(i)} [\chi_{k|k-1}^{(i)} - \hat{x}_{k|k-1}][\chi_{k|k-1}^{(i)} - \hat{x}_{k|k-1}]^T \tag{14}$$

The next step is update step in which the predicted estimates will be updated using the latest observation y_k. According to the Bayesian theorem, the PDF $p(x_k|Y^k)$ can be written as:

$$\begin{aligned} p(x_k|Y^k) &= p(x_k|y_k, Y^{k-1}) \\ &= \frac{p(y_k|x_k, Y^{k-1}).p(x_k|Y^{k-1})}{p(y_k|Y^{k-1})} \\ &= \frac{p(y_k|x_k).p(x_k|Y^{k-1})}{p(y_k|Y^{k-1})}, \end{aligned} \tag{15}$$

where the denominator $p(y_k|Y^{k-1})$ is a normalizing constant. We assume that all the PDFs are Gaussian, consequently, $p(x_k|Y^k)$ can be rewritten as:

$$
\begin{aligned}
p(x_k|Y^k) &\approx \frac{1}{c}N(y_k;h_k(x_k),v_k)\cdot N(x_k;\hat{x}_{k|k-1},P_{k|k-1}) \\
&\propto exp\{-\frac{1}{2}([y_k-h_k(x_k)]^T\cdot(M_k^v)^{-1}\cdot \\
&\quad [y_k-h_k(x_k)]+[\hat{x}_{k|k-1}-x_k]^T\cdot P_{k|k-1}^{-1}\cdot \\
&\quad [\hat{x}_{k|k-1}-x_k])\}
\end{aligned}
\tag{16}
$$

where M_k^v is the covariance matrix of the measurement noise. The minimum mean square error (MMSE) estimate is equivalent to the maximum a posteriori probability (MAP) estimate which is obtained through maximizing $p(x_k|Y^k)$. Taking logarithms on both sides of Equation (16), the following formula can be obtained:

$$
\begin{aligned}
log(p(x_k|Y^k)) = -\frac{1}{2}&([y_k-h_k(x_k)]^T\cdot(M_k^v)^{-1}\cdot \\
&[y_k-h_k(x_k)]+[\hat{x}_{k|k-1}-x_k]^T\cdot P_{k|k-1}^{-1}\cdot \\
&[\hat{x}_{k|k-1}-x_k])
\end{aligned}
\tag{17}
$$

Let $\ell(x) = -log(p(x_k|Y^k))$, consequently, maximizing $p(x_k|Y^k)$ is equivalent to minimizing $\ell(x)$. The Newton-Raphson iteration method can be used to find the minimum of $\ell(x)$ starting from $\bar{x}_0 = \hat{x}_{k|k-1}$. At iteration j, $\ell(x)$ can be expanded at $\bar{x}_{j-1}$ to a second order Taylor series approximation:

$$
\begin{aligned}
\ell(x) = \ell(\bar{x}_{j-1}) &+ (x-\bar{x}_{j-1})^T\frac{\partial\ell(\bar{x}_{j-1})}{\partial x} \\
&+\frac{1}{2}(x-\bar{x}_{j-1})^T\frac{\partial^2\ell(\bar{x}_{j-1})}{\partial x^2}(x-\bar{x}_{j-1})
\end{aligned}
\tag{18}
$$

In this equation, $\frac{\partial\ell}{\partial x}$ is the Jacobian while $\frac{\partial^2\ell}{\partial x^2}$ is the Hessian of $\ell(x)$. Differentiating Equation (18), and equating the differentiation to 0, we can obtain,

$$
\frac{\partial\ell(\bar{x}_{j-1})}{\partial x}+\frac{\partial^2\ell(\bar{x}_{j-1})}{\partial x^2}(x-\bar{x}_{j-1}) = 0
\tag{19}
$$

Then, the estimate $\bar{x}_j$ can be computed by:

$$
\bar{x}_j = \bar{x}_{j-1} - \left(\frac{\partial^2\ell(\bar{x}_{j-1})}{\partial x^2}\right)^{-1}\cdot\frac{\partial\ell(\bar{x}_{j-1})}{\partial x}
\tag{20}
$$

where the Jacobian and Hessian in this equation can be directly computed according to Equation (17) as:

$$
\begin{aligned}
\frac{\partial\ell(\bar{x}_{j-1})}{\partial x} = &P_{k|k-1}^{-1}\cdot(\bar{x}_{j-1}-\hat{x}_{k|k-1})- \\
&\left(\frac{\partial h(\bar{x}_{j-1})}{\partial x}\right)^T\cdot(M_k^v)^{-1}\cdot(y_k-h_k(\bar{x}_{j-1}))
\end{aligned}
\tag{21}
$$

$$
\frac{\partial^2\ell(\bar{x}_{j-1})}{\partial x^2} = P_{k|k-1}^{-1}+\left(\frac{\partial h(\bar{x}_{j-1})}{\partial x}\right)^T\cdot(M_k^v)^{-1}\cdot\frac{\partial h(\bar{x}_{j-1})}{\partial x},
\tag{22}
$$

where $J(\bar{x}_{j-1}) = \frac{\partial h(\bar{x}_{j-1})}{\partial x}$ is the Jacobian of $h_k(x)$ computed at $\bar{x}_{j-1}$. Here, we use a simplified form J_j. By substituting (21) and (22) into (20), the iterative formula can be obtained as follows:

$$
\begin{aligned}
\bar{x}_j = \bar{x}_{j-1} - &\left(P_{k|k-1}^{-1}+J_j^T(M_k^v)^{-1}J_j\right)^{-1}\cdot \\
&\left(P_{k|k-1}^{-1}(\bar{x}_{j-1}-\hat{x}_{k|k-1})-J_j^T(M_k^v)^{-1}(y_k-h_k(\bar{x}_{j-1}))\right)
\end{aligned}
\tag{23}
$$

The iteration number is usually empirically set as a fixed value L, i.e., $j = 1, 2, ..., L$. Thus, after all the iterations are finished, the final estimate is the value of the last iteration, i.e., $\hat{x}_k = \bar{x}_L$, and the covariance estimate is:

$$P_k = (P_{k|k-1}^{-1} + J_j^T (M_k^v)^{-1} J_j)^{-1} \tag{24}$$

3.1.2. Truncation of the PDF

Let us use $p_t(x_k|Y^k)$ to represent the truncated PDF which can be written as

$$p_t(x_k|Y^k) = \begin{cases} \dfrac{p(x_k|Y^k)}{\int_{\Psi_k} p(x_k|Y^k) dx_k}, & if \ \ x_k \in \Psi_k \\ 0, & if \ \ x_k \notin \Psi_k \end{cases} \tag{25}$$

This equation shows a closed-form description of x_k using the form of a PDF under the constraint Ψ_k. This truncated PDF will be used as the proposal distribution for drawing samples. As it is hard to find an easy-to-sample PDF, we will approximate this truncated PDF by a Gaussian PDF which is determined by its first two moments: $\hat{x}_k^t$ and P_k^t.

$$p_t(x_k|Y^k) \approx N(x_k; \hat{x}_k^t, P_k^t), \tag{26}$$

where the mean and covariance are obtained using the following formulas:

$$\hat{x}_k^t = \int x_k p_t(x_k|Y^k) dx_k \tag{27}$$

$$P_k^t = \int (x_k - \hat{x}_k^t)(x_k - \hat{x}_k^t)^T p_t(x_k|Y^k) dx_k \tag{28}$$

As the integrals in the above two equations are difficult to calculate, we can use the importance sampling based approximation method to find approximate values for $\hat{x}_k^t$ and P_k^t. Suppose a set of N_o samples $x_k^{(i)}$, $i = 1, 2, ..., N_o$, are drawn from an importance distribution $\rho(x_k)$. $\rho(x_k)$ is affected by only one condition, that is, $\rho(x_k) \neq 0$ for any $x_k \in \Psi_k$. It is intuitive that the the probability mass of $\rho(x_k)$ within the constrained region should be as close as possible to one in order to obtain higher accuracy. Among the N_o samples, N_o^t samples fall in the constrained region Ψ_k. This subset of samples is denoted as $x_k^{t,(j)}$, $j = 1, 2, ..., N_o^t$, which are used to approximate the mean and covariance of the truncated PDF as

$$\hat{x}_k^t = \frac{1}{\sum_{j=1}^{N_o^t} w_k^{(j)}} \sum_{j=1}^{N_o^t} x_k^{t,(j)} w_k^{(j)} \tag{29}$$

$$P_k^t = \frac{1}{\sum_{j=1}^{N_o^t} w_k^{(j)}} \sum_{j=1}^{N_o^t} (x_k^{t,(j)} - \hat{x}_k^t)(x_k^{t,(j)} - \hat{x}_k^t)^T w_k^{(j)} \tag{30}$$

The sample weight is $w_k^{(j)} = \frac{p(x_k^{t,(j)})}{\rho(x_k^{t,(j)})}$. The truncated PDF will be denoted as:

$$\{\hat{x}_k^t, P_k^t\} = Tr(\hat{x}_k, P_k, \Psi_k) \tag{31}$$

This truncated PDF will be used as proposal distribution in PF.

3.2. Iterative Truncated UPF

In this section, we will detail the iterative truncated unscented particle filter (ITUPF). The truncated iterative unscented Kalman filter presented in the previous section is used to generate the proposal distribution. Suppose we have obtained the particle set, at time $k-1$, $\{x_{k-1}^{(i)}, w_{k-1}^{(i)}\}_{i=1}^{N}$. The mean and covariance estimates can be computed by

$$\hat{x}_{k-1} = \frac{1}{N}\sum_{i=1}^{N} w_{k-1}^{(i)} x_{k-1}^{(i)} \tag{32}$$

$$P_{k-1} = \sum_{i=1}^{N} \left[x_{k-1}^{(i)} - \hat{x}_{k-1}\right]\left[x_{k-1}^{(i)} - \hat{x}_{k-1}\right]^{T} w_{k-1}^{(i)} \tag{33}$$

Then, the IUKF equations are used to calculate the mean and covariance estimates through the prediction and update steps (23) and (24). Afterwards, these two estimate are used to compute the constrained mean and covariance according to (31). These two estimates provide the proposal distribution for drawing particles, i.e., $N(x_k; \hat{x}_k^t, P_k^t)$. The ITUPF is summarized as in Algorithm 1. The prediction step of IUKF is denoted as $IUKF_{pred}$ and the update step is denoted as $IUKF_{upd}$.

Algorithm 1: Iterative Truncated Unscented Particle Filter

```
Step 1: Calculate mean and covariance: $\hat{x}_{k-1}$ and $P_{k-1}$ according to Equations (32) and (33);
Step 2: Calculate the first two moment estimations using the IUKF as:
    $\{\hat{x}_{k|k-1}, P_{k|k-1}\} = IUKF_{pred}(\hat{x}_{k-1}, P_{k-1})$    (34)
    $\{\hat{x}_{k}, P_{k}\} = IUKF_{upd}(\hat{x}_{k|k-1}, P_{k|k-1}, y_k)$    (35)
Step 3: Compute the constrained mean $\hat{x}_k^t$ and covariance $P_k^t$ of the truncated PDF according to (31);
Step 4: Draw particles from the truncated proposal distribution: $N(x_k; \hat{x}_k^t, P_k^t)$ until for $N$ particles the condition $x_k \in \Psi_k$ holds;
Step 5: Compute the particle weights:
    $w_k^{(i)} = \frac{p(y_k|x_k^{(i)})N(x_k^{(i)}; x_{k|k-1}, P_{k|k-1}))}{N(x_k^{(i)}; \hat{x}_k^t, P_k^t)}$    (36)
Step 6: Normalize the particle weights to ensure $\sum_{i=1}^{N} w_k^{(i)} = 1$;
Step 7: Using the particle set $\{x_k^{(i)}, w_k^{(i)}\}_{i=1}^{N}$ to calculate the state estimates, then continue with step 1;
```

4. Simulation Results

In this section, in order to demonstrate the improved performance of the proposed algorithm, we conduct three simulation experiments.

4.1. Univariate Nonstationary Growth Model I

For the first simulation, we considered a univariate nonstationary growth model (UNGM) [15]. The nonlinear system and measurement models are as follows:

$$x_k = \frac{x_{k-1}}{2} + 25\frac{x_{k-1}}{1 + x_{k-1}^2} + 8cos(1.2k) + u_k \tag{37}$$

$$y_k = \frac{x_k^3}{25} + v_k \tag{38}$$

where the system noise u_k is non-Gaussian with Gamma distribution $\gamma(3, 2)$, while the measurement noise is Gaussian with zero mean and variance 0.01. The particle number was set to $N = 200$ for all the algorithms, the total time step was $T = 60$, and 100 Monte Carlo runs were performed to calculate the root mean square error (RMSE) defined as follows:

$$RMSE = \left(\frac{1}{T} \sum_{k=1}^{T} (\hat{x}_k - x_k)^2 \right)^{1/2} \tag{39}$$

The constraint was defined as: $-25 \leq x_k \leq 25$. The proposed method was compared with the generic PF, UPF, IUPF, MCMC-UPF and TUPF. Figure 1 shows the estimated state of different particle filters in a single Monte Carlo run and the data generated in this simulation are shown in Figure 2.

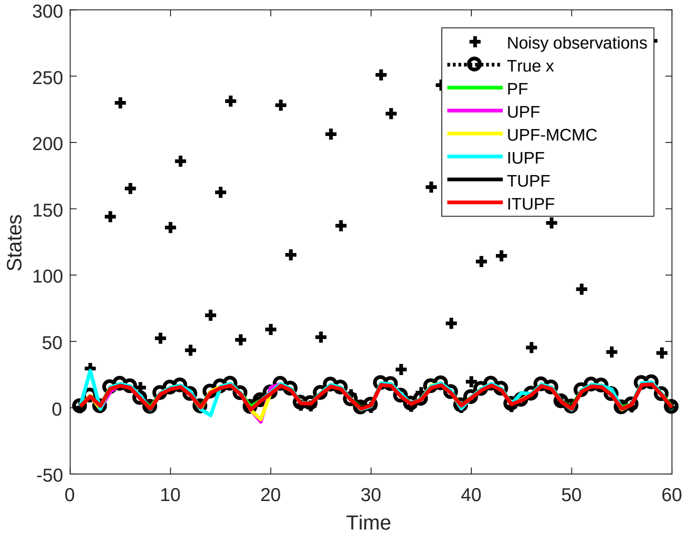

Figure 1. Estimated states of different filters and true states in model 1.

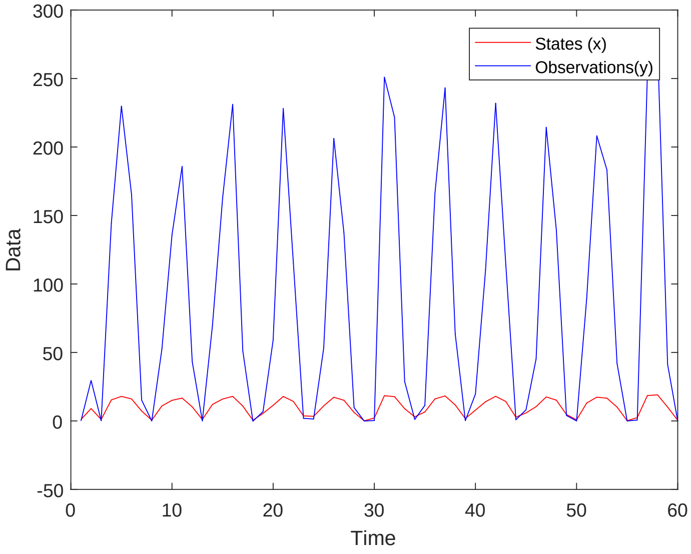

Figure 2. System data generated in a single Monte Carlo run.

Table 1 gives the means and variances of the RMSE generated by different particle filters after 100 Monte Carlo runs. It shows that UPF, IUPF and UPF-MCMC generated higher RMSE than PF and TUPF, while the ITUPF gave the lowest RMSE equal to 0.9272. Figure 3 shows the RMSEs of different algorithms after 100 Monte Carlo runs at different time steps, which clearly demonstrates that the

ITUPF gave the best and most stable estimation results compared to the other methods. The processing times of different methods are also shown in Table 1. As the iterative TUPF needed more iterations to achieve better estimation results, it needed more computational time than TUPF.

Table 1. RSME and computational time of different particle filters.

Filters	RMSE Mean	RMSE Variance	Average Computational Time
PF	1.2907	0.5968	0.3768
UPF	2.389	13.6974	0.9644
IUPF	2.3624	0.3168	1.6903
UPF-MCMC	2.4064	12.8904	2.2442
TUPF	0.9382	0.01955	0.9364
ITUPF	0.9272	0.01938	1.5063

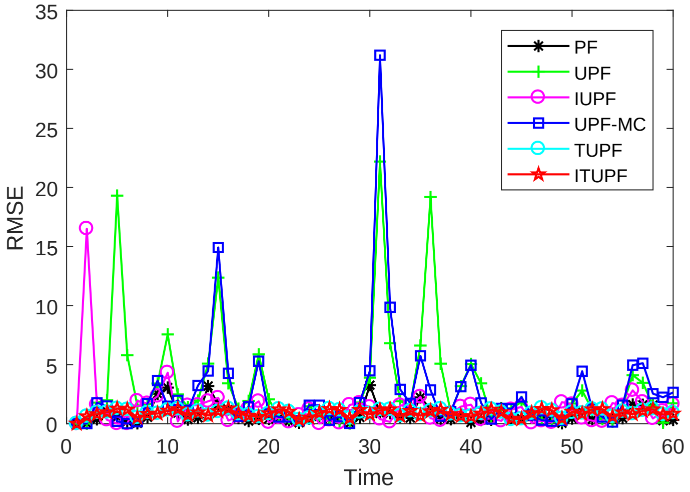

Figure 3. Root mean square error (RMSE) of different particle filters over 100 Monte Carlo runs.

4.2. Univariate Nonstationary Growth Model II

For the second simulation, we used another dynamical system described by the following model:

$$x_k = 1 + sin(\frac{1}{25}\pi(k-1)) + \frac{1}{2}x_{k-1} + u_k \tag{40}$$

$$y_k = \frac{1}{20}x_k^3 + v_k \tag{41}$$

where the system noise had the same distribution as in the previous experiment. The measurement noise was also Gaussian with zero mean and variance 0.0001. The total time step and Monte Carlo runs were the same as in the previous experiment. We used 100 particles for all the filters in this experiment. The RMSE was calculated to compare the performance of different particle filters. The constraint was defined as: $0 \leq x_k \leq 10$. Figure 4 shows the estimated and true states in a single Monte Carlo run with system data shown in Figure 5. The red curve denotes the state estimation results generated by the proposed ITUPF. Compared to the other estimation algorithms, ITUPF provided state estimates much closer to the true states.

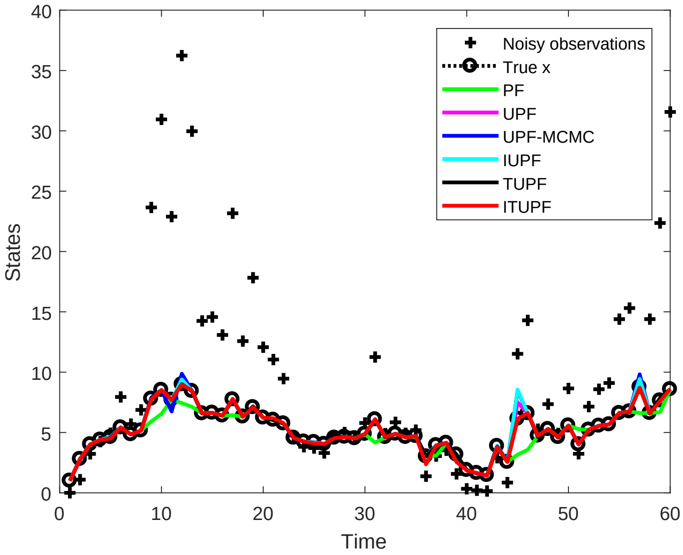

Figure 4. Estimated states of different filters and true states in model I.

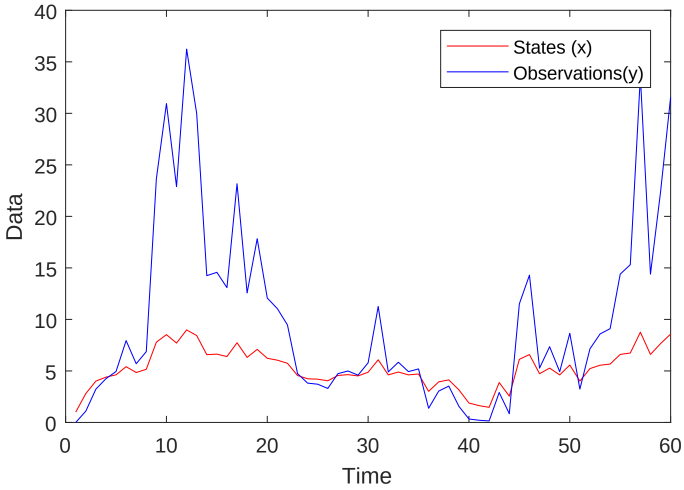

Figure 5. System data generated in a single Monte Carlo run.

The RMSE means and variances of different filters are shown in Table 2. PF gave the highest RMSE followed by UPF-MCMC, UPF, IUPF and TUPF, and ITUPF gave the lowest RMSE as expected. As for the computational time, we could draw the same conclusion as in the previous experiment.

Table 2. RMSE and computational time of different particle filters.

Filters	RMSE Mean	RMSE Variance	Average Computational Time
PF	0.6151	0.0454	0.1969
UPF	0.2957	0.0999	0.5026
IUPF	0.2884	0.0976	0.7978
UPF-MCMC	0.3014	0.1107	1.1462
TUPF	0.1240	0.00173	0.4964
ITUPF	0.1178	0.00102	0.7866

We also include the averaged RMSEs of each time step over the 100 Monte Carlo runs in Figure 6. The RMSEs of PF, UPF, IUPF and UPF-MCMC were obviously strongly fluctuating over time while TUPF and ITUPF were much more stable. The overall performance of our proposed method was better than the other filters.

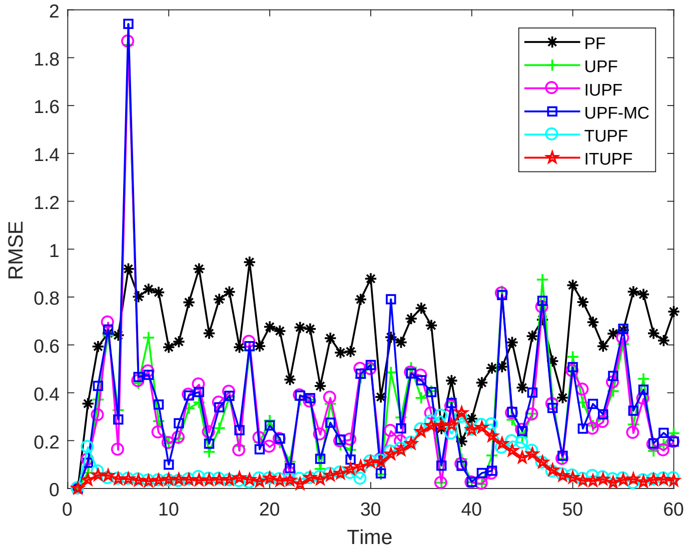

Figure 6. RMSE of different particle filters over 100 Monte Carlo runs.

4.3. Tracking an Vehicle Moving along a Circular Road

In order to further demonstrate the performance of the proposed ITUPF, we considered a problem of tracking a vehicle. The road was defined by two arcs with the radii $r_1 = 100$ and $r_2 = 96$ meters (m) while the center was at the origin of the Cartesian coordinate system. We supposed the vehicle kept angular velocity ω within $\omega \in [2.85, 5.7]$ degrees per second. In Figure 7, we show the outline of the the road and an example trajectory of the vehicle.

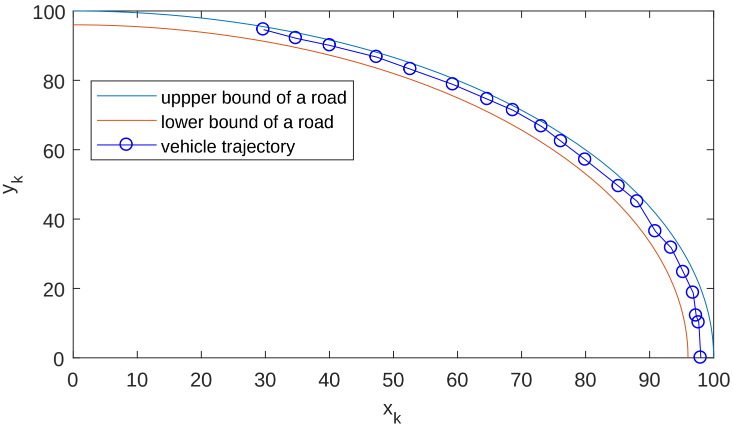

Figure 7. Circular road and an example trajectory.

The initial position of the vehicle was $[98, 0]^T$ determining the position in the x and y directions. The vehicle was supposed to follow the continuous white noise acceleration motion model. We modeled the state of the vehicle as $x_k = [x_{k,1}, x_{k,2}, x_{k,3}, x_{k,4}]^T$, where $x_{k,1}$ and $x_{k,2}$ are the positions in the x and y directions, while $x_{k,3}$ and $x_{k,4}$ represent the velocities in the x and y directions. The state evolved according to the following transition model:

$$x_k = \begin{pmatrix} 1 & T & 0 & 0 \\ 0 & 0 & 1 & T \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} x_{k-1} + \begin{pmatrix} 0.5T^2 & 0 \\ 0 & 0.5T^2 \\ T & 0 \\ 0 & T \end{pmatrix} + u_k \tag{42}$$

where $T = 1$ is the sampling period and u_k is a Gaussian zero-mean system noise with covariance matrix $Q = [1\ 0; 0\ 1]$. The vehicle traveled for $K = 20$ time steps, that is $k = 0, 1, 2, ..., 20$. A sensor was used to track the vehicle by measuring the range and bearings with sampling interval T. The measurement function was therefore defined as follows:

$$y_k = \begin{pmatrix} \sqrt{x_{k,1}^2 + x_{k,2}^2} \\ arctan(\frac{x_{k,2}}{x_{k,1}}) \end{pmatrix} + v_k \tag{43}$$

where v_k is a Gaussian zero-mean measurement noise with covariance $R = diag([8, 10^{-3}])$. When $k = 0$, the initial condition for the filters was:

$$p(x_0) = N\left(x_0 : \begin{pmatrix} 98 \\ 0 \\ 0 \\ 10 \end{pmatrix}, \begin{pmatrix} 10 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 10 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \right) \tag{44}$$

The constraint with respect to (3) is defined as $r_2 \leq \sqrt{x_{k,1}^2 + x_{k,2}^2} \leq r_1$. In this simulation experiment, we compare the TUPF and ITUPF. The particle number is set to $N = 1000$ for the two filters. The number of Monte Carlo simulations is set to $MC = 1000$. The performance of the filters is measured using the mean square error (MSE) which is defined as follows:

$$MSE = \frac{\sum_{m=1}^{MC} \sum_{k=1}^{K} \sum_{i=1}^{n_x} (x_{k,i}^{(m)} - \hat{x}_{k,i}^{(m)})^2}{MC(K+1)} \tag{45}$$

where n_x is the dimension of the state variable x_k, $x_{k,i}^{(m)}$ is the i-th component of the true state in the m-th Monte Carlo run at time step k, and $\hat{x}_{k,i}^{(m)}$ is the estimated state component. In this simulation, we only considered the position of the object when computing the mean square error (MSE).

Figure 8 shows the true trajectory and the estimated trajectories of both ITUPF and TUPF. It is clear that the trajectory estimated by ITUPF was much closer to the true trajectory than that of TUPF. The MSEs are summarized in Table 3. We can see that the MSE of TUPF was 3.6721 while that of ITUPF was 3.2655. This is caused by the fact that the proposal distribution generated by the iterative truncated UKF could accommodate the constrained region better than TUPF. The iteration operation could reduce the estimation error at the expense of higher computational load. The average computational cost of ITUPF was 7.47 which was much higher than that of TUPF.

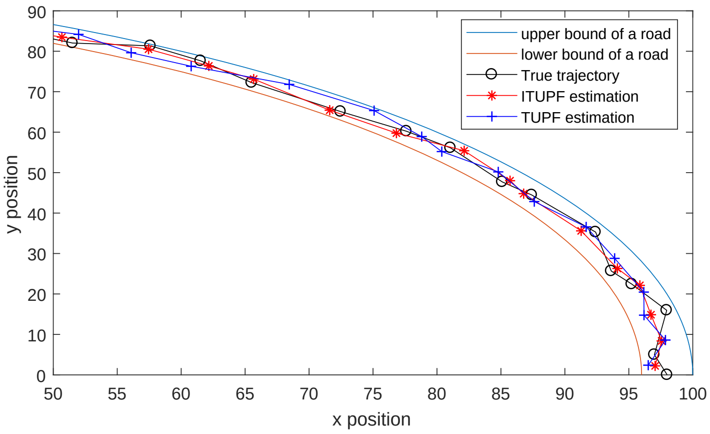

Figure 8. True trajectory and its estimates by the iterative truncated unscented particle filter (ITUPF) and truncated unscented particle filter (TUPF).

Table 3. Mean square error (MSE) and computational time of the ITUPF and TUPF.

Filters	MSE	Average Computational Time
TUPF	3.6721	4.84
ITUPF	3.2655	7.47

4.4. Particle Number Influence

In this section, we discuss the influence of particle numbers on the estimation performance. Intuitively, the performance of a particle filter directly depends on the particle number. The more particles, the better the estimation accuracy. We used Model II in Section 4.2 with different particle numbers to do this experiment. Each particle number corresponds to 100 Monte Carlo runs. The average RMSEs are calculated and shown in Figure 9. The generic PF showed an evident RMSE decrease as the particle number grew from 100 to 3000, which demonstrates that its performance highly denpended on the particle number. UPF, IUPF, and UPF-MCMC also showed a similar trend to the generic PF. However, such a trend was less evident than in the generic PF. When it comes to TUPF and proposed ITUPF, we can see from this figure that the RMSEs of the two methods decreased evidently when the particle number grew from 100 to 500. However, this downward trend became less apparent as the particle number continued to rise. The RMSEs of ITUPF was consistently lower than the other filters.

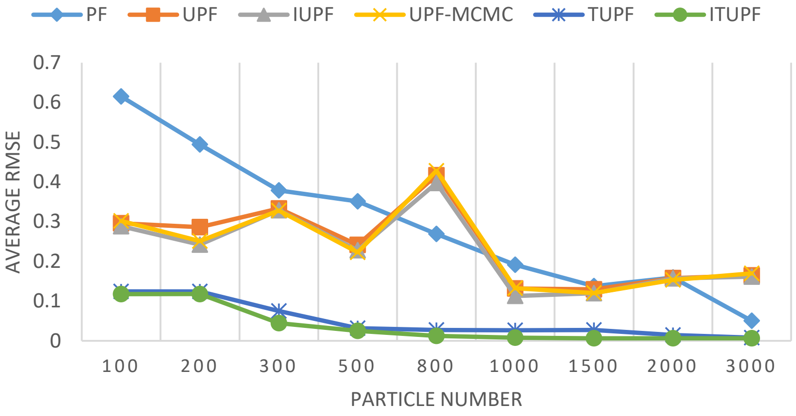

Figure 9. Particle number influence on the RMSEs of different particle filters.

4.5. Discussion

In this section, we have assessed the performance of the proposed particle filter through simulation experiments. As shown in the figures and tables, the proposed ITUPF shows improved performance compared to other similar particle filters. The iteration operation in the update step can make use of the current observation y_k at each iteration step (see Equation (23)). Meanwhile, the injection of the truncation step makes the particle filter well accommodate the inequality constraints imposed to the nonlinear systems, thus improving the performance of the truncated particle filter in constrained state estimation. In addition, simulation results also demonstrate that ITUPF is less sensitive to the variation of particle number. However, the computational cost of the ITUPF increases as the number of iteration steps L grows.

5. Conclusions

We proposed an iterative truncated unscented particle filter for constrained nonlinear state estimation. It uses a modified iterated UKF to construct the proposal distribution. A truncation step is injected into the IUKF in order to make the obtained PDF satisfy the constraints. This leads to an efficient sampling distribution which exploits the current observation and take into account inequality constraints. We use two nonlinear system models and a vehicle tracking model to carry out the simulation experiments showing that the proposed particle filter is superior to several other methods. In our future work, we will apply the ITUPF to bearing-only maneuvering target tracking and consider to reduce the computational cost of ITUPF.

Author Contributions: F.W. conceived and designed the study, wrote the manuscript. Y.W. implemented the method, reviewed and revised the manuscript. J.H. and F.S. provided technical suggestions and revised the manuscript. All authors read and approved the final manuscript.

Funding: This research was supported in part by the National Natural Science Foundation of China grant number 61972068 and 61976042, in part by the Natural Science Foundation of Liaoning Province grant number 2019-ZD-0171, in part by Dalian Science Foundation for Young Scholars under grant number 2017RQ151, in part by Program for innovative talents in Colleges and universities of Liaoning Province grant number LR2019020, and in part by Liaonig Baiqianwan Talents Program.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

PDF	Probability density function
PF	Particle Filter
EKF	Extended Kalman Filter
UKF	Unscented Kalman Filter
UPF	Unscented Particle Filter
IUKF	Iterative Unscented Kalman Filter
IEKF	Iterative Extended Kalman Filter
TUKF	Truncated Unscented Kalman Filter
MTUKF	Mixture Truncated UKF
ATPF	Auxiliary Truncated PF
ITUPF	Iterative TUPF
MSE	Mean Square Error
MCMC	Markov Chain Monte Carlo
RMSE	Root Mean Square Error

References

1. Sarkka, S. *Bayesian Filtering and Smoothing*; Cambridge University Press: Cambridge, UK, 2013.
2. Li, W.; Liu, L.; Feng, G. Cooperative control of multiple nonlinear benchmark systems perturbed by second-order moment processes. *IEEE Trans. Cybern.* **2020**, *50*, 902–910. [CrossRef] [PubMed]
3. Li, W.; Liu, L.; Feng, G. Distributed output-feedback tracking of multiple nonlinear systems with unmeasurable states. *IEEE Trans. Syst. Man Cybern.* **2019**. [CrossRef]
4. Ababsa, F.; Mallem, M. Robust camera pose tracking for augmented reality using particle filtering framework. *Mach. Vis. Appl.* **2011**, *22*, 181–195. [CrossRef]
5. Liu, C.; Li, B.; Chen, W. Particle filtering with soft state constraints for target tracking. *IEEE Trans. Aerospace Electr. Syst.* **2019**, *66*, 3492–3504. [CrossRef]
6. Wang, F.; Sun, F.; Zhang, J.; Lin, B.; Li, X. Unscented particle filter for online total image Jacobian matrix estimation in robot visual servoing. *IEEE Access* **2019**, *7*, 92020–92029. [CrossRef]
7. Lopes, H.F.; Tsay, R.S. Particle filters and Bayesian inference in financial econometrics. *J. Forecast.* **2011**, *30*, 168–209. [CrossRef]
8. Xiao, Y.; Pan, D. Research on Robust Visual Tracker Based on Multi-Cue Correlation Particle Filters. *IEEE Access* **2020**, *8*, 1–10. [CrossRef]
9. Cai, R.; Zhang, R.; Wu, Q.; Sun, H. Adaptation of unscented particle filter for visual tracking in electro-optic theodolite. In Proceedings of the 5th IET International Conference on Wireless, Mobile and Multimedia Networks (ICWMMN 2013), Beijing, China, 22–25 November 2013; pp. 181–185.
10. Van der Merwe, R.; Doucet, A.; Freitas, N.; Wan, E.A. The unscented particle filter. In Proceedings of the Neural Information Processing Systems (NIPS 2000), Denver, CO, USA, 27–30 November 2000; pp. 584–590.

11. Li, L.; Ji, H.; Luo, J. The iterated extended Kalman particle filter. In Proceedings of the International Symposium on Communications and Information Technologies (ISCIT 2005), Beijing, China, 12–14 October 2005; pp. 1213–1216.
12. Wang, F.; Li, X.; Lu, M. Improving Particle Filter with better Proposal Distribution for Nonlinear Filtering Problems. In Proceedings of the International Conference on Algorithms, Systems, and Applications of Wireless Networks (WASA 2013), Zhangjia, China, 7–10 August 2013; pp. 1–12.
13. Wang, F.; Zhang, J.; Lin, B.; Li, X. Two stage particle filter for nonlinear bayesian estimation. *IEEE Access* **2018**, *6*, 13803–13809. [CrossRef]
14. Amor, N.; Rasool, G.; Bouaynaya, N. Constrained State Estimation—A Review. *arXiv* **2018**, arXiv:1807.03463.
15. Fernandez, A.; Morelande, M.; Grajal, J. Truncated unscented kalman filter. *IEEE Trans. Signal Process.* **2012**, *60*, 3372–3386. [CrossRef]
16. Fernandez, A.; Morelande, M.; Grajal, J. Mixture truncated unscented kalman filtering. In Proceedings of the International Conference on Information Fusion (ICIF 2012), Singapore, 9–12 July 2012; pp. 479–486.
17. Amor, N.; Bouaynaya, N.; Shterenberg, R.; Chebbi, S. On the convergence of constrained particle filters. *IEEE Signal Process. Lett.* **2017**, *24*, 858–862. [CrossRef]
18. Straka, O.; Dunik, J.; Simandl, M. Truncated unscented particle filter. In Proceedings of the American Control Conference, San Francisco, CA, USA, 29 June–1 July 2011; pp. 1825–1830.
19. Li, L.; Xie, W.; Liu, Z. Auxiliary truncated particle filtering with least-square method for bearings-only maneuvering target tracking. *IEEE Trans. Aerospace Electr. Syst.* **2016**, *52*, 2562–2567. [CrossRef]
20. Zhang, H.; Li, L.; Xie, W. Constrained multiple model particle filtering for bearings-only maneuvering target tracking. *IEEE Access* **2018**, *6*, 51721–51734. [CrossRef]
21. Yu, M.; Chen, W.; Chambers, J. Truncated unscented particle filter for dealing with non-linear and inequality constraints. In Proceedings of the Sensor Signal Processing for Defence (SSPD 2014), Edinburgh, UK, 8–9 September 2014; pp. 1–5.
22. Zhao, Z.; Huang, B.; Liu, F. Constrained particle filtering methods for state estimation of nonlinear process. *AICHE J.* **2014**, *60*, 2072–2082. [CrossRef]
23. Heng, J.; Bishop, A.; Deligiannidis, G.; Doucet, A. Controlled sequential Monte Carlo. *arXiv* **2017**, arXiv:1708.08396.
24. Huang, Q. Inequality constrained state-space models. *J. Bus. Econ. Stat.* **2019**, *37*, 350–362.