entropy

MDPI

Article

Random Finite Set Based Parameter Estimation Algorithm for Identifying Stochastic Systems

Peng Wang *, **Ge Li, Yong Peng and Rusheng Ju**

College of Systems Engineering, National University of Defense Technology, Changsha 410073, China; geli@nudt.edu.cn (G.L.); yongpeng@nudt.edu.cn (Y.P.); jrscy@sina.com (R.J.)

* Correspondence: wangpeng_nudt@nudt.edu.cn; Tel.: +86-0731-8457-4584

Received: 13 June 2018; Accepted: 26 July 2018; Published: 31 July 2018

Abstract: Parameter estimation is one of the key technologies for system identification. The Bayesian parameter estimation algorithms are very important for identifying stochastic systems. In this paper, a random finite set based algorithm is proposed to overcome the disadvantages of the existing Bayesian parameter estimation algorithms. It can estimate the unknown parameters of the stochastic system which consists of a varying number of constituent elements by using the measurements disturbed by false detections, missed detections and noises. The models used for parameter estimation are constructed by using random finite set. Based on the proposed system model and measurement model, the key principles and formula derivation of the proposed algorithm are detailed. Then, the implementation of the algorithm is presented by using sequential Monte Carlo based Probability Hypothesis Density (PHD) filter and simulated tempering based importance sampling. Finally, the experiments of systematic errors estimation of multiple sensors are provided to prove the main advantages of the proposed algorithm. The sensitivity analysis is carried out to further study the mechanism of the algorithm. The experimental results verify the superiority of the proposed algorithm.

Keywords: parameter estimation; importance sampling; Probability Hypothesis Density (PHD); Random Finite Set (RFS); Markov Chain Monte Carlo (MCMC); simulated tempering

1. Introduction

In modern society, mathematical models are playing increasingly important roles in engineering systems when used for simulating, predicting, managing, and deciding [1]. Since system identification is essential for modeling, it is the potential and vital technology to meet the challenges of model based scientific research. System identification has been applied to many areas, including industrial systems, agricultural systems, biological systems, medical systems and economic systems, and even social systems [2–5].

Generally speaking, the existing system identification algorithms can be classified into non-parametric model identification algorithms and parametric model identification algorithms [6]. Parametric models are more powerful, because they can represent the studied system with fewer parameters and have more statistical power. The parametric model identification algorithms usually assume some model structure and estimate the associated parameters by reducing the error criterion function of the model and practical system. Therefore, parameter estimation is the core component of these algorithms.

1.1. Parameter Estimation

Parameter estimation refers to identifying, estimating and calibrating continuous or discrete model parameters [7]. The parameters to be estimated could be the time-invariant physical quantities, or the coefficients and other constants in a functional relationship that describe the practical physical system [8]. They can be represented by scalars, vectors, or matrixes. In this paper, we use an n elements vector $\boldsymbol{\theta} \in \Theta$ to represent the unknown model parameters.

The stochastic system is the system whose input, output and interference have random factors, or the system itself has some uncertainty. Only by parameter estimation can we get the model parameters for the stochastic system. Parameter estimation of the stochastic system is the basis of state filtering, stochastic control, fault diagnosis, forecasting and many other fields [9]. This paper deals with the identification problem of stochastic systems whose model structures are known and parameters to be identified. In this case, the identification of stochastic system minimizes the estimating of unknown parameters associated with the known models by using prior information and observations from the practical system.

The Sequential Monte Carlo (SMC) method is a powerful tool to perform optimal state and parameter estimating by using the nonlinear and non-Gaussian state space models. Since SMC based parameter estimation algorithms employ the non-parametric statistic inference based on Bayesian inference theory, and have no assumptions of the distribution and linearity of the studied system, they are widely used for parameter estimation [10–14].

According to [11], the SMC based parameter estimation algorithms can be broadly formulated into Maximum Likelihood (ML) formulation and Bayesian formulation. ML formulation is mainly related to finding a point estimate of the unknown parameter which is the most likely to be associated with the observations. The ML estimate of the parameters is the parameters that minimize the negative logarithm of the likelihood function based on the received measurements. The estimation process is shown as the following equation:

$$\hat{\boldsymbol{\theta}}_{ML} = \arg\max_{\boldsymbol{\theta}\in\Theta} p_{\boldsymbol{\theta}}(z_{1:T}) = \arg\max_{\boldsymbol{\theta}\in\Theta} \log p_{\boldsymbol{\theta}}(z_{1:T}) \tag{1}$$

where $\hat{\boldsymbol{\theta}}_{ML}$ is the estimated parameter vector, $z_{1:T}$ is the sequence of measurements, and $\boldsymbol{\theta}$ is the unknown parameter vector to be estimated.

ML method is only applicable to the estimation problem where a joint probability density function (PDF) can be assigned to the measurements [15]. In addition, ML estimate needs to use the gradient based search methods such as a damped Gauss–Newton method to actually compute estimated parameters [16].

In the Bayesian formulation, the unknown parameters $\boldsymbol{\theta}$ are regarded as an unobserved random variable or vector. The Bayesian parameter estimation gives a posterior PDF of the unknown parameters, conditioned on measurements and prior distribution. Thus, the parameter estimation problem refers to obtaining the posterior PDF of $\boldsymbol{\theta}$ by using the available measurements. According to Bayes' theory, the posterior distribution can be obtained by Equation (2), where $p_0(\boldsymbol{\theta})$ is the prior distribution.

$$p(\boldsymbol{\theta}|z_{1:T}) = \frac{p(z_{1:T}|\boldsymbol{\theta})p_0(\boldsymbol{\theta})}{p(z_{1:T})} \tag{2}$$

The first advantage of the Bayesian parameter estimation algorithm over the traditional algorithms is the conceptualization of the estimation result. In the traditional algorithms, there is a specific but unknown model that needs to be identified. In the Bayesian algorithm, the model is a random variable rather than deterministic. The estimation result is a probability distribution for the model parameters named as posterior distribution. With the probability distribution, we can answer many probabilistic questions about the model. Such questions are impossible in the traditional algorithms, because the identified model they provided is not a random variable.

The second advantage of the Bayesian parameter estimation algorithm over the traditional algorithms is the incorporation with prior information represented mathematically as a prior distribution for the model. The Bayesian parameter estimation algorithm naturally incorporates prior information about the estimation result, ranging from hard additional constraints to experience based intuition. Once measurements are collected, they are combined with the prior distribution using Bayesian inference to produce the desired posterior distribution of the unknown parameters.

Both ML and Bayesian parameter estimation can be implemented off-line or on-line. In an off-line parameter estimation algorithm, we infer the unknown parameters by iterating over a fixed measurement dataset. In an on-line method, the unknown parameters are sequentially estimated as the measurements become available. The role of SMC for both off-line and on-line parameter estimation is discussed in [10,17]. If we assume that the states of the studied system are unknown, there is a need to deal with the relationship between states and parameters. Marginalization and data augmentation are two different strategies for dealing with this kind of parameter estimation problem. Marginalization means integrating out the states from the problem and viewing the unknown parameters as the only unknown variable of interest. Usually, marginalization is implemented in an off-line manner. Data augmentation estimates the unknown parameters together with the states [11,18]. In other words, the data augmentation performs parameter estimation by extending the state with the unknown parameters and transforming the problem into an optimal filtering problem. Usually, data augmentation is implemented in an on-line manner.

1.2. Nomenclature

An introduction of some concepts used in this paper is provided in the following. More detailed information can be found in the provided references.

1.2.1. Random Finite set (RFS)

In the middle 1970s, G. Matheron systematically described the random set theory [19]. The random finite set (RFS) approach introduced by Mahler is an advanced theory that unifies much of information fusion under a single Bayesian paradigm [20,21]. The RFS ψ is defined as a random variable that draws its instantiations $\psi = Y$ from the hyperspace $\boldsymbol{X}$ of all finite subsets Y (the null set $\emptyset$ included) of some underlying space $\boldsymbol{X}_0$. Here, we assume that $\boldsymbol{X}_0 = \mathbb{R}^n$ is a Euclidean vector space and that, therefore, $\boldsymbol{X}$ consists of all finite sets of the form $Y = \emptyset$ or $Y = \{\boldsymbol{y}_1, \ldots, \boldsymbol{y}_n\}$ where $n \geq 1$ is an arbitrary positive integer and $\boldsymbol{y}_1, \ldots, \boldsymbol{y}_n$ are arbitrary vectors in $\mathbb{R}^n$.

1.2.2. Finite Set Statistics (FISST)

Here, we adopt FISST which provides the mathematical representation and manipulation of RFS. FISST is a set of tools for filtering and estimating problems involving RFS [22]. FISST enables the RFS based Bayesian inference by providing some mathematical tools. Fundamentally, it approaches the modeling of multi-object tracking by introducing sets that contain a random number of single-object states, each of which is a random vector. It allows the dynamics of each object (member of the object set) to vary according to some motion model while the number of objects (members in the set) is allowed to vary according to some point process model.

1.2.3. Probability Hypothesis Density (PHD)

The difficulty of RFS based Bayesian inference is its computational complexity. To make the RFS based estimation possible, Mahler proposed the PHD (probability hypothesis density) filter [21,23,24]. The PHD $D_{k|k}(\boldsymbol{x}|\boldsymbol{Z}_k)$ of the posterior probability PDF $f_{k|k}(\boldsymbol{X}_k|\boldsymbol{Z}_k)$ is a density function defined on the single object state $\boldsymbol{x} \in \boldsymbol{X}_0$ as follows:

$$D_{k|k}(\boldsymbol{x}|\boldsymbol{Z}_k) = \int f_{k|k}(\{\boldsymbol{x}\} \cup \boldsymbol{X}_k|\boldsymbol{Z}_k)\delta\boldsymbol{X}_k \tag{3}$$

Here, we use the abbreviation $D_{k|k}(\boldsymbol{x}) = D_{k|k}(\boldsymbol{x}|\boldsymbol{Z}_k)$. In point theory, $D_{k|k}(\boldsymbol{x})$ is defined as the intensity density. $D_{k|k}(\boldsymbol{x})$ is not a probability density, but it represents the density of expected number of points at $\boldsymbol{x}$. If $\boldsymbol{S}$ represents a region in the single object state space $\boldsymbol{X}_0$, the integral $\int_{\boldsymbol{S}} D_{k|k}(\boldsymbol{x})d\boldsymbol{x}$

represents the expected number of targets in the state space $\boldsymbol{S}$. More information about PHD can be found in [22].

1.2.4. Simulated Tempering

Simulated tempering is a stochastic optimization method that is used for improving the probability of an optimization routine converging to a global optima in highly multimodal objectives [25]. It adopts the properties of thermodynamical systems and deals with two important problems with Markov Chain Monte Carlo (MCMC) methods: one is improving exploration of multimodal distributions, and the other one is allowing estimation of the normalizing constant of the target distribution. Simulated tempering augments the Markov chain state with a discrete index variable controlling the inverse temperature of the system [26].

The objective function that we want to minimize is defined by the energy function ϕ of the system and the variables being optimized with the state $\boldsymbol{x} \in X$. An increasing schedule of K inverse temperatures $\{\beta_k\}_{k=1}^{K}$ is chosen with $0 = \beta_1 \leq \beta_2 \leq \cdots \leq \beta_K = 1$. The variables $\boldsymbol{x} \in X$ on which the target distribution P is defined are augmented with a discrete index variable $k \in \{0, \ldots, K\}$. A joint density on the target variable $\boldsymbol{x}$ and temperature index k is then defined as

$$p_{\boldsymbol{x},k}(\boldsymbol{x}, k) = \frac{1}{C} \exp(-\beta_k \phi(\boldsymbol{x}) - (1 - \beta_k)\psi(\boldsymbol{x}) + \omega_k) \quad (4)$$

where C is a constant, and $\{\omega_k\}_{k=0}^{K}$ is a set of prior weights associated with each inverse temperature value, and which can be used to help shape the marginal distribution on the temperature index k. The energy function ϕ is defined as the negative logarithm of the unnormalized target density i.e., $\phi(\boldsymbol{x}) = -\log \tilde{p}(\boldsymbol{x})$. Function ψ defines a corresponding energy function for a base distribution Q with normalized density $q(\boldsymbol{x}) = \exp(-\psi(\boldsymbol{x}))$.

If $k = 0$, the conditional distribution $P_{\boldsymbol{x}|k}$ on the target variables $\boldsymbol{x}$ corresponds to the base distribution Q and $\beta_0 = 0$. If $k = K$, it will correspond to the target distribution P and $\beta_K = 1$. We can therefore use the $\boldsymbol{x}$ components of sampled chain states for which $k = K$ to estimate the variables of interest which is related to the target distribution P. In simulated tempering, a Markov chain with invariant distribution corresponding to (4) is constructed by alternating updates of the target variables $\boldsymbol{x}$ given the current value of temperature index k, with updates of the temperature index k given the current value of the target variables $\boldsymbol{x}$. More information about simulated tempering can be found in [27].

1.3. Motivation and Advantages of the Proposed Parameter Estimation Algorithm

In this paper, we choose to use the Bayesian formulation represented by Equation (2). The unknown parameters are regarded as a random vector and associated with a suitable prior information modeled by the prior distribution. The posterior PDF of the parameters is to be characterized by using the given measurements. The proposed parameter estimation algorithm can be regarded as the Monte Carlo batch techniques [28], and it is perfect for estimating parameters of stochastic dynamic systems. The proposed parameter estimation algorithm is an off-line Bayesian parameter estimation algorithm, and it is an updated version of the marginalization based algorithms.

The conventional Bayesian parameter estimation algorithms depend on the vector based representation of data including states and measurements. The vector based representation makes these algorithms have three essential disadvantages. The first one is that they must assume that the studied system is a single permanently active system and the model is presumed. They cannot be used for estimating the dynamic system that switches on and off randomly. The transition from one mode to another is impossible. The second disadvantage is that they are based on the assumption that the detection is perfect which means no clutters and no missed detections, and they also need the number and sort order of measurements to be previously designated. The third disadvantage is that they are not appropriate for estimating the systems with varying state dimensions caused by the varying number of constituent elements contained in the studied system.

The limits of the commonly used random vector based Bayesian estimation algorithms are analyzed in [29]. We can find that the root of their disadvantages is that they must ensure the dimension and elements' order in each vector to be equal and fixed. They also need necessary operations outside of the Bayesian recursion to ensure the consistency of the vectors used for calculation. The determination of newly observed measurements and missed measurements is through vector augmentation and truncation which are very computationally intensive and irreversible. In this paper, we propose to employ the RFS theory to overcome these disadvantages of the standard vector based Bayesian parameter estimation algorithms.

An RFS has a broader application prospect than a random vector, because an RFS has a random number of constituent vectors and it allows these constituent vectors themselves to be random, distinct and unordered [30]. In practice, the number of measurements is not fixed and the ordering of measurements is irrelevant, so the measurements are better to be modeled as an RFS. RFS based data representation generalizes the system model and measurement model for parameter estimation, because it takes into account a more realistic situation where the varying number of objects, detection uncertainty, clutters, missed detections and data association uncertainty are all taken into consideration.

Unlike the traditional parameter estimation algorithms and the vector based Bayesian parameter estimation algorithms, the proposed Bayesian parameter estimation algorithm employs RFS based modeling, and has the following advantages:

1. It can estimate the parameters by using the imperfect detections including noises, clutters, data association uncertainty and missed detections. Thus, it enables the measurement model used for parameter estimation to be more universal and consistent with the practical system.
2. It can estimate the parameters for the stochastic systems which consist of a varying number of constituent components or objects. It can take the object birth, object death and spawned objects of the studied system into consideration.
3. The proposed algorithm can also accurately estimate the states of the studied system and the number of objects contained in the studied system while estimating the unknown parameters. The estimation of states and object number depends on the estimated parameters.

The rest of the paper is structured as follows. We present the RFS based modeling for both measurement model and system model in Section 2. A detailed description of the PHD based Bayesian inference and the SMC based implementation are given in Section 3. The proposed RFS based parameter estimation algorithm is detailed in Section 4. After describing the proposed algorithm, in Section 5, we take a particular problem of estimating systematic errors of multiple sensors as an example to verify this algorithm. In order to better promote academic exchanges, we provide the source codes for the experiments in the supplementary materials part. Finally, the conclusion is provided in Section 6.

2. RFS Based Modeling

The RFS based parameter estimation algorithm uses two kinds of models to describe the real world: one is the Markov transition density based system model; the other one is the measurement likelihood based measurement model. System model is used to model the state transition of the studied system, and the measurement model is used to model the practical measurement system. Here, we focus on estimating the static parameter vector $\boldsymbol{\theta}$ which features in the state space models that represent system dynamics and measurement system using observations from the practical system. The RFS based representation of measurements and states is the foundation of RFS based algorithm. In this section, we give the RFS based problem formulation and the RFS based models used in the proposed parameter estimation algorithm.

2.1. Problem Formulation

Suppose at time step $k = 0, 1, 2, \ldots$, the studied system consists of n_k objects. Their states can, respectively, be represented by the vectors $\boldsymbol{x}_{k,1}, \ldots, \boldsymbol{x}_{k,n_k}$, and each state vector is obtained from the state space $\mathscr{X} \subseteq \mathbb{R}^{n_x}$. In practice, the number n_k and the individual states in $\mathscr{X}$ of system's constituent elements are both time varying and random. In this paper, the system state at k is described by an RFS $\boldsymbol{X}_k = \{\boldsymbol{x}_{k,1}, \ldots, \boldsymbol{x}_{k,n_k}\} \in \mathscr{F}(\mathscr{X})$. The state evolution of the RFS based system model can be described by the Markov system state process conditioned on the static parameters $\boldsymbol{\theta} \in \Theta$. The state evolution is characterized by a prior density $\boldsymbol{X}_0 \sim \mu(\boldsymbol{X}_0|\boldsymbol{\theta})$ and a multiple objects transition density $\Pi_{k|k-1}(\boldsymbol{X}_k|\boldsymbol{X}_{k-1}, \boldsymbol{\theta})$.

The measurement process is imperfect in the sense that there exists detection uncertainty. The measurement devices usually create false detections. If observation datasets are available at time $k = 1, 2, \ldots$, and there are m_k elements in the observation dataset at time k, and the observations take a values in the measurement space $\mathscr{Z} \subseteq \mathbb{R}^{n_z}$. Then, the observations can be modeled by an RFS $\boldsymbol{Z}_k = \{z_{k,1}, \ldots, z_{k,m_k}\} \in \mathscr{F}(\mathscr{Z})$. Here, we assume that the measurement process is conditionally independent from the system's state evolution, and can be fully described by the likelihood $\varphi_k(\boldsymbol{Z}_k|\boldsymbol{X}_k, \boldsymbol{\theta})$ which is conditioned on $\boldsymbol{\theta} \in \Theta$.

We assign a prior distribution to the unknown parameter vector $\boldsymbol{\theta}$, and estimate it on the parameter vector space according to Bayesian theory. The parameter estimation problem in this paper mainly refers to estimating the posterior PDF $p(\boldsymbol{\theta}|\boldsymbol{Z}_{1:K})$; here, $\boldsymbol{Z}_{1:K} = \boldsymbol{Z}_1, \ldots, \boldsymbol{Z}_K$ is the measurement set sequence gathered up to time index K. If the prior density $p_0(\boldsymbol{\theta})$ is given, according to Equation (2), the estimation result in the Bayesian approach is given by:

$$p(\boldsymbol{\theta}|\boldsymbol{Z}_{1:K}) = \frac{f(\boldsymbol{Z}_{1:K}|\boldsymbol{\theta})p_0(\boldsymbol{\theta})}{f(\boldsymbol{Z}_{1:K})} \propto f(\boldsymbol{Z}_{1:K}|\boldsymbol{\theta})p_0(\boldsymbol{\theta}) \tag{5}$$

The computation of $f(\boldsymbol{Z}_{1:K}|\boldsymbol{\theta})$ is quite difficult, because it refers to performing estimation on the joint space of the unknown system states history $\boldsymbol{X}_{1:K}$ and the unknown parameter vector $\boldsymbol{\theta}$. To solve this problem, we choose to consider a broader problem instead of solving the problem directly. The parameter estimation problem can be regarded as a subproblem of the chosen problem where the main task is to obtain the following posterior PDF :

$$f(\boldsymbol{\theta}, \boldsymbol{X}_{1:K}|\boldsymbol{Z}_{1:K}) \propto f(\boldsymbol{X}_{1:K}|\boldsymbol{Z}_{1:K}, \boldsymbol{\theta})f(\boldsymbol{Z}_{1:K}|\boldsymbol{\theta})p_0(\boldsymbol{\theta}) \tag{6}$$

where $f(\boldsymbol{X}_{1:K}|\boldsymbol{Z}_{1:K}, \boldsymbol{\theta})$ is the posterior PDF of the system states conditioned on $\boldsymbol{\theta}$ and all the given observation datasets.

2.2. RFS Based Measurement Model

For the given predicted system state $\boldsymbol{X}_k = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n\}$, the random measurement set collected by the sensor can be represented by $\boldsymbol{Z}_k = \boldsymbol{\Gamma}_k(\boldsymbol{X}_k) \cup \boldsymbol{C}_k$, where $\boldsymbol{\Gamma}_k(\boldsymbol{X}_k)$ is the object detection set and it has the form $\boldsymbol{\Gamma}_k(\boldsymbol{X}_k) = \boldsymbol{\Gamma}_k(\boldsymbol{x}_1) \cup \cdots \cup \boldsymbol{\Gamma}_k(\boldsymbol{x}_n)$; $\boldsymbol{\Gamma}_k(\boldsymbol{x}_j)$ is the detection set for state $\boldsymbol{x}_j$; and $\boldsymbol{C}_k$ is the set of Poisson clutters.

Any given object in the studied system can generate either a single measurement or no measurement at all. Consequently, $\boldsymbol{\Gamma}_k(\boldsymbol{x}_i)$ can be modeled as $\boldsymbol{\Gamma}_k(\boldsymbol{x}_i) = \emptyset^{p_D(\boldsymbol{x}_i|\boldsymbol{\theta})} \cap \{\boldsymbol{Z}_i\}$. Here, $\boldsymbol{Z}_i = h(\boldsymbol{x}_i, \boldsymbol{W}_i|\boldsymbol{\theta})$ is the sensor-noise model associated with ith state $\boldsymbol{x}_i$; and $\emptyset^{p_D(\boldsymbol{x}_i|\boldsymbol{\theta})}$ is the discrete random subset of the baseline measurement space $\mathscr{Z}_0$. Detection uncertainty is modeled as a Bernoulli RFS as follows:

$$Pr(\emptyset^p = \boldsymbol{T}) = \begin{cases} 1 - p_D(\boldsymbol{x}|\boldsymbol{\theta}), & \text{if } \boldsymbol{T} = \emptyset \\ p_D(\boldsymbol{x}|\boldsymbol{\theta}) \cdot g(\boldsymbol{Z}_i|\boldsymbol{\theta}), & \text{if } \boldsymbol{T} = \mathscr{Z}_0 \\ 0, & \text{otherwise} \end{cases} \tag{7}$$

where $p_D(\boldsymbol{x}|\boldsymbol{\theta})$ is the probability that a single object whose state is $\boldsymbol{x}$ gives a single observation, and $1 - p_D(\boldsymbol{x}|\boldsymbol{\theta})$ is the probability that the sensor generates no measurement at all.

The fact that the clutter process is Poisson means that the clutters can be represented by the set $\boldsymbol{C} = \{\boldsymbol{c}_1, \ldots, \boldsymbol{c}_M\}$, where M is a random nonnegative integer with probability distribution $p_M(m) = e^{-\lambda}\lambda^m/m!$; $\boldsymbol{c}_1, \ldots, \boldsymbol{c}_M$ are independent, and identically distributed random measurement vectors with density $c(\boldsymbol{z}) = g_C(\boldsymbol{z}|\boldsymbol{\theta})$.

For the state set $\boldsymbol{X}_k$, the PDF of receiving the measurement set $\boldsymbol{Z}_k$ is described by the true likelihood function $\varphi_k(\boldsymbol{Z}_k|\boldsymbol{X}_k,\boldsymbol{\theta})$, which describes the sensor measurement for the studied system and characterizes the missed detections, clutters, and object generated observations. As described in [29], the true likelihood function can be calculated as follows:

$$\varphi_k(\boldsymbol{Z}_k|\boldsymbol{X}_k,\boldsymbol{\theta}) = \varphi_k((\boldsymbol{\Gamma}_k(\boldsymbol{X}_k) \cup \boldsymbol{C}_k)|\boldsymbol{X}_k,\boldsymbol{\theta}) = \sum_{W \subseteq \boldsymbol{Z}_k} p_{\boldsymbol{\Gamma}_k(\boldsymbol{X}_k)}(W|\boldsymbol{X}_k,\boldsymbol{\theta}) p_{\boldsymbol{C}_k|\boldsymbol{\theta}}(\boldsymbol{Z}_k - W) \tag{8}$$

$$= e^{\lambda} \cdot g_C(\boldsymbol{Z}_k|\boldsymbol{\theta}) \cdot g_k(\emptyset|\boldsymbol{X}_k,\boldsymbol{\theta}) \cdot \sum_{\xi} \prod_{i:\xi(i)>0} \frac{p_D(\boldsymbol{x}_i|\boldsymbol{\theta}) \cdot g_k(z_{\xi(i)}|\boldsymbol{x}_i,\boldsymbol{\theta})}{(1 - p_D(\boldsymbol{x}_i|\boldsymbol{\theta})) \cdot \lambda \cdot c(z_{\xi(i)}|\boldsymbol{\theta})} \tag{9}$$

where $g_C(\boldsymbol{Z}_k|\boldsymbol{\theta}) = e^{-\lambda} \prod_{z \in \boldsymbol{Z}_k} \lambda c(z|\boldsymbol{\theta})$ and $g_k(\emptyset|\boldsymbol{X}_k,\boldsymbol{\theta}) = e^{-\lambda} \prod_{\boldsymbol{x} \in \boldsymbol{X}_k} (1 - p_D(\boldsymbol{x}|\boldsymbol{\theta}))$; the summation is taken over all associated hypothesis $\xi : \{1, \ldots, n\} \longrightarrow \{0, 1, \ldots, m\}$. In time, the clutter process is Poisson distributed, and the mean value is λ. In space, the clutter process is distributed according to an arbitrary density $c(z)$, and $c_{k|k}(\boldsymbol{Z}_k|\boldsymbol{\theta}) = e^{-\lambda}\lambda^m_{k|k} c_{k|k}(z_1) \cdots c_{k|k}(z_m)$ is the probability that a set $\boldsymbol{Z}_k = \{z_1, \ldots, z_m\}$ of clutters will be generated. $g_{k+1}(z|\boldsymbol{x},\boldsymbol{\theta}) = g_{W_{k+1}}(z - h_{k+1}(\boldsymbol{x}|\boldsymbol{\theta}))$ is the likelihood function of a single sensor.

2.3. RFS Based System Model

Some systems may consist of multiple objects, and the number of these objects can be constantly varying, just as the objects appear and disappear from the system. Existing objects can give rise to new objects through spawning. Objects can likewise leave the system, as when disappearing, or they can be damaged or destroyed. All of these instances of multiple object dynamics should be explicitly modeled. If multiple objects contained in the studied system are related to birth, death, spawning, merging, and so on, the standard vector based estimation algorithms are inapplicable, because they fail to accurately estimate the number of objects in the studied system.

We model the time evolution of the system which consists of multiple objects by employing RFS. We take the object birth, death, spawn, and motion into consideration. The system Markov density $\Pi_{k|k-1}(\boldsymbol{X}_k|\boldsymbol{X}_{k-1},\boldsymbol{\theta})$ characterizes the time evolution of the system states. Here, we give the true system Markov density function $\Pi_{k|k-1}(\boldsymbol{X}_k|\boldsymbol{X}_{k-1},\boldsymbol{\theta})$ for the proposed system model, where $\boldsymbol{X}_k = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{n_k}\}$ and $\boldsymbol{X}_{k-1} = \{\boldsymbol{x}'_1, \ldots, \boldsymbol{x}'_{n_{k-1}}\}$. The RFS based representation of system' states has the following mathematical form

$$\boldsymbol{X}_k = \boldsymbol{S}_{k|k-1}(\boldsymbol{X}_{k-1}) \cup \boldsymbol{B}_{k|k-1}(\boldsymbol{X}_{k-1}) \cup \boldsymbol{B}_k \tag{10}$$

where $\boldsymbol{X}_k$ is the set of predicted system states, $\boldsymbol{S}_{k|k-1}(\boldsymbol{X}_{k-1})$ is a set of states of persisting objects that are continuing to exist in the system, $\boldsymbol{B}_{k|k-1}(\boldsymbol{X}_{k-1})$ represents the state set of spawned elements, $\boldsymbol{B}_k$ represents the state set of spontaneous objects. $\boldsymbol{S}_{k|k-1}(\boldsymbol{X}_{k-1})$ has the form $\boldsymbol{S}_{k|k-1}(\boldsymbol{X}_{k-1}) = \boldsymbol{S}_{k|k-1}(\boldsymbol{x}'_1) \cup \cdots \cup \boldsymbol{S}_{k|k-1}(\boldsymbol{x}'_{n_{k-1}})$, and $\boldsymbol{S}_{k|k-1}(\boldsymbol{x}'_i)$ is the set of predicted states evolved from $\boldsymbol{x}'_i$. $\boldsymbol{B}_{k|k-1}(\boldsymbol{X}_{k-1})$ has the form $\boldsymbol{B}_{k|k-1}(\boldsymbol{X}_{k-1}) = \boldsymbol{B}_{k|k-1}(\boldsymbol{x}'_1) \cup \cdots \cup \boldsymbol{B}_{k|k-1}(x_{n_{k-1}})$, and $\boldsymbol{B}_{k|k-1}(\boldsymbol{x}'_i)$ is the set of spawned states by the object whose previous state is $\boldsymbol{x}'_i$.

As detailed in [29], the RFS based system model is identical in mathematical form to the RFS based measurement model. The corresponding true Markov density is

$$\Pi_{k|k-1}(\boldsymbol{X}_k|\boldsymbol{X}_{k-1},\boldsymbol{\theta}) = \Pi_{k|k-1}((\boldsymbol{S}_{k|k-1}\cup\boldsymbol{B}_{k|k-1}\cup\boldsymbol{B}_k)|\boldsymbol{X}_{k-1},\boldsymbol{\theta}) \tag{11}$$

$$= \sum_{W_1\cap W_2=\emptyset, W_1\cup W_2\subseteq \boldsymbol{X}_{k-1}} p_{\boldsymbol{S}_{k|k-1}}(W_1|\boldsymbol{X}_{k-1},\boldsymbol{\theta})p_{\boldsymbol{B}_{k|k-1}}(W_2|\boldsymbol{X}_{k-1}|\boldsymbol{\theta})\cdot p_{\boldsymbol{B}_k}(\boldsymbol{X}_k - W_1 - W_2|\boldsymbol{\theta}) \tag{12}$$

$$= e^{\mu(\boldsymbol{X}_{k-1}|\boldsymbol{\theta})} f_{\boldsymbol{B}_{k|k-1}}(\boldsymbol{X}_k|\boldsymbol{\theta}) f_{k|k-1}(\emptyset|\boldsymbol{X}_{k-1},\boldsymbol{\theta}) \sum_{\xi}\prod_{\xi(i)>0}\frac{p_s(\boldsymbol{x}'_i|\boldsymbol{\theta})\pi_{k|k-1}(\boldsymbol{x}_{\xi(i)}|\boldsymbol{x}'_i,\boldsymbol{\theta})}{(1-p_s(\boldsymbol{x}'_i|\boldsymbol{\theta}))\mu(\boldsymbol{X}_{k-1})b(\boldsymbol{x}_{\xi(i)}|\boldsymbol{X}_{k-1},\boldsymbol{\theta})} \tag{13}$$

where $\pi_{k|k-1}(\boldsymbol{x}_k|\boldsymbol{x}_{k-1},\boldsymbol{\theta})$ is a Markov transition density, which corresponds to a single object system model $\boldsymbol{x}_k = \pi(\boldsymbol{x}_{k-1},\boldsymbol{V}_{k-1}|\boldsymbol{\theta})$. It is the likelihood that a single object will have state vector $\boldsymbol{x}_k$ at time step k if it had state vector $\boldsymbol{x}_{k-1}$ at time step $k-1$. $p_s(\boldsymbol{x}_k|\boldsymbol{\theta}) \overset{abbr.}{=} p_s^{k|k-1}(\boldsymbol{x}_{k-1}|\boldsymbol{\theta})$ denotes the probability of object's surviving into time step k. $b(\boldsymbol{B}_{k|k-1}(\boldsymbol{X}_{k-1})|\boldsymbol{x}_{k-1},\boldsymbol{\theta})$ is the probabilistic distribution of spawning a set $\boldsymbol{B}_{k|k-1}(\boldsymbol{X}_{k-1})$ of new objects at time step k. The summation is taken over all association hypotheses $\xi : \{1,\ldots,n_{k-1}\} \longrightarrow \{0,1,\ldots,n_k\}$.

The related definitions in Equation (13) are as follows:

$$f_{k|k-1}(\emptyset|\boldsymbol{X}_{k-1},\boldsymbol{\theta}) = e^{-\mu(\boldsymbol{X}_{k-1}|\boldsymbol{\theta})}\prod_{\boldsymbol{x}_{k-1}\in\boldsymbol{X}_{k-1}}(1-p_s(\boldsymbol{x}_{k-1}|\boldsymbol{\theta})) \tag{14}$$

$$\mu(\boldsymbol{X}_{k-1}|\boldsymbol{\theta}) = \mu_0 + \mu_1(\boldsymbol{x}'|\boldsymbol{\theta}) + \cdots + \mu_{n_{k-1}}(\boldsymbol{x}'|\boldsymbol{\theta}) \tag{15}$$

$$b(\boldsymbol{x}|\boldsymbol{X}_{k-1},\boldsymbol{\theta}) = \frac{\mu_0 b_0(\boldsymbol{x}|\boldsymbol{\theta}) + \mu_1(\boldsymbol{x}'|\boldsymbol{\theta})b(\boldsymbol{x}|\boldsymbol{x}'_1,\boldsymbol{\theta}) + \cdots + \mu_{n_{k-1}}(\boldsymbol{x}'|\boldsymbol{\theta})b(\boldsymbol{x}|\boldsymbol{x}'_{n_{k-1}},\boldsymbol{\theta})}{\mu_0 + \mu_1(\boldsymbol{x}'|\boldsymbol{\theta}) + \cdots + \mu_{n_{k-1}}(\boldsymbol{x}'|\boldsymbol{\theta})} \tag{16}$$

Here, μ_0 is the expected number of spontaneously generated new objects and $b_0(\boldsymbol{x}|\boldsymbol{\theta})$ is their physical distribution. In addition, $\mu(\boldsymbol{x}'_i|\boldsymbol{\theta})$ is the expected number of new objects spawned by an object whose previous state is $\boldsymbol{x}'_i$, and $b(\boldsymbol{x}|\boldsymbol{x}'_i,\boldsymbol{\theta})$ is their physical distribution.

3. PHD Filter

3.1. PHD Based Bayesian Equations

The Bayesian estimator sequentially determines the posterior PDF of the system states $f_{k|k} = (\boldsymbol{X}_k|\boldsymbol{Z}_{1:k},\boldsymbol{\theta})$ at each time step k, where $\boldsymbol{Z}_{1:k} \equiv \boldsymbol{Z}_1,\ldots,\boldsymbol{Z}_k$ denotes the gathered measurement set sequence up to time step k. The posterior PDF is calculated through the prediction and the correction recursion sequentially. On the assumption that the posterior PDF $f_{k-1|k-1}(\boldsymbol{X}_{k-1}|\boldsymbol{Z}_{1:k-1},\boldsymbol{\theta})$ at time $k-1$ is already known, if the set of measurements $\boldsymbol{Z}_k$ which is related to time k has been given, the prediction step and corrected step can be described as in [22]:

$$f_{k|k-1}(\boldsymbol{X}_k|\boldsymbol{Z}_{1:k-1},\boldsymbol{\theta}) = \int \Pi_{k|k-1}(\boldsymbol{X}_k|\boldsymbol{X}_{k-1},\boldsymbol{\theta})\cdot f_{k-1|k-1}(\boldsymbol{X}_{k-1}|\boldsymbol{Z}_{1:k-1},\boldsymbol{\theta})\delta\boldsymbol{X}_{k-1} \tag{17}$$

$$f_{k|k}(\boldsymbol{X}_k|\boldsymbol{Z}_{1:k},\boldsymbol{\theta}) = \frac{\varphi(\boldsymbol{Z}_k|\boldsymbol{X}_k,\boldsymbol{\theta})\cdot f_{k|k-1}(\boldsymbol{X}_k|\boldsymbol{Z}_{1:k-1},\boldsymbol{\theta})}{\int \varphi(\boldsymbol{Z}_k|\boldsymbol{X},\boldsymbol{\theta})\cdot f_{k|k-1}(\boldsymbol{X}|\boldsymbol{Z}_{1:k-1},\boldsymbol{\theta})\delta\boldsymbol{X}} \tag{18}$$

The PHD filter is an approximation of the Bayesian estimator described by Equations (17) and (18) [31]. Rather than propagating the FISST PDF $f_{k|k}(\boldsymbol{X}_k|\boldsymbol{Z}_{1:k},\boldsymbol{\theta})$, we can only propagate PHD $D_{k|k}(\boldsymbol{x}|\boldsymbol{Z}_{1:k},\boldsymbol{\theta})$, defined by Equation (3).

The prediction equation of the PHD filter can be represented as follows [21]:

$$D_{k|k-1}(\boldsymbol{x}|\boldsymbol{\theta}) = \gamma_{k|k-1}(\boldsymbol{x}|\boldsymbol{\theta}) + p_s\int \pi_{k|k-1}(\boldsymbol{x}|\boldsymbol{x}',\boldsymbol{\theta})\cdot D_{k-1|k-1}(\boldsymbol{x}'|\boldsymbol{\theta})d\boldsymbol{x}' \tag{19}$$

where $\gamma_{k|k-1}(\boldsymbol{x}|\boldsymbol{\theta})$ represents the PHD related to the object birth RFS between step $k-1$ and k. Once the observation dataset $\boldsymbol{Z}_k$ for time k is available, the correction equation of the PHD filter can be represented as follows [21]:

$$D_{k|k}(\boldsymbol{x}|\boldsymbol{\theta}) = (1 - p_D) \cdot D_{k|k-1}(\boldsymbol{x}|\boldsymbol{\theta}) + \sum_{z \in \mathbf{Z}_k} \frac{p_D \cdot g_k(z|\boldsymbol{x}, \boldsymbol{\theta}) \cdot D_{k|k-1}(\boldsymbol{x}|\boldsymbol{\theta})}{\kappa_k(z|\boldsymbol{\theta}) + p_D \int g_k(z|\boldsymbol{x}, \boldsymbol{\theta}) \cdot D_{k|k-1}(\boldsymbol{x}|\boldsymbol{\theta}) d\boldsymbol{x}} \tag{20}$$

where $\kappa_k(z|\boldsymbol{\theta})$ is the PHD related to the clutter RFS at step k. Here, the clutters are characterized by the Poisson RFS. The PHD of the Poisson RFS is $\kappa_k(z|\boldsymbol{\theta}) = \lambda c(z|\boldsymbol{\theta})$; here, λ denotes the mean number at step k, and $c(z|\boldsymbol{\theta})$ denotes the spatial distribution of clutters in the measurement space.

3.2. SMC based Implementation

Beginning with Sidenbladh [32] and Zajic [33], many researchers have studied how to implement the PHD filter by using the SMC method. By using sequential importance sampling to each terms of Equation (19), the particle approximation of Equation (19) can be obtained. After choosing the importance densities $p_k(\cdot|\mathbf{Z}_k, \boldsymbol{\theta})$ and $q_k(\cdot|\boldsymbol{x}_{k-1}, \mathbf{Z}_k, \boldsymbol{\theta})$ for spontaneous birth and persisting objects, Equation (19) can be rewritten as

$$D_{k|k-1}(\boldsymbol{x}_k|\boldsymbol{\theta}) = \frac{\gamma_{k|k-1}(\boldsymbol{x}_k|\boldsymbol{\theta})}{p_k(\boldsymbol{x}_k|\mathbf{Z}_k, \boldsymbol{\theta})} p_k(\boldsymbol{x}_k|\mathbf{Z}_k, \boldsymbol{\theta}) + \sum_{i=1}^{L_{k-1}} \omega_{k-1}^{(i)} \frac{p_s \cdot \pi_{k|k-1}(\boldsymbol{x}_k|\boldsymbol{x}_{k-1}^{(i)}, \boldsymbol{\theta})}{q_k(\boldsymbol{x}_k|\boldsymbol{x}_{k-1}^{(i)}, \mathbf{Z}_k, \boldsymbol{\theta})} q_k(\boldsymbol{x}_k|\boldsymbol{x}_{k-1}^{(i)}, \mathbf{Z}_k, \boldsymbol{\theta}) \tag{21}$$

Therefore, we can get the particle approximation of $D_{k|k-1}(\boldsymbol{x}|\boldsymbol{\theta})$ as follows

$$D_{k|k-1}(\boldsymbol{x}|\boldsymbol{\theta}) = \sum_{i=1}^{L_{k-1}+J_k} \omega_{k|k-1}^{(i)} \delta_{\boldsymbol{x}_{k|k-1}^{(i)}}(\boldsymbol{x}) \tag{22}$$

where

$$\boldsymbol{x}_{k|k-1}^{(i)} \sim \begin{cases} q_k(\cdot|\boldsymbol{x}_{k-1}^{(i)}, \mathbf{Z}_k, \boldsymbol{\theta}), i = 1, \ldots, L_{k-1} \\ p_k(\cdot|\mathbf{Z}_k, \boldsymbol{\theta}), i = L_{k-1} + 1, \ldots, L_{k-1} + J_k \end{cases} \tag{23}$$

$$\omega_{k|k-1}^{(i)} = \begin{cases} \frac{p_s \pi_{k|k-1}(\boldsymbol{x}_{k|k-1}^{(i)}|\boldsymbol{x}_{k-1}^{(i)}, \boldsymbol{\theta})}{q_k(\boldsymbol{x}_{k|k-1}^{(i)}|\boldsymbol{x}_{k-1}^{(i)}, \mathbf{Z}_k, \boldsymbol{\theta})} \omega_{k-1}^{(i)}, i = 1, \ldots, L_{k-1} \\ \frac{\gamma_{k|k-1}(\boldsymbol{x}_{k|k-1}^{(i)}|\boldsymbol{\theta})}{J_k p_k(\boldsymbol{x}_{k|k-1}^{(i)}|\mathbf{Z}_k, \boldsymbol{\theta})}, i = L_{k-1} + 1, \ldots, L_{k-1} + J_k \end{cases} \tag{24}$$

For the correction step of the PHD filter, the predicted PHD can be represented by $\{\omega_{k|k-1}^{(i)}, \boldsymbol{x}_k^{(i)}\}_{i=1}^{L_{k-1}+J_k}$ after prediction step, where $\boldsymbol{x}_k^{(i)} = \boldsymbol{x}_{k|k-1}^{(i)}$. Applying the correction Equation (20), we can get the following particle approximation

$$D_{k|k}(\boldsymbol{x}|\boldsymbol{\theta}) = \sum_{i=1}^{L_{k-1}+J_k} \omega_k^{(i)} \delta_{\boldsymbol{x}_k^{(i)}}(\boldsymbol{x}) \tag{25}$$

where

$$\omega_k^{(i)} = (1 - p_D)\omega_{k|k-1}^{(i)} + \sum_{z \in \mathbf{Z}_k} \frac{p_D \cdot g_k(z|\boldsymbol{x}_k^{(i)}, \boldsymbol{\theta}) \cdot \omega_{k|k-1}^{(i)}}{\kappa_k(z|\boldsymbol{\theta}) + \sum_{j=1}^{L_{k-1}+J_k} p_D \cdot g_k(z|\boldsymbol{x}_k^{(j)}, \boldsymbol{\theta}) \cdot \omega_{k|k-1}^{(j)}} \tag{26}$$

By updating the weights of the particles in term of Equation (26), the correction step maps the predicted particle system $\{\omega_{k|k-1}^{(i)}, \boldsymbol{x}_k^{(i)}\}_{i=1}^{L_{k-1}+J_k}$ into the corrected particle system $\{\omega_k^{(i)}, \boldsymbol{x}_k^{(i)}\}_{i=1}^{L_{k-1}+J_k}$.

4. RFS based Parameter Estimation Algorithm

The system states conditioned on $\boldsymbol{\theta}$, are estimated by using the Bayesian theory. The main task is to obtain an estimate of PDF $f(\mathbf{X}_{1:K}|\mathbf{Z}_{1:K}, \boldsymbol{\theta})$ for each possible value of $\boldsymbol{\theta}$ by running a PHD filter implemented by SMC. By this way, we can evaluate the measurement likelihood $f(\mathbf{Z}_{1:K}|\boldsymbol{\theta})$, and then we can indirectly obtain a practical solution for $p(\boldsymbol{\theta}|\mathbf{Z}_{1:K})$.

4.1. Formula Derivation

By employing PHD to approximate Equation (6), we can estimate $f(\boldsymbol{X}_{1:K}|\boldsymbol{Z}_{1:K},\boldsymbol{\theta})$ which depends on RFS based modeling. According to the relationship between the PHD and the RFS based PDF detailed in [22], Equation (6) can be rewritten as the following equation

$$D(\boldsymbol{\theta},\boldsymbol{x}_{1:K}|\boldsymbol{Z}_{1:K}) \propto D(\boldsymbol{x}_{1:K}|\boldsymbol{Z}_{1:K},\boldsymbol{\theta}) \cdot f(\boldsymbol{Z}_{1:K}|\boldsymbol{\theta}) \cdot p_0(\boldsymbol{\theta}) \tag{27}$$

The advantage of Equation (27) is that the posterior PHD $D(\boldsymbol{x}_{1:K}|\boldsymbol{Z}_{1:K},\boldsymbol{\theta})$ is defined on the standard vector based state space $\mathscr{X}$, so the calculation of the RFS based posterior PDF $f(\boldsymbol{X}_{1:K}|\boldsymbol{Z}_{1:K},\boldsymbol{\theta})$ is quite easy.

The observed data likelihood $f(\boldsymbol{Z}_{1:K}|\boldsymbol{\theta})$ of the accumulated sequence of observation dataset which features in Equations (5) and (27) can be calculated by using the following equation:

$$f(\boldsymbol{Z}_{1:K}|\boldsymbol{\theta}) = \prod_{k=1}^{K} f(\boldsymbol{Z}_k|\boldsymbol{Z}_{1:k-1},\boldsymbol{\theta}) \tag{28}$$

Now, the problem is how to calculate the measurement likelihood $f(\boldsymbol{Z}_k|\boldsymbol{Z}_{1:k-1},\boldsymbol{\theta})$ using the results of PHD filter. The equations and SMC based implementation of the PHD filter for recursively estimating $D_{k|k}(\boldsymbol{x}|\boldsymbol{Z}_{1:k},\boldsymbol{\theta})$ have been introduced in Sections 3.1 and 3.2. The measurement likelihood $f(\boldsymbol{Z}_k|\boldsymbol{Z}_{1:k-1},\boldsymbol{\theta})$, which is defined as in [21], can be calculated as follows

$$f(\boldsymbol{Z}_k|\boldsymbol{Z}_{1:k-1},\boldsymbol{\theta}) = \int \varphi(\boldsymbol{Z}_k|\boldsymbol{X}_k,\boldsymbol{\theta}) \cdot \Pi(\boldsymbol{X}_k|\boldsymbol{Z}_{1:k-1},\boldsymbol{\theta})\delta\boldsymbol{X}_k \tag{29}$$

From Equations (28) and (29), we can find the tight relationship between the parameter estimation problem and the state estimation problem. The key challenge is how to deal with the unknown system states. The PHD filter implemented by the SMC method is used to solve this problem. Thus, $f(\boldsymbol{Z}_k|\boldsymbol{Z}_{1:k-1},\boldsymbol{\theta})$ can be obtained by using PHD filer, and can be calculated as follows:

$$\begin{aligned} f(\boldsymbol{Z}_k|\boldsymbol{Z}_{1:k-1},\boldsymbol{\theta}) = {} & \alpha e^{-\int p_D(\boldsymbol{x}_k|\boldsymbol{\theta}) \cdot D_{k|k-1}(\boldsymbol{x}_k|\boldsymbol{Z}_{1:k-1},\boldsymbol{\theta})d\boldsymbol{x}_k} \\ & \times \prod_{z\in\boldsymbol{Z}_k} \left(\kappa_k(z|\boldsymbol{\theta}) + \int p_D(\boldsymbol{x}_k|\boldsymbol{\theta}) \cdot g(z|\boldsymbol{x}_k,\boldsymbol{\theta}) \cdot D_{k|k-1}(\boldsymbol{x}_k|\boldsymbol{Z}_{1:k-1},\boldsymbol{\theta})d\boldsymbol{x}_k\right) \end{aligned} \tag{30}$$

where α is a proportionality constant.

As described in Section 3.2, we can use the weighted particle system $\{\boldsymbol{x}_k^{(i)},\omega_k^{(i)}\}_{i=1}^{N}$ to represent the posterior PHD $D_{k|k}(\boldsymbol{x}|\boldsymbol{Z}_{1:k},\boldsymbol{\theta})$. To estimate the posterior $p(\boldsymbol{\theta}|\boldsymbol{Z}_{1:K})$, we should firstly obtain the approximation of $f(\boldsymbol{Z}_k|\boldsymbol{Z}_{1:k-1},\boldsymbol{\theta})$ given by Equation (30) by running the PHD filter. According to Equation (22), the predicted PHD $D_{k|k-1}(\boldsymbol{x}|\boldsymbol{Z}_{1:k-1},\boldsymbol{\theta})$ is approximated by the following particles

$$\widehat{D}_{k|k-1}(\boldsymbol{x}|\boldsymbol{Z}_{1:k-1},\boldsymbol{\theta}) = \sum_{i=1}^{N_k} \omega_{k|k-1}^{(i)} \delta_{\boldsymbol{x}_{k|k-1}^{(i)}}(\boldsymbol{x}) \tag{31}$$

Substituting Equation (31) into Equation (30), we can obtain the following result:

$$\begin{aligned} \widehat{f}(\boldsymbol{Z}_k|\boldsymbol{Z}_{1:k-1},\boldsymbol{\theta}) = {} & \alpha \exp\left\{-\sum_{i=1}^{N_k} p_D(\boldsymbol{x}_{k|k-1}^{(i)}|\boldsymbol{\theta})\omega_{k|k-1}^{(i)}\right\} \\ & \times \prod_{z\in\boldsymbol{Z}_k} \left(\kappa_k(z|\boldsymbol{\theta}) + \sum_{i=1}^{N_k} p_D(\boldsymbol{x}_{k|k-1}^{(i)}|\boldsymbol{\theta}) \cdot g(z|\boldsymbol{x}_{k|k-1}^{(i)},\boldsymbol{\theta})\omega_{k|k-1}^{(i)}\right) \end{aligned} \tag{32}$$

Therefore, according to Equation (28), $f(\boldsymbol{Z}_{1:K}|\boldsymbol{\theta})$ can be calculated as follows:

$$\widehat{f}(\boldsymbol{Z}_{1:K}|\boldsymbol{\theta}) = \alpha^K \exp\left\{-\sum_{k=1}^{K}\sum_{j=1}^{N_k} p_D(\boldsymbol{x}_{k|k-1}^{(j)}|\boldsymbol{\theta})\omega_{k|k-1}^{(j)}\right\}$$
$$\times \prod_{k=1}^{K}\prod_{z\in\boldsymbol{Z}_k}\left(\kappa_k(\boldsymbol{z}|\boldsymbol{\theta}) + \sum_{j=1}^{N_k} p_D(\boldsymbol{x}_{k|k-1}^{(j)}|\boldsymbol{\theta})\cdot g(\boldsymbol{z}|\boldsymbol{x}_{k|k-1}^{(j)},\boldsymbol{\theta})\omega_{k|k-1}^{(j)}\right) \tag{33}$$

4.2. Simulated Tempering Based Importance Sampling

We have obtained the estimate of $f(\boldsymbol{Z}_{1:K}|\boldsymbol{\theta})$ by running the PHD filter, the following step is to compute $p(\boldsymbol{\theta}|\boldsymbol{Z}_{1:K})$ according to Equation (5). Since the analytic solution for $p(\boldsymbol{\theta}|\boldsymbol{Z}_{1:K})$ is difficult to compute, we solve Equation (5) by using the simulated tempering based importance sampling method.

Since we are interested in the values of the unknown parameters, we can compute them as the posterior mean by using the following equation:

$$E(\boldsymbol{\theta}|\boldsymbol{Z}_{1:K}) = \int \boldsymbol{\theta} p(\boldsymbol{\theta}|\boldsymbol{Z}_{1:K})d\boldsymbol{\theta} \tag{34}$$

To avoid doing integral calculus, we propose to approximate Equation (34) via importance sampling. Here, we assume the sample size is M, we draw a sample $\boldsymbol{\theta}_1,\ldots,\boldsymbol{\theta}_M$ from the importance density ξ. Thus, the integral in Equation (34) can be approximated by the following equation:

$$\widehat{\boldsymbol{\theta}} = \sum_{j=1}^{M} \ell_j \boldsymbol{\theta}_j \tag{35}$$

The sample weights $\ell_j \geq 0, (j = 1,\ldots,M)$ are defined as:

$$\ell_j = \frac{\tilde{\ell}_j}{\sum_{m=1}^{M}\tilde{\ell}_m}, \text{ with } \tilde{\ell}_j = \frac{f(\boldsymbol{Z}_{1:K}|\boldsymbol{\theta}_j)p_0(\boldsymbol{\theta}_j)}{\xi(\boldsymbol{\theta}_j)} \tag{36}$$

According to Equation (36), we find that α can be omitted, since it cancels out if we normalize the weights. Thus, for convenience, we assume $\alpha = 1$. The determination of sample weight ℓ_j depends on the selection of importance density $\xi(\boldsymbol{\theta}_j)$. A good selection of importance density should be proportional to the posterior PDF $f(\boldsymbol{Z}_{1:K}|\boldsymbol{\theta}_j)$ and produce sample weights with a small variance.

Since we can only obtain an estimation of $f(\boldsymbol{Z}_{1:K}|\boldsymbol{\theta}_j)/\alpha^K$ after running the SMC based PHD filter, the selection of the importance density ξ is quite important. If the sample size $M \to \infty$, the approximation Equation (35) will be quite accurate for many importance densities. However, if M is finite, the approximation accuracy will depend greatly on the specific selection of importance density ξ. As shown in Equation (36), the accuracy of importance sampling by using the prior information is not satisfactory for any possible sample size, since the posterior PDF $p(\boldsymbol{\theta}|\boldsymbol{Z}_{1:K})$ which is proportional to $f(\boldsymbol{Z}_{1:K}|\boldsymbol{\theta})p_0(\boldsymbol{\theta})$ is unknown. We propose a simulated tempering [2] based importance sampling method to obtain a better importance density ξ which is similar to the posterior PDF $p(\boldsymbol{\theta}|\boldsymbol{Z}_{1:K})$.

Here, we use S to represent the maximum stages number and ϑ_s, $s = 1,\ldots,S$, represent the proposal density for the s-th stage. The main idea of tempering is to sequentially generate a serious of importance densities from which we can sequentially draw samples. The first importance density usually resembles the prior density $p_0(\boldsymbol{\theta})$, and the final importance density is the posterior PDF $p(\boldsymbol{\theta}|\boldsymbol{Z}_{1:K})$. The sequential importance density in the sequence should have very small difference. We obtain $\vartheta_S = p(\boldsymbol{\theta}|\boldsymbol{Z}_{1:K})$ at the final stage S . A sequence of importance densities that begin with the prior density and increasingly resemble the posterior PDF can be generated by the following equation, for $s = 1,\ldots,S$

$$\vartheta_s(\boldsymbol{\theta}) \propto [f(\boldsymbol{Z}_{1:K}|\boldsymbol{\theta})]^{\Lambda_s} p_0(\boldsymbol{\theta}) \tag{37}$$

where $\Lambda_s = \sum_{j=1}^{s} \beta_j$ with $\beta_j \in (0,1]$ and at the final stage we have $\Lambda_S = 1$. Thus, Λ_s increases with the growth of s and its upper bound is 1.

During the tempering process, we should sequentially draw samples from importance densities $\vartheta_1, \vartheta_2, \ldots, \vartheta_S$. At the first stage ($s = 1$), we draw the sample $\{\boldsymbol{\theta}_j^1\}_{j=1}^{M}$ from the prior density $\vartheta_0(\boldsymbol{\theta}) = p_0(\boldsymbol{\theta})$. At the s-th stage, we use the sample $\{\boldsymbol{\theta}_j^{s-1}\}_{j=1}^{M}$ drawn from ϑ_{s-1} at the $(s-1)$th stage to obtain a sample from ϑ_s. Firstly, we need to compute weights from particles in $\{\boldsymbol{\theta}_j^{s-1}\}_{j=1}^{M}$ by using the equation $\tilde{\ell}_j^s = f(\boldsymbol{Z}_{1:K}|\boldsymbol{\theta}_j^{s-1})^{\beta_s}$ for $j = 1, \ldots, M$. Then, we normalize the weights.

Resampling is quite necessary for tempering, because it can help to remove the particles with lower weights from the sample $\{\boldsymbol{\theta}_j^{s-1}\}_{j=1}^{M}$ and improve the diversity of the remaining particles. Thus, the second step is resampling. Resampling means selecting M indices $i_1^s, \ldots, i_M^s$ with probability $Pr\{i_j^s = j\} = \ell_j^s$.

After resampling, the particles $\{\tilde{\boldsymbol{\theta}}_j^s\}_{1 \le j \le M}$ will almost surely have repeated elements, because the weights $\ell_1^s, \ldots, \ell_M^s$ will be the most possible to be uneven. Here, the MCMC (Markov Chain Monte Carlo) [34] is employed to the sample $\{\tilde{\boldsymbol{\theta}}_j^s\}_{1 \le j \le M}$ that have been resampled. MCMC can help to remove the duplicate particles, thus the diversity of the sample will be increased. For each particle of $\{\tilde{\boldsymbol{\theta}}_j^s\}_{1 \le j \le M}$, a new sample particle can be drawn as follows

$$\boldsymbol{\theta}_j^{s,*} \sim g_s(\cdot|\tilde{\boldsymbol{\theta}}_j^s) \tag{38}$$

where g_s represents the importance density for the s-th stage. The decision of accepting or rejecting $\boldsymbol{\theta}_j^{s,*}$ with certain probability is made by using the Metropolis–Hastings algorithm. If the MCMC move is accepted, $\boldsymbol{\theta}_j^s = \boldsymbol{\theta}_j^{s,*}$. If the move is rejected, $\boldsymbol{\theta}_j^s = \tilde{\boldsymbol{\theta}}_j^s$. Thus, we can obtain a new sample $\{\boldsymbol{\theta}_j^s\}_{j=1}^{M}$. The probability of accepting the MCMC move is determined by the following equation

$$\eta = \min\left\{1, \frac{\vartheta_s(\boldsymbol{\theta}_j^{s,*}) g_s(\tilde{\boldsymbol{\theta}}_j^s|\boldsymbol{\theta}_j^{s,*})}{\vartheta_s(\tilde{\boldsymbol{\theta}}_j^s) g_s(\boldsymbol{\theta}_j^{s,*}|\tilde{\boldsymbol{\theta}}_j^s)}\right\} \tag{39}$$

To diversify the sample, the importance density g_s should generate the candidate particles over a fairly large area of the parameter space. We should ensure that g_s is still within the area where the likelihood is high. We can select the importance density by the following method as introduced in [35]:

$$g_s(\boldsymbol{\theta}^*|\boldsymbol{\theta}) = g_s(\boldsymbol{\theta}^*) = \mathscr{N}(\boldsymbol{\theta}^*; \hat{\boldsymbol{\mu}}_s, \hat{\boldsymbol{C}}_s) \tag{40}$$

where $\mathscr{N}(\boldsymbol{\theta}; \boldsymbol{\mu}, \boldsymbol{C})$ denotes a Gaussian distribution; $\hat{\boldsymbol{\mu}}_s$ and $\hat{\boldsymbol{C}}_s$ are respectively the mean and covariance matrix of the weighted particles $\{\ell_j^s, \boldsymbol{\theta}_j^{s-1}\}_{j=1}^{M}$.

The pseudo-code of the proposed parameter estimation algorithm is shown in Algorithm 1. Here, $1 \le H \le M$ and $\phi > 0$ are parameters that should be defined by users.

Algorithm 1: Random Finite Set based Parameter Estimation Algorithm

```
Input: Accumulated measurement sets $\mathbf{Z}_{1:K}$; prior information $p_0(\boldsymbol{\theta})$
Output: Sample $\{\boldsymbol{\theta}^S_{j=1}\}^M_{j=1}$ and maximum a posteriori estimate $\widehat{\boldsymbol{\theta}}$
Initialization:
Initialize $s = 0$, and $\Lambda_0 = 0$;
for $j = 1, \ldots, M$ do
    Draw $\boldsymbol{\theta}^0_j \sim p_0(\boldsymbol{\theta})$;
    Execute the PHD filter with $\boldsymbol{\theta}^0_j$ to estimate $\psi^0_j = \log f(\mathbf{Z}_{1:K}|\boldsymbol{\theta}^0_j)$ according to (33);
end
Simulated Tempering based importance sampling:
while $\Lambda_s < 1$ and $s < S$ do
    Update stage number $s = s + 1$ ;
    Sort negative log-likelihoods: $-\psi^{s-1}_{(1)} < -\psi^{s-1}_{(2)} < \cdots < -\psi^{s-1}_{(M)}$;
    Update the correction factor $\beta_s = \min\{1 - \Lambda_{s-1}, -\phi/\psi^{s-1}_{(H)}\}$;
    Update $\Lambda_s = \Lambda_{s-1} + \beta_s$;
    for $j = 1, \ldots, M$ do
        Compute sample weight $\tilde{\ell}^s_j = \exp\{\beta_s \cdot \psi^{s-1}_{(j)}\}$;
    end
    Normalize the sample weight: $\ell^s_j = \tilde{\ell}^s_j / \sum_{m=1}^M \tilde{\ell}^m_j$;
    Compute the mean $\hat{\boldsymbol{\mu}}_s$ and covariance matrix $\hat{\mathbf{C}}_s$;
    for $j = 1, \ldots, M$ do
        Select $i_j = j$ with probability $\ell^s_j$ and let $\tilde{\boldsymbol{\theta}}^s_j = \boldsymbol{\theta}^{s-1}_{i_j}$ and $\tilde{\psi}^s_j = \psi^{s-1}_{i_j}$;
        Draw $\boldsymbol{\theta}^{s,*}_j \sim g_s(\cdot; \hat{\boldsymbol{\mu}}_s, \hat{\mathbf{C}}_s)$ according to (40);
        Execute PHD filter using $\boldsymbol{\theta}^{s,*}_j$ to estimate $\psi^{s,*}_j = \log f(\mathbf{Z}_{1:K}|\boldsymbol{\theta}^{s,*}_j)$ according to (33);
        Compute acceptance probability $\eta = \min\left\{1, \exp[\beta_s(\psi^{s,*}_j - \tilde{\psi}^s_j)]\frac{g_s(\tilde{\boldsymbol{\theta}}^s_j)}{g_s(\boldsymbol{\theta}^{s,*}_j)}\right\}$;
        Sample $u \sim \mathscr{U}_{[0,1]}$;
        if $u < \eta$ then
            Set $\boldsymbol{\theta}^s_j = \boldsymbol{\theta}^{s,*}_j$ and $\psi^s_j = \psi^{s,*}_j$;
        else
            Set $\boldsymbol{\theta}^s_j = \tilde{\boldsymbol{\theta}}^s_j$ and $\psi^s_j = \tilde{\psi}^s_j$;
        end
    end
end
Compute the maximum a posterior (MAP) estimate $\widehat{\boldsymbol{\theta}}$ from the sample $\{\boldsymbol{\theta}^S_j\}^M_{j=1}$.
```

5. Experiments

In this section, we verify the correctness and validity of the proposed algorithm by using an example of multiple sensors. Estimating the systematic errors of multiple sensors is important for sensor registration in cooperative networked surveillance [36]. In this example, we estimate the systematic errors of two sensors. The advantages of the proposed algorithm introduced in Section 1.3 are all successfully proved. In order to ensure that those who are interested in the experiments can conduct these experiments smoothly, we also provide the source codes as the supplementary materials.

5.1. Experimental Models

Here, we use the measurements given by R (here $R = 2$) static sensors to estimate the systematic errors of these sensors. Here, we denote the observation dataset reported by sensor $r_k \in \{1, \ldots, R\}$ at time k by $\boldsymbol{Z}_k^{(r_k)} = \{\boldsymbol{z}_{k,1}^{(r_k)}, \ldots, \boldsymbol{z}_{k,m_{k,r_k}}^{(r_k)}\}$. To verify the first advantage introduced in Section 1.3, we use the detection probability p_D to characterize the detection uncertainty, and missed detection. The sensor observations are affected by the systematic errors and the measurement noises. The clutters are characterized by using Poisson RFS. We assume that the observations provided by the sensors are the azimuth and range to the objects in the surveillance system. For an observation $\boldsymbol{z} \in \boldsymbol{Z}_k^{(r_k)}$ which is related to the object with state $\boldsymbol{x} \in \boldsymbol{X}_k$, the measurement likelihood of the r-th sensor is modeled as:

$$g_k^{(r_k)}(\boldsymbol{z}|\boldsymbol{x}, \boldsymbol{\theta}) = \mathcal{N}\left(\boldsymbol{z}; h_k^{(r_k)}(\boldsymbol{x}) - \boldsymbol{\theta}^{(r_k)}, \boldsymbol{\Sigma}_k^{(r_k)}\right) \tag{41}$$

where $h_k^{(r_k)}(\boldsymbol{x})$ is the measurement function of the r-th sensor, and its specific form is as follows:

$$h_k^{(r_k)}(\boldsymbol{x}) = \begin{bmatrix} \sqrt{(x - x_r)^2 + (y - y_r)^2} \\ \arctan(\frac{x - x_r}{y - y_r}) \end{bmatrix} \tag{42}$$

where $(x_r,\ y_r)$ is the position of the r-th sensor. The unknown parameter vector $\boldsymbol{\theta}$ consists of four parameters as follows:

$$\boldsymbol{\theta} = [\Delta\eta_1,\ \Delta\rho_1,\ \Delta\eta_2,\ \Delta\rho_2]^T \tag{43}$$

where $\boldsymbol{\theta}^{(r)} = [\Delta\eta_r,\ \Delta\rho_r]^T$, which appears in Equation (41), denotes the systematic error vector for the r-th sensor ($r = 1, 2$), $\Delta\eta$ is the azimuth systematic error and $\Delta\rho$ is the range systematic error. $\boldsymbol{\Sigma}_k^{(r_k)} = diag\left[(\sigma_\eta^{(r)})^2, (\sigma_\rho^{(r)})^2\right]$ is the covariance matrix in (41). $p_D^{(r)}$ denotes the detection probability of the r-th sensor.

To verify the second advantage of the proposed algorithm introduced in Section 1.3, the targeted system surveilled by the sensors consists of a varying number of objects. We use the probability of survival p_s to characterize the uncertainty of target existence. The state of each individual object is described by the vector $\boldsymbol{x} = [x,\ \dot{x},\ y,\ \dot{y}]^T$, here $(x,\ y)$ denotes the position of the object and $(\dot{x},\ \dot{y})$ denotes the velocity.

The state of each object in the surveillance system evolves according to the Markov transitional density in time, and it is independent of the unknown parameters $\boldsymbol{\theta}$. Here, we use the following motion model:

$$\pi_{k|k-1}(\boldsymbol{x}|\boldsymbol{x}_{k-1}) = \mathcal{N}(\boldsymbol{x}; \boldsymbol{F}_k\boldsymbol{x}_{k-1}, \boldsymbol{Q}_k) \tag{44}$$

where

$$\boldsymbol{F}_k = \boldsymbol{I}_2 \otimes \begin{bmatrix} 1 & T_k \\ 0 & 1 \end{bmatrix},\ \boldsymbol{Q}_k = \boldsymbol{I}_2 \otimes \varpi \begin{bmatrix} \frac{T_k^3}{3} & \frac{T_k^2}{2} \\ \frac{T_k^2}{2} & T_k \end{bmatrix} \tag{45}$$

and $\otimes$ is the Kroneker product, $T_k = t_{k+1} - t_k$ denotes the time interval of the sensors, and ϖ denotes the intensity of process noises.

5.2. Experimental Setup

The typical observation dataset sequence $\boldsymbol{Z}_{1:K}$, the real states of the multiple objects and the location of the sensors in the experiment are shown in Figure 1. The number of observation datasets used for estimating parameter is $K = 20$. The number of component objects is varying in the experiment, with $n_k = 3$ for $k = 1, 2, 3, 4, 5$; $n_k = 5$ for $k = 6, 7, 8, 9, 10$; $n_k = 4$ for $k = 11, 12, 13, 14, 15$; and $n_k = 3$ for $k = 16, 17, 18, 19, 20$. The two sensors in the experiment work asynchronously: Sensor 1 provides observation datasets at odd time steps, and Sensor 2 provides observation datasets at even time steps. $T_k = 1$ s is the time interval between k and $k - 1$. The location of two sensors are (x_1, y_1)

= (0 km, 0 km) and (x_2, y_2) = (200 km, 100 km). The probability of detection is $p_D^{(1)} = p_D^{(2)} = 0.98$, the expected numbers of Poisson clutters for two sensors are: $\lambda_1 = \lambda_2 = 5$, while $c_1(\boldsymbol{z}|\boldsymbol{\theta}) = c_2(\boldsymbol{z}|\boldsymbol{\theta})$ are uniform distribution over $\mathscr{Z} = [-\pi, \pi] \times [0 \text{ km}, 1000 \text{ km}]$. The standard deviation of observation noises is $\sigma_\rho^{(1)} = \sigma_\rho^{(2)}$ =1 km and $\sigma_\eta^{(1)} = \sigma_\eta^{(2)} = 0.2°$. The true systematic errors are $\Delta\eta_1 = -3°$, $\Delta\rho_1$ =10 km, $\Delta\eta_2 = 3°$, $\Delta\rho_2 = -10$ km.

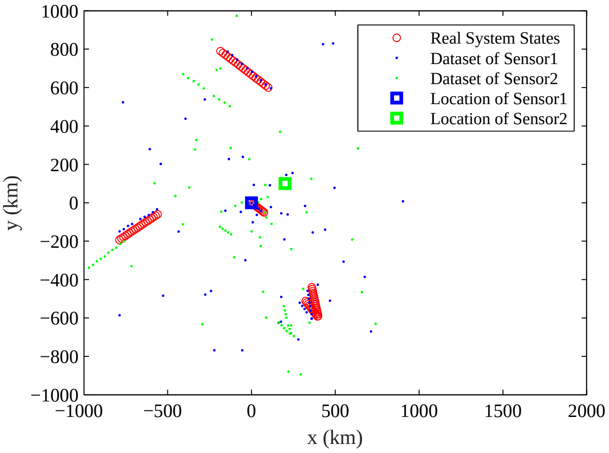


Figure 1. The location of sensors (marked by square), the real states of the studied system which contains multiple objects and the accumulated observation dataset $\boldsymbol{Z}_{1:K}$ for two sensors.

The SMC based implementation of PHD filter uses $N = 1000$ particles per newborn object and persistent object. The probability of object survival is $p_s = 0.99$. The expected number of object birth is 0.03. The prior distribution $p_0(\boldsymbol{\theta})$ is a uniform distribution over $[-10°, +10°] \times [-20\,\text{km}, +20\,\text{km}] \times [-10°, +10°] \times [-20\,\text{km}, +20\,\text{km}]$. The sample number in the systematic errors space is $M = 1000$. The parameters used in adaptively choosing β are: $\phi = 65$ and $H = 0.5\,M$. We set the maximum number of stages by $S = 10$.

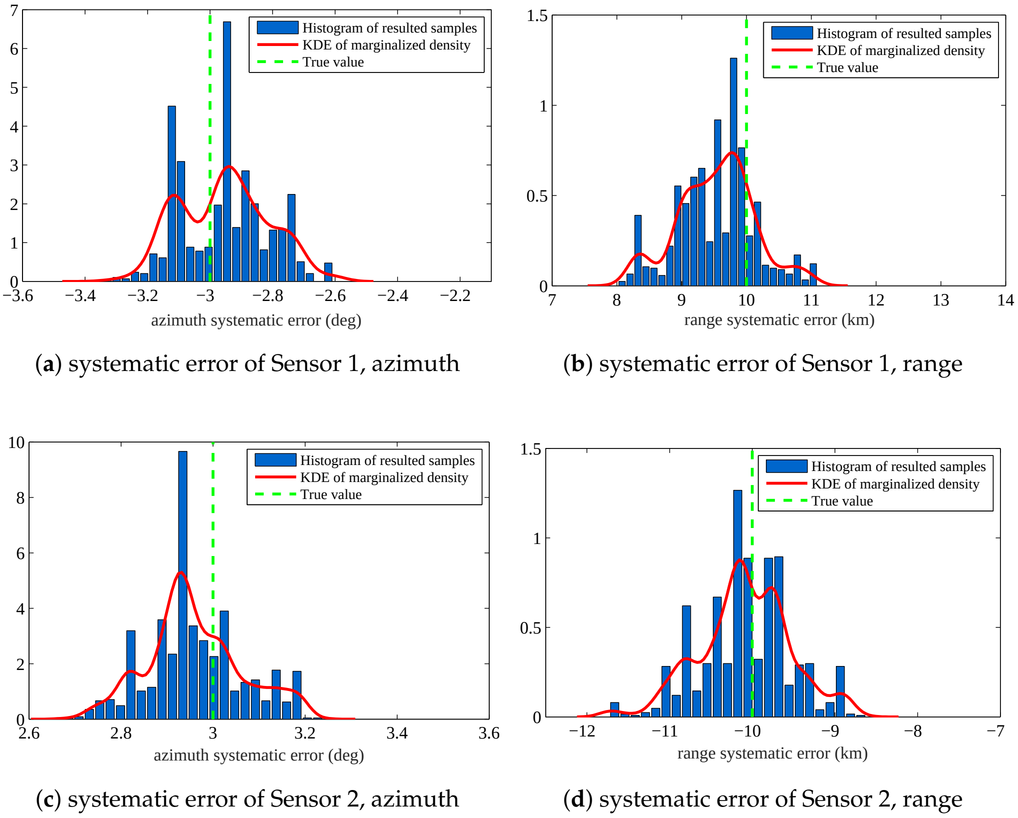
5.3. Experimental Results

The estimated systematic errors for the sensors are given in Figure 2. Figure 2 is the histogram with a distribution fit of the resulted sample $\{\boldsymbol{\theta}_i^S\}_{i=1}^M$. Figure 2a–d marginalizes the posterior PDF $p(\boldsymbol{\theta}|\boldsymbol{Z}_{1:K})$ approximated by the resulted samples to one dimension and obtain the estimated azimuth systematic error $\Delta\eta$ and range systematic error $\Delta\rho$ for the sensors. The solid lines in each figure are the kernel density estimate (KDE) of the marginal posterior PDF. The vertical dashed lines in each figure are the true values of the sensors' systematic errors. Since we have obtained the KDE, we use the maximum a posterior (MAP) method to estimate the sensors' systematic errors. The estimation results are shown in Table 1.

In Figure 2 and Table 1, we can know that the proposed RFS based parameter estimation algorithm can provide the accurate estimated parameters by using the RFS based measurement model and the measurements set that consists of noises, false detections, and missed detections. This is consistent with the first advantage introduced in Section 1.3. Moreover, the proposed algorithm gives the accurate estimated parameters for the stochastic system in which the number of constituent objects (targets) is varying over time. This is consistent with the second advantage introduced in Section 1.3. The obtained experimental results show an outstanding accuracy of the proposed algorithm, although the size of dataset is only 20 scans.

Table 1. True and estimated sensors' systematic errors.

	$\Delta\eta_1$ **(deg)**	$\Delta\rho_1$ **(km)**	$\Delta\eta_2$ **(deg)**	$\Delta\rho_2$ **(km)**
True values	-3	10	3	-10
Estimated values	-2.9399	9.9790	2.9418	-10.0558



(**a**) systematic error of Sensor 1, azimuth

(**b**) systematic error of Sensor 1, range

(**c**) systematic error of Sensor 2, azimuth

(**d**) systematic error of Sensor 2, range

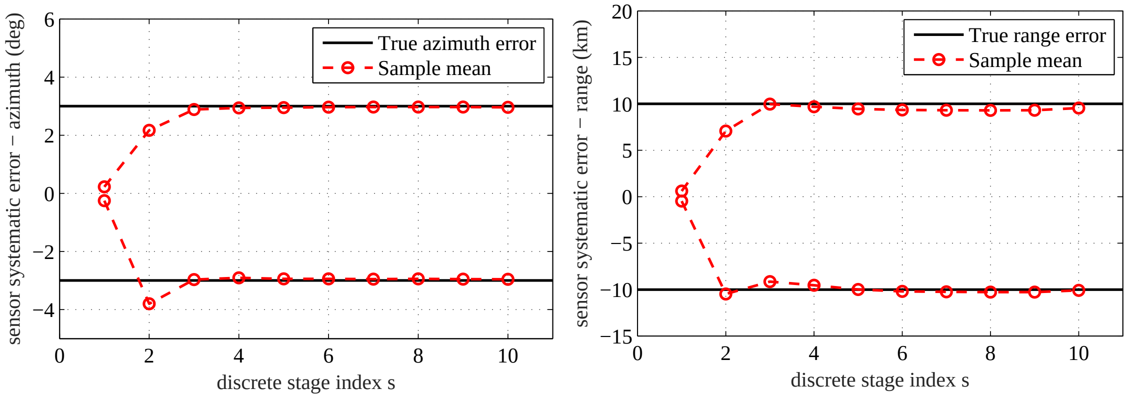
Figure 2. Histograms with distribution fit of the systematic error sample $\{\boldsymbol{\theta}_j^S\}_{j=1}^M$ that are marginalized to $\Delta\eta_1$: (**a**) systematic error of Sensor 1, azimuth, $\Delta\rho_1$; (**b**) systematic error of Sensor 1, range, $\Delta\eta_2$; (**c**) systematic error of Sensor 2, azimuth, $\Delta\rho_2$; and (**d**) systematic error of Sensor 2, range.

To quantitatively assess the convergence of the algorithm, we adopt the convergence diagnostic named as the estimated potential scale reduction (EPSR). The definition of EPSR is given in [37,38]. The resulted sequence of correction factors β and the EPSR at each stage $s = 1, \ldots, 10$ are listed in Table 2. From this table, we can know that the simulated tempering process works effectively. As the stage number increases, the correction factors of each stage increase, and EPSR at each stage decreases, which means that the resulted sample in the simulated tempering process increasingly concentrates in the area of the parameter space characterized by the true likelihood.

Table 2. Correction factors and EPSR during simulated tempering.

s	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
β_s	0.0557	0.0663	0.0854	0.0883	0.0896	0.0903	0.0906	0.0908	0.0910	0.0700
EPSR	1.0043	0.9492	0.8862	0.8731	0.8680	0.8673	0.8671	0.8670	0.8669	0.8667

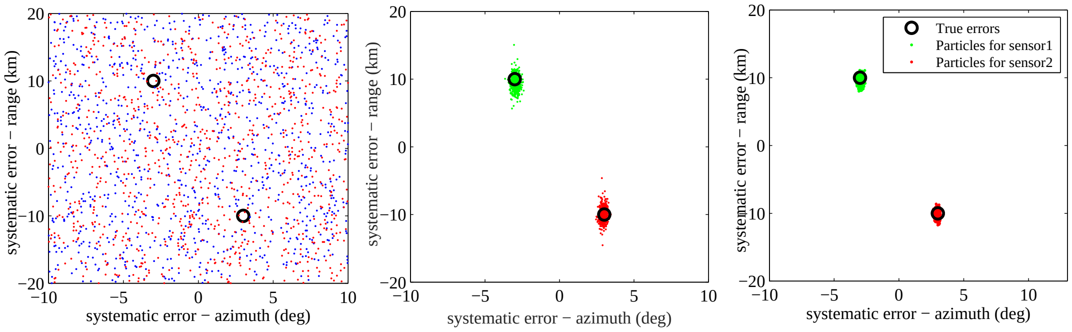
The estimates $\boldsymbol{\theta}_s$ (the mean of all the particles at each stage) over the simulated tempering stages obtained by running the proposed algorithm are shown in Figure 3. The black solid lines are the true range and azimuth systematic errors of the sensors. The red circles are the sample mean at different tempering stages. From this figure, we can see that, after the fourth stage, the estimation results are very close to the real systematic errors. Since $\Lambda_s = \sum_{j=1}^{s} \beta_j$ is still less than 1, the estimation process will continue until reaching the maximum stage number $S = 10$.



(**a**) (**b**)

Figure 3. True sensors' systematic errors and sample mean for all stages: (**a**) sensors' systematic errors–azimuth; and (**b**) sensors' systematic errors–range.

The SMC approximation of the posterior PDF of the sensors' systematic errors according to Equation (35) at the first stage, fifth stage and tenth stage are shown in Figure 4. The true systematic errors for sensors are represented by black circles. In Figure 4, we can see that, as the simulated tempering process evolves, the particles gradually concentrate on the real systematic errors of the sensors.

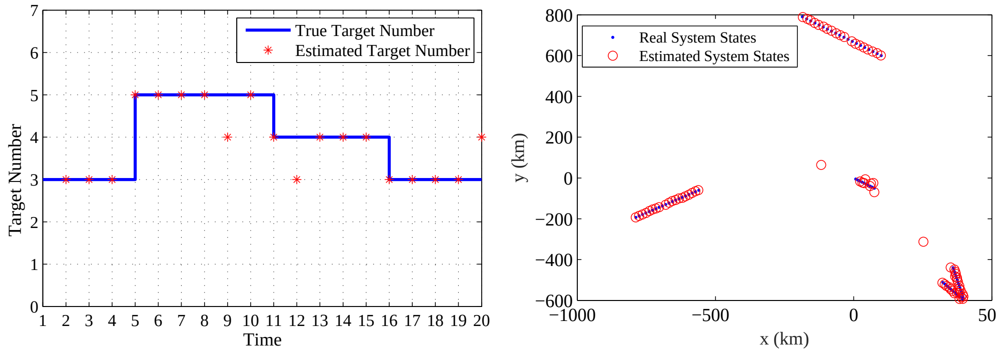


(**a**) (**b**) (**c**)

Figure 4. Particle approximation of the posterior PDF of the systematic errors of the sensors at: $s = 1$ (**a**); $s = 5$ (**b**); and $s = 10$ (**c**).

5.4. System States and Object Number Estimation

As given in Section 1.3, the third advantage of the proposed algorithm is that it can provide the estimate of the system states and the number of targets. Figure 5a gives the real target number contained in the system and the estimated one. Figure 5b gives the real and estimated system states at all the sampling time. From this figure, we can find that the proposed algorithm can accurately estimate the varying number of targets and system states at the final stage by using the estimated systematic errors of sensors.



(**a**) (**b**)

Figure 5. The estimated target number and system states given by the proposed algorithm: (**a**) the real and estimated target number; and (**b**) the real and estimated system states.

5.5. Comparison with Metropolis Hastings Algorithm

There are many probability distributions from which it is impossible to sample directly. The Metropolis Hastings (MH) algorithm is a typical MCMC algorithm that can generate a sequence of random samples from these distributions. Since it is the most widely used method in statistics and in statist physics, we verify the superiority of the proposed algorithm by comparing with MH algorithm. Figure 6 displays the histogram of the resulted samples given by using MH algorithm with the identical experimental setup. From this figure, we can find that the resulted samples given by MH algorithm are more decentralized than the proposed algorithm. The estimated results are shown in Table 3. We can find that the proposed algorithm are more accurate than MH algorithm.

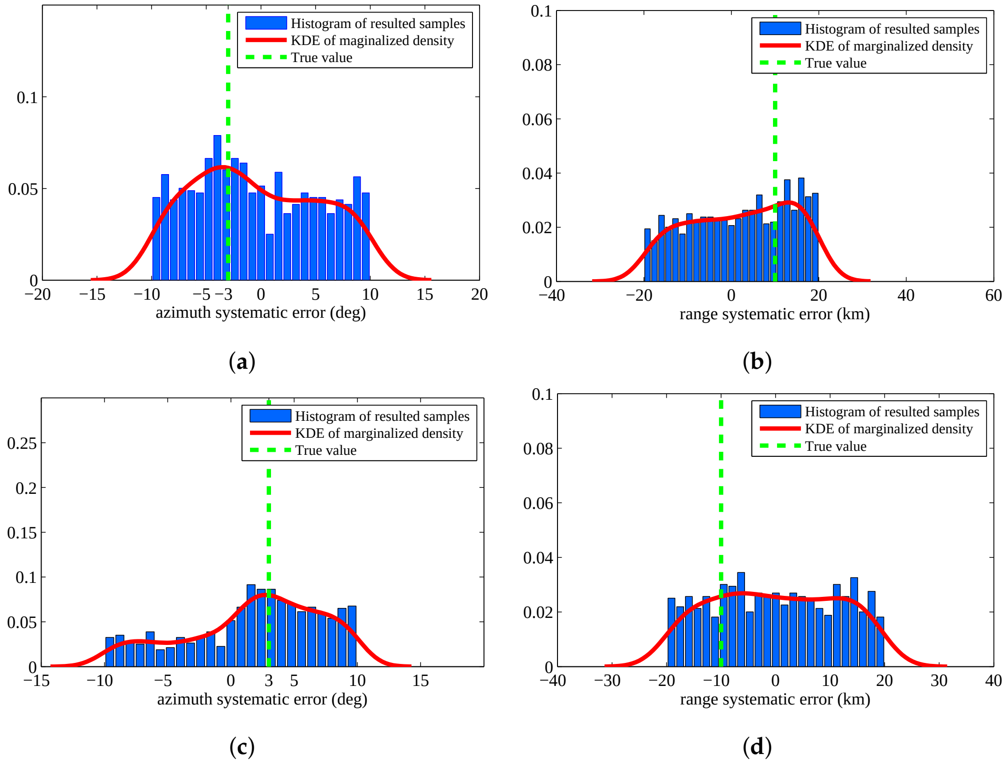


Figure 6. Histograms with distribution fit of the systematic error sample $\{\boldsymbol{\theta}_j^S\}_{j=1}^M$ by using Metropolis Hastings algorithm: (**a**) systematic error of Sensor 1, azimuth; (**b**) systematic error of Sensor 1, range; (**c**) systematic error of Sensor 2, azimuth; and (**d**) systematic error of Sensor 2, range.

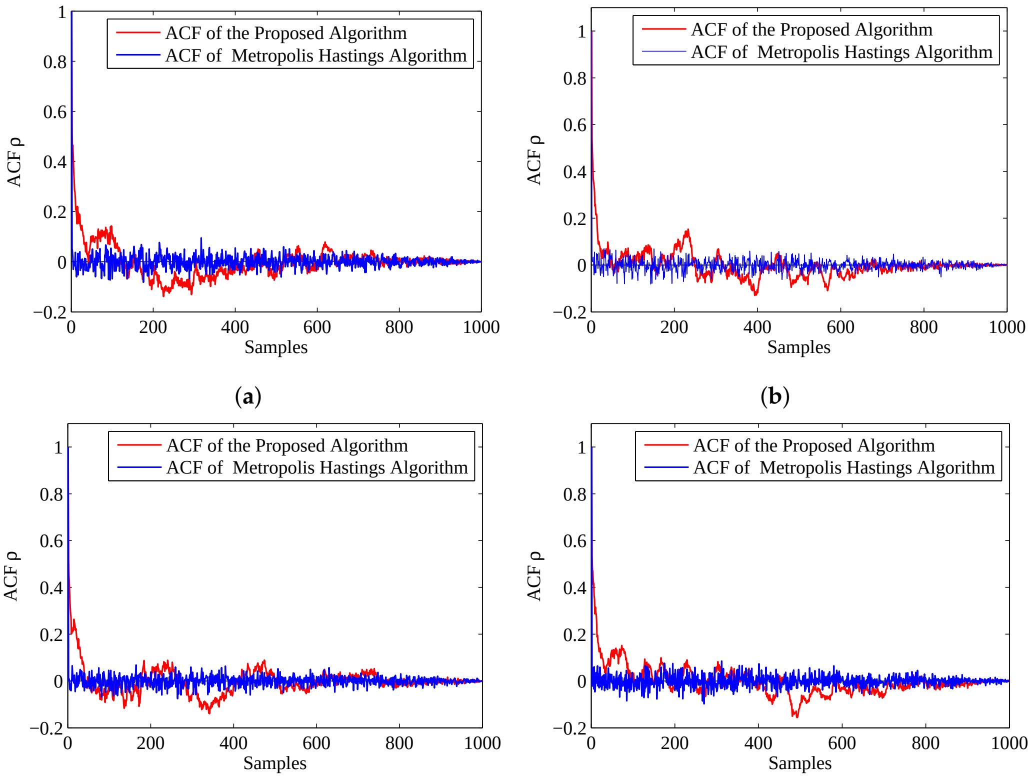
Table 3. True and estimated sensors' systematic errors by using two algorithms.

	$\Delta\eta_1$ (deg)	$\Delta\rho_1$ (km)	$\Delta\eta_2$ (deg)	$\Delta\rho_2$ (km)
True values	-3	10	3	-10
Estimated values by proposed algorithm	-2.9399	9.9790	2.9418	-10.0558
Estimated values by MH algorithm	-3.6075	13.1868	2.7403	-5.9794

To compare the accuracy of the importance sampling process, we adapt the evaluation criteria named as autocorrelation function (ACF) [39,40]. According to the fundamentals of MCMC, the samples generated by using MCMC are auto-correlated. Compared with the independent samples, the information content of the samples is relatively reduced. The sample based ACF at lag k of a set of samples is defined as follows:

$$\rho_k = \frac{\frac{1}{M-k}\sum_{s=1}^{M-k}(\theta_s-\bar{\theta})(\theta_{s+k}-\bar{\theta})}{\frac{1}{M-1}\sum_{s=1}^{M}(\theta_s-\bar{\theta})^2} \tag{46}$$

where $\bar{\theta} = \frac{1}{M}\sum_{s=1}^{M}\theta_s$ is the Monte Carlo estimate, and θ_s is a sample. From the definition, we can see that, if the ACF dies off to 0 more rapidly, it indicates that the samples are less auto-correlated and more accurate. Figure 7 displays the ACF of the proposed algorithm and MH algorithm. From this figure, we can find that the ACF of the proposed algorithm dies off to 0 more rapidly than the MH algorithm. Thus, the accuracy of the prosed algorithm is much higher.



(**a**)

(**b**)

(**c**)

(**d**)

Figure 7. ACF of the proposed algorithm and MH algorithm: (**a**) ACF for Sensor 1, azimuth; (**b**) ACF for Sensor 1, range; (**c**) ACF for Sensor 2, azimuth; and (**d**) ACF for Sensor 2, range.

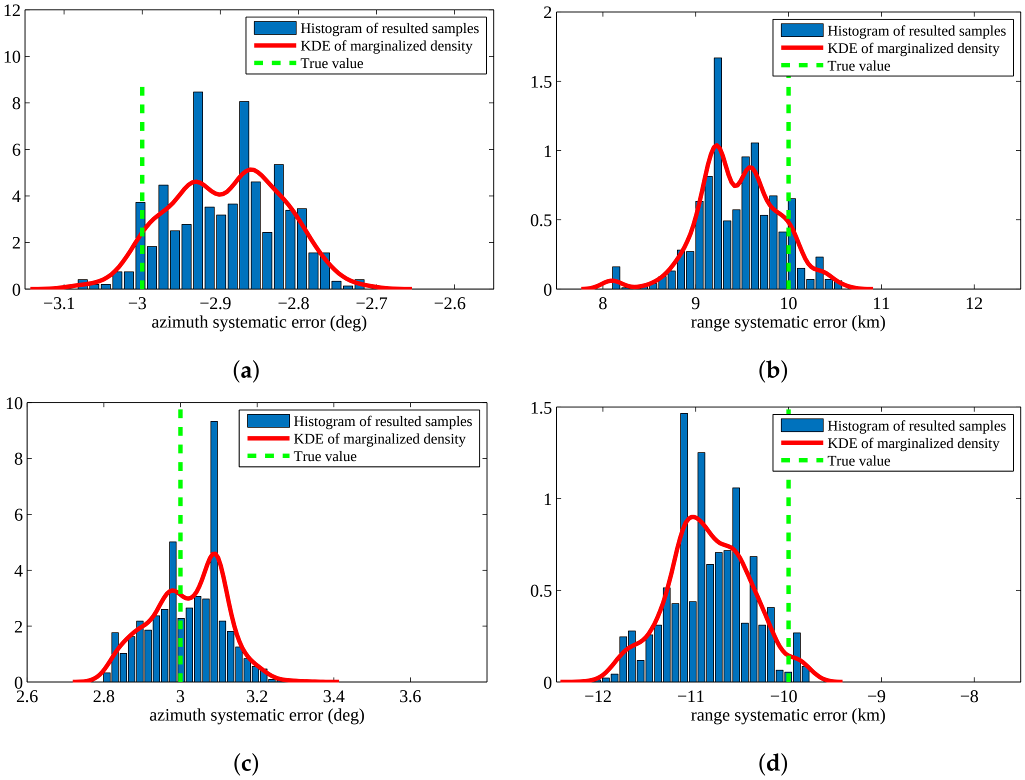
The former experiments assume that the prior $p_0(\boldsymbol{\theta})$ is a uniform distribution. To verify the superiority of the proposed algorithm in a more general case, we assume the prior $p_0(\boldsymbol{\theta})$ is a Gaussian distribution with mean value $[-2.5;\ 9.5;\ 2.5; -9.5]$ and covariance diag $[10,\ 20,\ 10,\ 20]$. Figure 8 displays the histogram of the resulted samples by using the proposed algorithm. We can find that the resulted KDE densities have deviation by comparing with the former experiment. This is caused by the inaccurate prior distribution. Figure 9 displays the histogram of the resulted samples given by HM algorithm. From this figure, we can find that the resulted samples are still very decentralized and the distribution of samples is heavily depending on the prior distribution.

The estimated results of the two algorithms are given in Table 4. We can find that the accuracy of the proposed algorithm deteriorates slightly, but it is still better than the results of MH algorithm.

The ACF of the two algorithms is given in Figure 10. We can find that the ACF of the proposed algorithm dies off to 0 more slowly than the former experiment, but still more rapidly than the MH algorithm. Thus, the accuracy of the estimated results of the proposed algorithm is a little better than the MH algorithm.

Table 4. True and estimated sensors' systematic errors for Gaussian distribution.

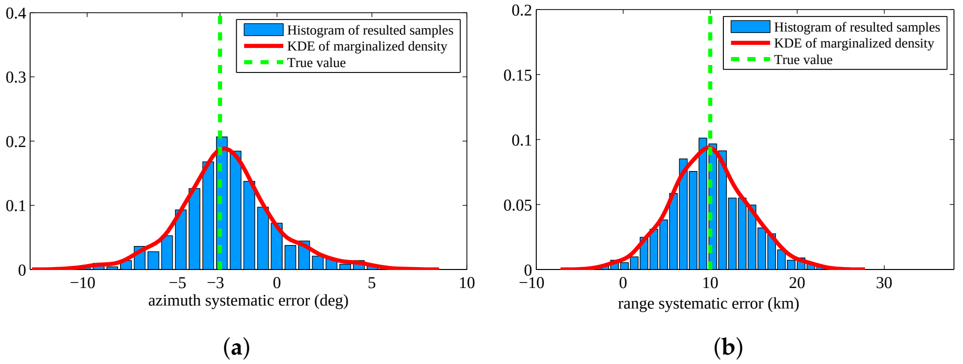
	$\Delta\eta_1$ (deg)	$\Delta\rho_1$ (km)	$\Delta\eta_2$ (deg)	$\Delta\rho_2$ (km)
True values	−3	10	3	−10
Estimated values by proposed algorithm	−2.8619	9.2276	3.0909	−11.0458
Estimated values by MH algorithm	−2.7219	9.7624	2.4841	−8.8048



(**a**) (**b**)

(**c**) (**d**)

Figure 8. Histograms with distribution fit of the sample $\{\boldsymbol{\theta}_j^S\}_{j=1}^M$ by using the proposed algorithm when the prior is Gaussian distribution: (**a**) systematic error of Sensor 1, azimuth; (**b**) systematic error of Sensor 1, range; (**c**) systematic error of Sensor 2, azimuth; and (**d**) systematic error of Sensor 2, range.



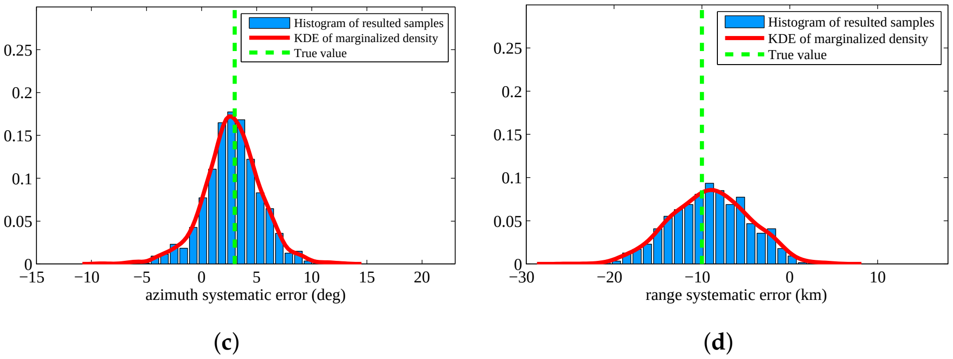


Figure 9. Histograms with distribution fit of the sample $\{\theta_j^S\}_{j=1}^M$ by using MH algorithm when the prior is Gaussian distribution: (**a**) systematic error of Sensor 1, azimuth; (**b**) systematic error of Sensor 1, range; (**c**) systematic error of Sensor 2, azimuth; and (**d**) systematic error of Sensor 2, range.

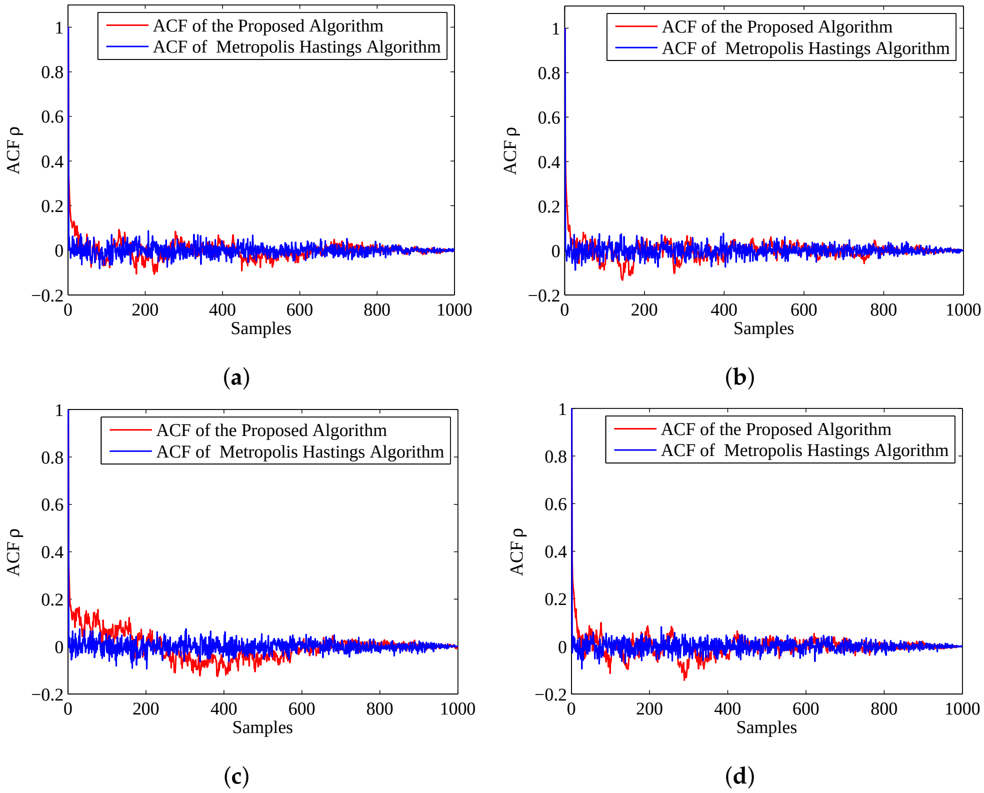


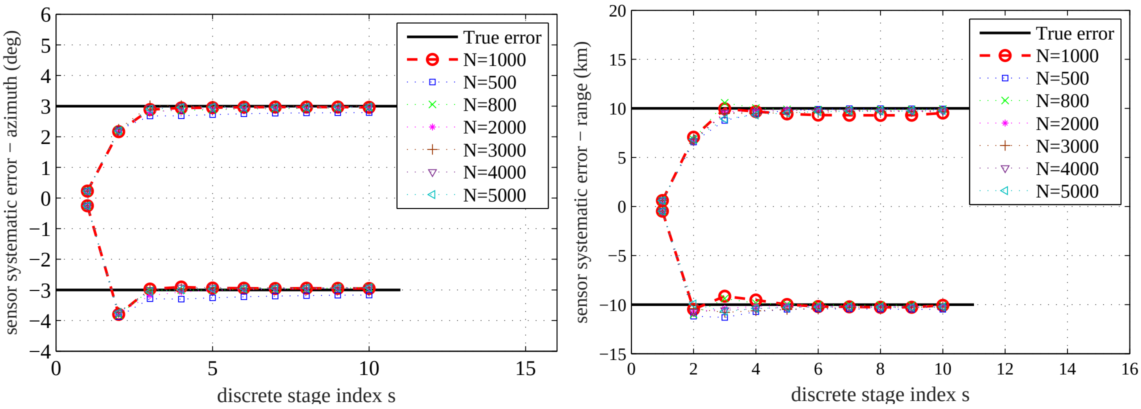
Figure 10. ACF of the proposed algorithm and MH algorithm when the prior is Gaussian distribution: (**a**) ACF for Sensor 1, azimuth; (**b**) ACF for Sensor 1, range; (**c**) ACF for Sensor 2, azimuth; and (**d**) ACF for Sensor 2, range.

5.6. Sensitivity Analysis

In this section, we analyze the sensitivity of the proposed algorithm to a number of important parameters, such as the particle number N for PHD filter, the sample number M in parameter space as well as the parameter ϕ related to the simulated tempering process.

5.6.1. Particle Number N

The PHD filter can provide the estimation of the likelihood $f(\boldsymbol{Z}_{1:K}|\boldsymbol{\theta})$. Hence, the accuracy of the proposed parameter estimation algorithm may depend on the particle number N used by the SMC based PHD filter. The influence of the particle number N on the parameter estimation is presented in Figure 11. The estimated systematic errors at the 10th stage for different particle number N by using MAP are given in Table 5.



(**a**) (**b**)

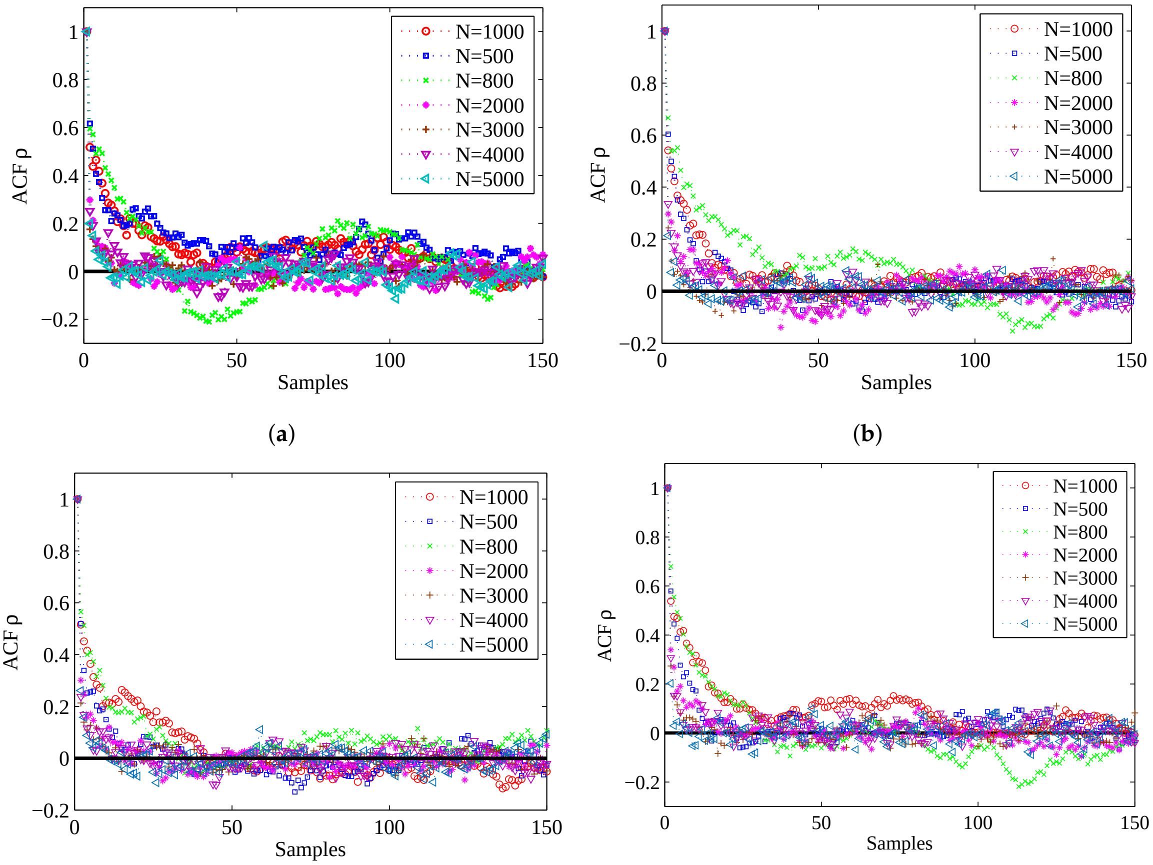
Figure 11. The sample mean for different particle number N used by PHD filter: (**a**) sensors' systematic errors–azimuth; and (**b**) sensors' systematic errors–range

Table 5. Estimated systematic errors at stage s = 10 for different particle number N.

N	500	800	1000	2000	3000	4000	5000
$\Delta\eta_1$ (deg)	3.1676	−2.9190	−2.9399	−2.9443	−2.9465	−2.9415	−2.9418
$\Delta\rho_1$ (km)	9.9420	9.9475	9.9790	9.9776	9.9874	9.9828	9.9813
$\Delta\eta_2$ (deg)	2.7879	2.9364	2.9418	2.9739	3.0019	2.9791	2.9827
$\Delta\rho_2$ (km)	−10.4852	−10.0786	−10.0558	−10.0259	−10.0569	−10.0507	−10.0481

In Figure 11 and Table 5, we can find that the estimation error has a decreasing tendency as N increases. With more particles, PHD filter can give more accurate likelihood $f(\boldsymbol{Z}_{1:K}|\boldsymbol{\theta})$, and thus leads to a better coverage of the parameter space. The plot shows that the decreasing tendency in estimation errors by increasing N is not proportional, and when N increases to a certain degree, the improvement of estimation accuracy is very small. The result shows that, if N increases to a certain degree, it will not significantly affect the parameter estimation results. As the accuracy of the estimation results is restricted by the quality of the model and data used in the parameter estimation process, the estimation results without any errors by continuously increasing N is impossible. The computational complexity will also be increased as N increases. Therefore, we should achieve a balance between computational complexity and estimation accuracy.

Figure 12 shows the ACF by using different values of N. For the convenience of analysis, we give the ACF of the first 150 samples. From this figure, we can find that, if N is larger, the corresponding ACF will die off to 0 more rapidly, so the accuracy is higher.



(**a**) (**b**)

(**c**) (**d**)

Figure 12. ACF for different particle number *N*: (**a**) ACF for Sensor 1, azimuth; (**b**) ACF for Sensor 1, range; (**c**) ACF for Sensor 2, azimuth; and (**d**) ACF for Sensor 2, range.

Table 6 shows the EPSR at all the stages for different particle number *N* used by the PHD filter. From this table, we can find that, as the number of stage increases, the EPSR of each stage decreases. This means that as the simulated tempering process progresses, the sample increasingly concentrates in the area of the parameter space characterized by the true likelihood. The increase of the particle number *N* seams to have no influence on the convergence of the proposed algorithm.

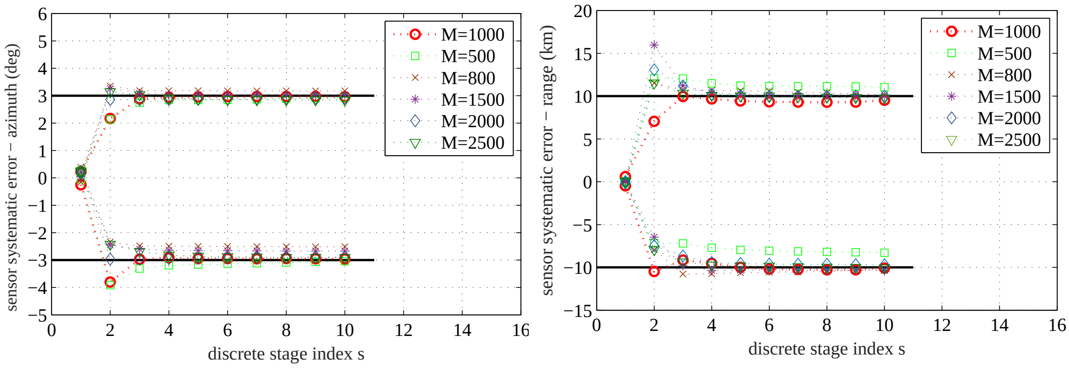
Table 6. EPSR for different particle number *N*.

N	500	800	1000	2000	3000	4000	5000
$s = 1$	1.0043	1.0043	1.0043	1.0043	1.0043	1.0043	1.0043
$s = 2$	0.9482	0.9447	0.9492	0.9426	0.9439	0.9344	0.9440
$s = 3$	0.8778	0.8765	0.8862	0.8756	0.8726	0.8780	0.8744
$s = 4$	0.8694	0.8692	0.8731	0.8684	0.8679	0.8691	0.8687
$s = 5$	0.8667	0.8674	0.8680	0.8674	0.8673	0.8677	0.8674
$s = 6$	0.8663	0.8670	0.8673	0.8672	0.8671	0.8672	0.8671
$s = 7$	0.8662	0.8670	0.8671	0.8670	0.8671	0.8671	0.8670
$s = 8$	0.8662	0.8670	0.8670	0.8668	0.8670	0.8671	0.8670
$s = 9$	0.8661	0.8669	0.8669	0.8668	0.8669	0.8670	0.8668
$s = 10$	0.8661	0.8668	0.8667	0.8668	0.8667	0.8669	0.8668

5.6.2. Sample Number *M*

The sample number *M* in the parameter space may also have influence on the accuracy of parameter estimation. The influence of the sample number *M* on the parameter estimation result is summarized in Figure 13. Table 7 is the MAP estimate at the 10th stage for different M. From the plot, we can find that the decreasing tendency in estimation errors by increasing M is quite similar to the result caused by increasing N. If M increases to a certain degree, the improvement of estimation accuracy is very small. There also needs a balance between computational complexity and estimation accuracy.

The EPSR of different sample size M is given in Table 8. From this table, we can find that the increase of M will result in the increase of EPSR, which means that the convergence of the proposed algorithm will be increased.



(**a**) (**b**)

Figure 13. The sample mean for different sample number M in the parameter space: (**a**) sensors' systematic errors–azimuth; and (**b**) sensors' systematic errors–range.

Table 7. Estimated systematic errors at stage $s = 10$ for different sample numbers M.

M	500	800	1000	1500	2000	2500
$\Delta\eta_1$ (deg)	−3.0428	−2.5120	−2.9399	−2.9704	−2.9844	−2.9816
$\Delta\rho_1$ (km)	11.4438	9.7102	9.9790	9.9889	10.0056	10.0567
$\Delta\eta_2$ (deg)	3.2298	2.8761	2.9418	3.0196	2.9482	2.9519
$\Delta\rho_2$ (km)	−8.2334	−10.3842	−10.0558	−10.2542	−9.8020	−10.0881

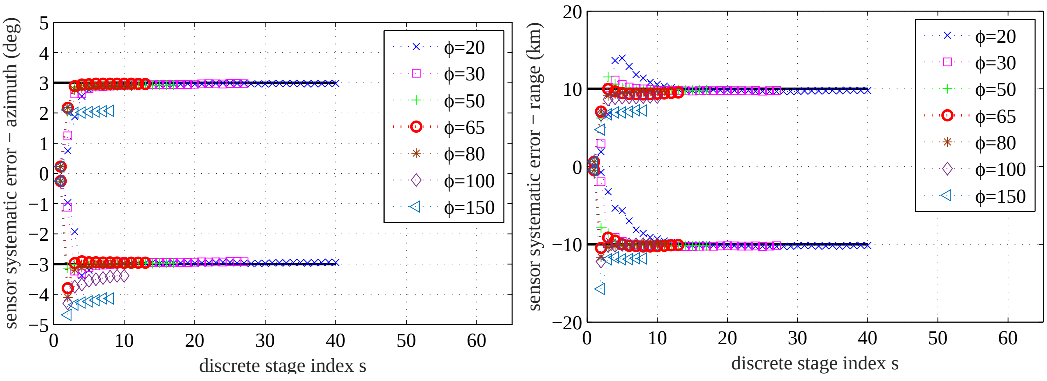
Table 8. EPSR of different sample size M.

M	500	800	1000	1500	2000	2500
$s = 1$	1.0039	0.9905	1.0043	1.0030	1.0661	1.0560
$s = 2$	0.9348	0.9111	0.9492	0.9621	0.9852	0.9933
$s = 3$	0.8784	0.8681	0.8862	0.9028	0.8929	0.9016
$s = 4$	0.8700	0.8671	0.8731	0.8774	0.8780	0.8791
$s = 5$	0.8680	0.8668	0.8680	0.8772	0.8773	0.8779
$s = 6$	0.8674	0.8667	0.8673	0.8671	0.8671	0.8673
$s = 7$	0.8671	0.8666	0.8671	0.8668	0.8669	0.8671
$s = 8$	0.8666	0.8666	0.8670	0.8667	0.8668	0.8670
$s = 9$	0.8665	0.8665	0.8669	0.8666	0.8667	0.8668
$s = 10$	0.8665	0.8665	0.8667	0.8666	0.8666	0.8667

5.6.3. Simulated Tempering Parameter

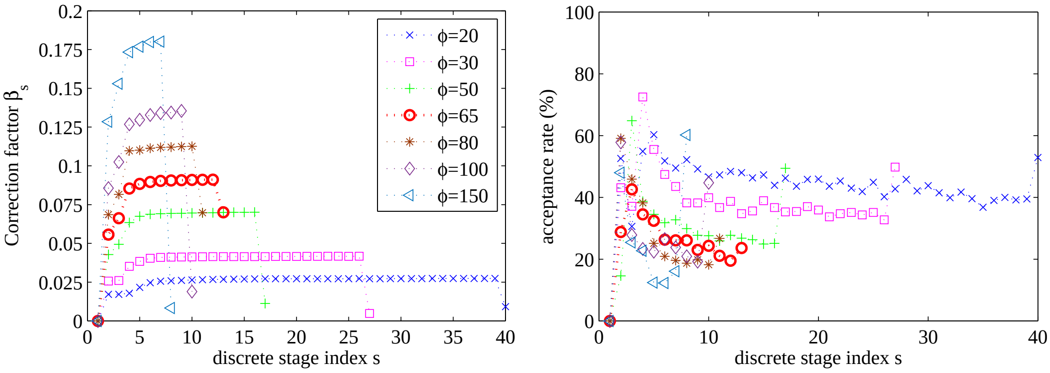
By setting the maximum stage number to be $S = \infty$, the influence of the parameter ϕ on the parameter estimation results is presented in Figures 14 and 15. In Figure 14, we can find that a smaller ϕ will need more stages and get more accurate estimation result. If ϕ is bigger than a certain degree, the number of stages will be very small, and the estimation errors will be bigger. In Figure 15b, we can find that, if a lower value of parameter ϕ is used, the MCMC acceptance rate will be increased.

In Figure 15a, we can find that, if a lower value of parameter ϕ is used, the number of stages S will be increased, because the increments of correction factors β_s become smaller.



(**a**) (**b**)

Figure 14. The sample mean for different ϕ: (**a**) sensors' systematic errors, azimuth; and (**b**) sensors' systematic errors, range.



(**a**) (**b**)

Figure 15. The relationship between correction factors, acceptance rates and ϕ: (**a**) correction factors; and (**b**) acceptance rates.

The computational complexity of simulated tempering based importance sampling is determined by the number of stages S and the correction factors $\beta_1, \cdots, \beta_S$. To reduce the computational complexity, we tend to use a small value of S. However, at the same time, if we choose a small S, the sequential intermediate distribution ϑ_s will vary rapidly. There is a balance between the number of stages (related to the computational complexity) and the estimation accuracy.

We give the estimated systematic errors at the final stage for different ϕ in Table 9. It seems that, if ϕ decreases, the stage number will increase and the estimated results will be more accurate.

Table 9. Estimated systematic errors at the final stage for different ϕ.

ϕ	20	30	50	65	80	100	150
Maximum stage number	40	27	17	13	11	10	8
$\Delta\eta_1$ (deg)	−2.9850	−2.9832	−2.9717	−2.9399	−3.0010	−3.3901	−4.1380
$\Delta\rho_1$ (km)	9.9788	9.9784	9.9484	9.9790	9.7276	8.9416	7.2124
$\Delta\eta_2$ (deg)	2.9770	2.9701	2.9295	2.9418	2.8938	2.6447	2.0738
$\Delta\rho_2$ (km)	−9.9075	−10.0889	−10.1899	−10.0558	−9.9267	−10.4167	−11.8311

6. Conclusions

We have presented the proposed RFS based parameter estimation algorithm to overcome some disadvantages of the vector based Bayesian algorithms. We also detail how to implement this algorithm by using the SMC based PHD filter and simulated tempering based importance sampling. By applying the SMC based PHD filter, the proposed algorithm successfully extends the application of parameter estimation from the single object system to the stochastic system which consists of a varying number of constituent objects. Without any assumptions under ideal conditions, it can fit the dynamic nature of the studied system and the detection uncertainty of the measurement system very well. It can be successfully used for identification of the stochastic systems which consist of a varying number of constituent objects. It can also be used for parameter estimation problem where the observation dataset is affected by false detections, missed detections and noises.

Supplementary Materials: The supplementary files are the source codes for the experiments of this paper. The related experimental results are also given. The files are available online at http://www.mdpi.com/1099-4300/20/8/569/s1.

Author Contributions: P.W. conceived, designed and performed the simulations and wrote the manuscript. Y.P. provided the basic ideas and analyzed the experimental results. G.L. and R.J. reviewed the manuscript.

Funding: The authors are funded by National Natural Science Foundation of China, grant number [No. 61374185].

Acknowledgments: The authors have received patient guidance and great assistance from Professor Kedi Huang. The authors have also obtained the enthusiastic help in English writing from Dongling Liu.

Conflicts of Interest: The authors declare no conflict of interest. The founding sponsors had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, and in the decision to publish the results.

Abbreviations

The following abbreviations are used in this manuscript:

FISST	Finite Set Statistics
KDE	Kernel Density Estimate
MCMC	Markov Chain Monte Carlo
ML	Maximum Likelihood
PDF	Probability Density Function
PF	Particle Filter
PHD	Probability Hypothesis Density
RFS	Random Finite Set
SMC	Sequential Monte Carlo

References

1. Osman, B.; George, L.B.; Katherine, L.M.; Ernest, P. Model, Reuse, Composition, and Adaptation. In *Research Challenges in Modeling and Simulation for Engineering Complex Systems*; Fujimoto, R., Bock, C., Chen, W., Page, E., Panchal, J.H., Eds.; Springer: Berlin/Heidelberg, Germany, 2017; pp. 87–115.
2. Gupta, S.; Hainsworth, L.; Hogg, J.; Lee, R.; Faeder, J. Evaluation of Parallel Tempering to Accelerate Bayesian Parameter Estimation in Systems Biology. In Proceedings of the 26th Euromicro International Conference on Parallel, Distributed, and Network-Based Processing, Cambridge, UK, 21–23 March 2018; pp. 690–697. [CrossRef]
3. Sima, S.; Mort, N.-P. Online Detection and Parameter Estimation with Correlated Data in Wireless Sensor Networks. In Proceedings of the 2018 IEEE Wireless Communications and Networking Conference (WCNC), Barcelona, Spain, 15–18 April 2018; pp. 1–6. [CrossRef]
4. Jordehi, A.R. Parameter estimation of solar photovoltaic (PV) cells: A review. *Renew. Sustain. Energy Rev.* **2016**, *61*, 354–371. [CrossRef]
5. Ljung, L.; Oderstrom, S. Theory and Practice of Recursive Identification. *IEEE Trans. Autom. Control* **1983**, *30*, 1054–1056. [CrossRef]

6. Ljung, L. Convergence analysis of parametric identification methods. *IEEE Trans. Autom. Control* **1978**, *23*, 770–783. [CrossRef]
7. Yi, L.H.; Yuan, K.W. Review on model validation and parameter estimation approaches of wind power generators. In Proceedings of the 6th International Conference on Renewable Power Generation (RPG), Wuhan, China, 19–20 October 2017; Volume 13, pp. 2407–2411. [CrossRef]
8. Ljung, L. Perspectives on system identification. *Ann. Rev. Control* **2008**, *34*, 1–12. [CrossRef]
9. Rushikesh, K. Simultaneous State and Parameter Estimation for Second-Order Nonlinear Systems. In Proceedings of the 2017 IEEE 56th Annual Conference on Decision and Control (CDC), Melbourne, VIC, Australia, 12–15 December 2017; pp. 2164–2169. [CrossRef]
10. Ding, F.; Chen, T. Hierarchical least squares identification methods for multivariable systems. *IEEE Trans. Autom. Control* **2005**, *50*, 242. [CrossRef]
11. Schon, T.B.; Lindsten, F.; Dahlin, J. Sequential Monte Carlo Methods for System Identification. *IFAC PapersOnLine* **2015**, 775–786. [CrossRef]
12. Poyiadjis, G.; Doucet, A.; Singh, S.S. Particle approximations of the score and observed information matrix in state space models with application to parameter estimation. *Biometrika* **2011**, *98*, 65–80. [CrossRef]
13. Kantas, N.; Doucet, A.; Singh, S.S.; Maciejowski, J.M. An overview of sequential Monte Carlo methods for parameter estimation in general state-space models. *IFAC Proc. Vol.* **2009**, *42*, 774–785, [CrossRef]
14. Dan, C.; Joaquin, M. Nested particle filters for online parameter estimation in discrete-time state-space Markov models. *Bernoulli* **2018**, *24*, 3039–3086. [CrossRef]
15. Zhu, J.; Wang, X.; Lin, X.; Gu, Y. Maximum likelihood estimation from sign measurements with sensing matrix perturbation. *IEEE Trans. Signal Process.* **2014**, *62*, 3741–3753. [CrossRef]
16. Ljung, L. *System Identification, Theory for the User*, 2nd ed.; System Sciences Series; Prentice Hall: Upper Saddle River, NJ, USA, 1999; ISBN 0-138-81640-9.
17. Doucet, A.; Tadic, V.B. Parameter estimation in general state-space models using particle methods (Special section on New trends in statistical information processing). *Ann. Ins. Stat. Math.* **2003**, *55*, 409–422. [CrossRef]
18. Kantas, N.; Doucet, A.; Singh, S.S.; Maciejowski, J.; Chopin, N. On particle methods for parameter estimation in state-space models. *Stat. Sci.* **2015**, *30*, 328–351. [CrossRef]
19. Li, M.; Jing, Z.; Pan, H.; Dong, P. Joint registration and multi-target tracking based on labelled random finite set and expectation maximisation. *IET Radar Sonar Navig.* **2017**, *12*, 312–322. [CrossRef]
20. Xu, J.; Yao, A.L. Random Set Theory. *Lecture Notes Econ. Math. Syst.* **2011**, *647*, 1–33. [CrossRef]
21. Mahler, R.P.S. Multitarget Bayes filtering via first-order multitarget moments. *IEEE Trans. Aeros. Electron. Syst.* **2004**, *39*, 1152–1178. [CrossRef]
22. Mahler, R.P.S. *Statistical Multisource-Multitarget Information Fusion*; Artech House, Inc.: Norwood, MA, USA, 2007; ISBN 1596930926.
23. Barbieri, F.; Rifflart, C.; Vo, B.T.; Rajakaruna, S.; Ghosh, A. Intra-Hour Cloud Tracking Based on Probability Hypothesis Density Filtering. *IEEE Trans. Sustain. Energy* **2018**, *9*, 340–349. [CrossRef]
24. Evers, C.; Naylor, P.A. Optimized Self-Localization for SLAM in Dynamic Scenes using Probability Hypothesis Density Filters. *IEEE Trans. Signal Proc.* **2018**, *66*, 863–878. [CrossRef]
25. Enzo, M.; Giorgio, P. Simulated tempering: A new Monte Carlo scheme. *Europhys. Lett.* **1992**, *19*, 451–458. [CrossRef]
26. Marcos, C.; Scotland, L. Weighted particle tempering. *Comput. Stat. Data Anal.* **2017**, *114*, 26–37. [CrossRef]
27. Graham, M.M. Auxiliary Variable Markov Chain Monte Carlo Methods, Ph.D. Dissertation, University of Edinburgh, Edinburgh, Scotland, 2018.
28. Cappe, O.; Godsill, S.J.; Moulines, E. An Overview of Existing Methods and Recent Advances in Sequential Monte Carlo. *Proc. IEEE* **2007**, *95*, 899–924. [CrossRef]
29. Wang, P.; Li, G.; Ju, R.; Peng, Y. Random Finite Set Based Data Assimilation for Dynamic Data Driven Simulation of Maritime Pirate Activity. *Math. Prob. Eng.* **2017**, *2017*. [CrossRef]
30. Goodman, I.R.; Mahler, R.P.; Nguyen, H.T. *Mathematics of Data Fusion*; Springer: Berlin/Heidelberg, Germany, 1997, ISBN 9048148871.
31. Li, C.; Wang, W.; Kirubarajan, T.; Sun, J.; Lei, P. PHD and CPHD Filtering with Unknown Detection Probability. *IIEEE Trans. Signal Proc.* **2018**, *66*, 3784–3798. [CrossRef]

32. Sidenbladh, H. Multi-target particle filtering for the probability hypothesis density. In Proceedings of the Sixth International Conference of IEEE Information Fusion, Queensland, Australia, 8–11 July 2003; pp. 800–806. [CrossRef]
33. Zajic, T.; Mahler, R. Particle-systems implementation of the PHD multitarget tracking filter. In Proceedings of the SPIE—The International Society for Optical Engineering, Orlando, FL, USA, 25 August 2003; Volume 5096, pp. 291–299. [CrossRef]
34. Baig, F.; Faiz, M.A.; Khan, Z.M.; Shoaib, M. Utilization of Markov chain Monte Carlo approach for calibration and uncertainty analysis of environmental models. In Proceedings of the 2018 International Conference on Computing, Mathematics and Engineering Technologies (iCoMET), Sukkur, Pakistan, 3–4 March 2018; doi:10.1109/ICOMET.2018.8346456.
35. Morelande, M.R.; Ristic, B. Radiological Source Detection and Localisation Using Bayesian Techniques. *IEEE Trans. Signal Proc.* **2009**, *57*, 4220–4231. [CrossRef]
36. Fisher, J.W.; Moses, R.L.; Willsky, A.S. Nonparametric belief propagation for self-calibration in sensor networks. *IEEE J. Sel. Areas Commun.* **2005**, *23*, 809–819. [CrossRef]
37. Gonzalez. M.; Martín, J.; Martínez, R.; Mota, M. Non-parametric Bayesian estimation for multitype branching processes through simulation-based methods. *Comput. Stat. Data Anal.* **2008**, *52*, 1281–1291. [CrossRef]
38. Kevin, P.M. *Machine Learning A Probabilistic Perspective*; The MIT Press: London, UK, 2012; pp. 837–874, ISBN 9780262018029.
39. Neusser, K. *Estimation of the Mean and the Autocorrelation Function*; Time Series Econometrics; Springer: Berlin/Heidelberg, Germany, 2016; ISBN 3319328611.
40. Bykhovsky, D.; Trigano, T. Numerical Generation of Compound Random Processes with an Arbitrary Autocorrelation Function. *Fluct. Noise Lett.* **2018**, *17*, 1850001. [CrossRef]