

Research Article

Random Finite Set Based Data Assimilation for Dynamic Data Driven Simulation of Maritime Pirate Activity

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

College of Systems Engineering, National University of Defense Technology, Changsha 410073, China

Correspondence should be addressed to Peng Wang; wangpengnudu@163.com

Received 22 July 2017; Accepted 9 November 2017; Published 11 December 2017

Academic Editor: Filippo Cacace

Copyright © 2017 Peng Wang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Maritime piracy is posing a genuine threat to maritime transport. The main purpose of simulation is to predict the behaviors of many actual systems, and it has been successfully applied in many fields. But the application of simulation in the maritime domain is still scarce. The rapid development of network and measurement technologies brings about higher accuracy and better availability of online measurements. This makes the simulation paradigm named as dynamic data driven simulation increasingly popular. It can assimilate the online measurements into the running simulation models and ensure much more accurate prediction of the complex systems under study. In this paper, we study how to utilize the online measurements in the agent based simulation of the maritime pirate activity. A new random finite set based data assimilation algorithm is proposed to overcome the limitations of the conventional vectors based data assimilation algorithms. The random finite set based general data model, measurement model, and simulation model are introduced to support the proposed algorithm. The details of the proposed algorithm are presented in the context of agent based simulation of maritime pirate activity. Two groups of experiments are used to practically prove the effectiveness and superiority of the proposed algorithm.

1. Introduction

In modern society, global maritime transport system is a critical part of the world's transportation system. According to information, nearly ninety percent of internationally traded goods are transported by sea [1]. Since 2010, insurance rate has increased more than 10 times for ships transiting the areas where the maritime piracy is rampant. In 2016, the insured losses caused by maritime piracy worldwide amounted to 2.46 billion US dollars and are rising continually. The number of pirate attacks against ships worldwide in 2016 was 191. The increasingly rampant maritime pirate activities have become a great threat to the international maritime transport system throughout the world.

Since entering the new century, many researchers have studied how to fight against the dramatically increased maritime piracy [2]. Many of them offer insight into how the problem may be addressed through global political, security studies, international relations, or some other approaches. These approaches proposed by them target the restoration of

order and law in weak states, but they offer only the most general intelligence while state fragility still remains [3–5]. For example, many approaches only aim to tell where attacks are most likely to happen without giving advice on how to prevent or stop them.

The main purpose of simulation is to predict the behaviors of many actual systems, such as natural disasters, wildfires, and terrorist attacks. The simulation systems have been used to support policy design and operational management in the ground transportation [6] and air transportation field [7] for a very long time. But the applications of simulation systems in the maritime transportation field are scarce.

The agent based dynamic data driven simulation (DDDS) of maritime pirate activity focuses on the accurate prediction of marine pirates in the real world and takes full advantage of the real-time sensors. The primary purpose of the simulation system is to accurately assess the potential outcomes of actions taken by decision-makers (including ship commanders and operators) who interact with the system. It also can be used to find the real-time optimal penetration routes

through designated piracy zones by assimilating the real-time measurements from all kinds of sensors. Data assimilation techniques, which are the foundation of DDDS, can fuse real-time measurements with simulated states to best determine the states of the studied system. Since actions are chosen according to some policy based on the current simulation states of the system, simulation is deservedly used to determine the utility of these actions. The facts that make the data assimilation technology necessary for DDDS of maritime pirate activity are as follows.

Highly Dynamic and Changing Real Operational Context. Future maritime mission planning environments are expected to be complex and distributed due to rapid advancement in cyber-physical systems and the ubiquitous use of intelligent heterogeneous assets. The maritime environmental context is always changing dynamically. The simulation results based on current data may no longer be accurate as additional data is received. Unforeseen pirates can appear in the designated piracy zone and critical rescue equipment can be damaged, thereby making the current simulation results useless. In order to adapt to the dynamic and changing real operational context, we need to use the data assimilation technology to assimilate the changing environmental information into the simulation system.

Imprecise Simulation Data and Models. The accuracy of simulation is determined by many factors, such as data used for initialization and models for describing the real system. Taking the simulation of maritime pirate activity as example, the accuracy of simulation depends on the weather data, merchant vessel data, and piracy vessel data in designated piracy zones. Due to current technical constraints, it is impossible to obtain all the required data without errors. Besides data errors, the simulation model of pirates also brings errors. Being subject to these constraints, the simulation results of the simulation models will certainly be inaccurate. In order to overcome the adverse impacts of the imprecise input data and models, we need the data assimilation technology.

The differences between the simulation system and actual system will continue to expand, if we do not assimilate measurements from the actual system and dynamically adjust the simulation results. We propose a random finite set (RFS) based recursive algorithm for online assimilation of measurements in the presence of noises, data association uncertainty, detection uncertainty, and clutters. We model the simulation states and measurements as RFS and use the RFS based Bayesian inference recursion to assimilate the real-time measurements in time.

The structure of the paper is organized as follows. Section 2 introduces two related works and highlights the gaps of the literature covered by the current algorithms. Section 3 indicates the main RFS based models (including general data model, measurement model, and simulation model) which are the foundation of the proposed data assimilation algorithm. Section 4 introduces the agent based models and the combination of data assimilation and agent. Section 5 presents the detailed design and implementation of the RFS based data assimilation algorithm. Section 6 mentions the

experiment design and the results obtained from the experiments. Section 7 presents the conclusions.

2. Related Works

2.1. Maritime Piracy Simulation. The applications of simulation systems in the maritime transportation domain are especially scarce. The majority of existing works only focus on traffic in national, coastal waters and ports. They tend to use high level equation based models for modeling maritime piracy [8]. A little work has been done to optimize ship routing in the maritime domain, but it did not take the security aspect into consideration [9].

Some researchers attempt to apply mathematical modeling and computer simulation to study the maritime piracy in recent years. In [10], the authors use the discrete event simulator PANOPEA to model the maritime pirates around the Gulf of Aden. They have evaluated the effectiveness of some command and control models. The limitation is that they only consider the application for a specified scenario, and the method and models are not applicable in other scenarios.

In [11], the author employs intelligence reports, meteorological forecasts, and historical pirate incidents to describe Next-generation Piracy Performance Surface (PPSN) model and mainly focuses on predicting the areas around the Horn of Africa. In [12], the authors introduce the Pirate Attack Risk Surface (PARS) model by using the probabilistic models to further improve the PPSN model. Both these two models are numerical models. Since they do not directly model behaviors or interactions of individual ships, their ability for what-if type of simulation and prediction is limited.

In [13], the authors use the agent based modeling framework named as MANA (Map-Aware, Non-Uniform Automata) to find the factors that are important for the escort through the Gulf of Aden. MANA is used for defense related modeling and simulation by specifying the properties of agents. They model the escorting scenario on a tactical level by using MANA. In [14], the authors use the MANA framework to defend vessels against pirate activities by employing nonlegal deterrents. They model the single encounter behaviors in detail, and have not taken the whole system into consideration. The limitation is that it has not taken the measurements of real situation into consideration, which makes it unpractical in applications.

Jakob et al. are the first to use agent based simulation to study the international maritime transport security [15]. In [16], they describe and model the transportation system from a macro perspective. They describe the pirates and piracy countermeasures by using the agent based models. And they drive the agent model by inputting the offline data stored in the database. They conduct the simulation with thousands of agents, and each agent is related to a vessel. They can capture the dynamic of the real transportation system and evaluate the effectiveness of piracy countermeasures. But they only focus on the whole maritime transportation system from a macro perspective but fail to support decision for the ship operator in a tactical level. The data used to drive the simulation are offline data that are stored in the database.

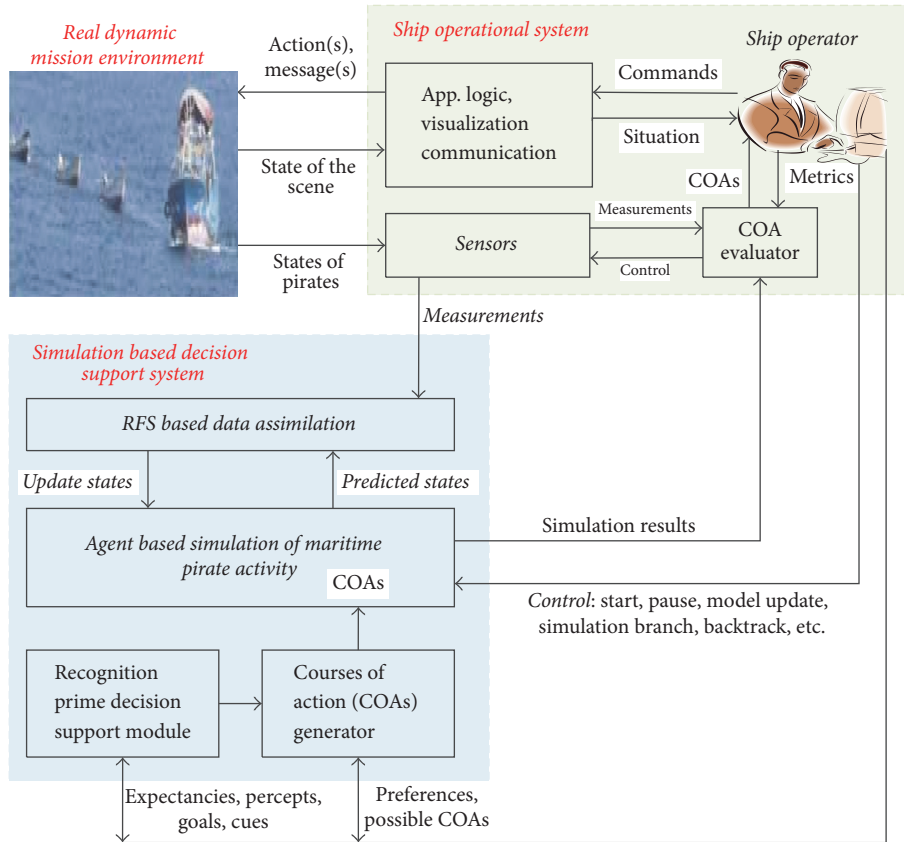


FIGURE 1: Agent and simulation based decision support system for maritime counter-piracy.

They use the stored data to drive the simulation instead of real-time measurements, so the simulation cannot be used to support real-time decision-making.

The agent based DDDS of maritime pirate activity studies the problem of the merchant vessel agent trying to cross or penetrate a piracy-infested area patrolled by one or more pirate vessel agents by assimilating the online measurements into the simulation models. The merchant vessel agent in the simulation can choose its self-relief measures. The simulation system can be used to help minimize the chance of hostile encounter or being hijacked and to avoid detection and interception. So the simulation system can be used by the merchant vessel operators or commanders to penetrate a piracy-infested area. This simulation also can be applied in reality, such as logistics in illegal border crossing, insecure regions, and/or smuggling interdiction. The schematic diagram of the simulation system based decision support system for maritime counter-piracy is as shown in Figure 1.

Many real-world problems would benefit from the ability to assimilate measurements and predict the state of complex systems in real time. Since many complex systems, such as modeling counter-piracy operations, can take on very different states at critical decision points in the simulation, it is necessary for these simulation systems to support branching through either the generation of separate Monte Carlo replications or the use of multiple hypotheses branching techniques that support multiple hypotheses within a single

execution. Multiple hypotheses branching can also be used to explore what-if possibilities or alternative plans. The traditional estimation and prediction control theory is combined with optimistic simulation techniques to support the data assimilation of complex systems. Real-time simulation based decision support system for maritime counter-piracy projects the future through simulation. It uses real-time measurements to calibrate the current state and requires multiple hypothesis what-if branching capabilities to simultaneously explore the effectiveness of various decision options.

Figure 2 is the schematic diagram of data assimilation for agent based DDDS of maritime pirate activity. In the simulation based decision support system, the newly assimilated simulation states of the real system at time t_k are obtained by optimally combining noisy and cluttered measurements with the previously predicted state. Then, the new state estimated is extrapolated to time t_{k+1} to form the next prediction of the system. The prediction/update cycle repeats as time evolves and new measurements become available. This iterative process is shown in Figure 2, where X represents the abstract state of the system. Accurate assimilation must balance the uncertainty of available measurements with unknown or unpredictable behaviors associated with the model.

2.2. DDDS and Data Assimilation. Until now, the simulation models have been using the static information as initial settings to simulate the real systems in the conventional

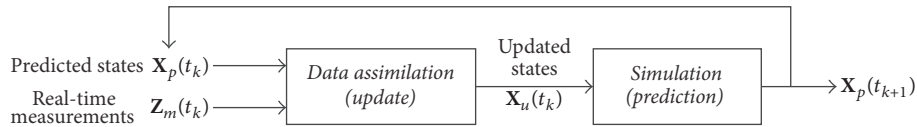


FIGURE 2: The relationship between data assimilation and simulation.

simulation systems. In the dynamic maritime pirate activity modeling, since the static initial parameters fail to capture the real-time environmental changes timely, the simulation states are always quite different from the actual ones of the real system. DDDS has a golden opportunity for development, for the online measurements can reflect the latest updated states of the real system. It means dynamically assimilating online measurements when executing simulations. This idea has been applied in some research areas, such as traffic management, disaster forecasting, environmental science, and wildfire [17, 18]. The simulation system in DDDS paradigm is continually adjusted by assimilating real-time measurements for much more accurate prediction of the studied systems [19]. The rapid development of network and measurement technologies contributes to the higher accuracy and better availability of online measurements, which has facilitated the development of DDDS.

DDDS highly depends on data assimilation technology which is in charge of assimilating online measurements into a running simulation system. Data assimilation has the potential to greatly improve the accuracy of simulation systems. As the simulation models become increasingly complicated, if we still make little usage of online measurements from the studied system, traditional simulations will be largely decoupled from real systems. In data assimilation, the measurements are assimilated into the running simulation models to generate more accurate simulation results [20]. The data assimilation algorithms aim to minimize the differences between the simulated states and the real world. There are two main types of data assimilation algorithms: one is the constant statistical algorithms which include 3DVAR [21] and 4DVAR [22] and the other one is the adaptive statistical algorithms, such as Kalman filter [23], ensemble Kalman filter (EnKF) [24], particle filter (PF) [25], and hierarchical Bayesian method [26]. The data assimilation algorithms that have been successfully used in simulation applications include 3DVAR, 4DVAR, Kalman filter, and PF based algorithms.

The basic principle of 3DVAR and 4DVAR is to avoid the explicit calculation of the gain matrix and make its inversion by using a minimization procedure of the cost function. They are cost function based and use a suited descent algorithm to iteratively evaluate the cost function. The limitation of variational methods is that they use the minimization of a cost function and they do not provide any estimate of predictive uncertainty. Kalman filter and its variance [27, 28] are important and conventional Bayes estimation techniques used in data assimilation. Kalman filter based data assimilation algorithms are optimal for the assimilation of linear Gaussian systems. The requirements of using Kalman filter based data assimilation algorithms are that the

studied systems should be modeled in linear equations and the distributions of processing noises and observation noises are Gaussian. The constraints greatly restrict the applicability of Kalman filter based algorithms in data assimilation for systems that are nonlinear and non-Gaussian. In this paper, we extend the Kalman filter by using RFS based models and unscented transformation.

These conventional data assimilation algorithms fail to support data assimilation in the simulation field because of the nonlinear, non-Gaussian properties of a lot of simulation models. The sequential Monte Carlo method is also named as particle filter (PF). PF based data assimilation has been used for some simulation applications including traffic and wildfire. Particle filter is the exact method for the nonlinear and non-Gaussian applications. It has been the most widely used data assimilation algorithm in simulation applications in the recent years [19]. PF based data assimilation in traffic simulation is presented by Xie et al. in [29] and Wu et al. in [30]. PF based data assimilation in wildfire simulation is presented by Xue et al. in [31, 32]. They all use the same nonparametric statistic inference method based on particle filter, because the particle filter based data assimilation algorithms need no assumptions about the distribution or linearity of the studied simulation systems.

PF based data assimilation algorithms are widely used, because they are the exact methods for the nonlinear and non-Gaussian systems. But they have the following essential disadvantages:

- (i) They are based on the assumption that the studied systems are single dynamic systems that are permanently active. What is more, they use the presumed physical models of individual agents, and the transition of agents from one model to another is less common.
- (ii) They are based on the assumption that there is exactly one measurement at each time and the detection is perfect with no false detections or misdetections. In other words, they assume that the sensors never fail to make a measurement of the state and never pick up false measurements.
- (iii) They are based on the assumption that the measurements are precise and the measurement models are always precise as well. They always model the measurements as points in the measurement space, and the number and ordering of measurements are previously designated.

The standard PF based data assimilation algorithms cannot be used for data assimilation of the dynamic systems that are switching on and off randomly. Switching is quite common in maritime pirate activity, as it did in the case where

moving pirate vessels appear and vanish in the designated areas. The restriction in which the studied system must be a single dynamic system limits the applicability of the PF based algorithms. In the maritime pirate activity, multiple dynamic pirate vessels can appear and vanish during the measurement intervals. The measurement models used in PF based data assimilation algorithm are quite unreasonable, since there are always false detections and misdetections in the real world, and the measurement functions are impossible to be precisely known. The PF based algorithm cannot jointly estimate the number of pirates and the states of every pirate either.

For the agent based simulation of the maritime pirate activity, we employ RFS to assimilate online measurements into the simulation model. A major task for the data assimilation is to dynamically update the simulation states of the real pirate vessels and then provide the updated states to the simulation model for follow-on simulations. The proposed RFS based data assimilation algorithm can overcome the limitations of the standard PF based data assimilation algorithms effectively.

The RFS theory proposed by Goodman et al. as finite set statistics (FISST) is a recently developed theory that unifies much of information fusion under a single Bayesian paradigm [33, 34]. FISST is the first systematic treatment of multisensor multitarget filtering as part of a unified framework for data fusion using random set theory [35, 36]. FISST is a means of directly extending the familiar methodology of single-sensor, single-target Bayes statistics to multisensor multitarget problems. FISST gives some applicable mathematical tools that enable direct application of Bayesian inference. FISST provides the concept of integration/density for set-valued variable. More details can be found in [37].

3. RFS Based Models for Data Assimilation

The data assimilation algorithm based on the RFS theory has markedly widened the application fields of data assimilation. It supports both permanently active systems (i.e., present pirates) and on/off systems (i.e., appearing/vanishing pirates). It supports both the single dynamic system and the multiple dynamic systems. It also supports both the perfect detection and the imperfect detection. It can be applied to both precise measurements and the imprecise measurements. It can jointly estimate the number of agents and the state of each of the agents. We will tell how the proposed method overcomes the three limitations of the traditional PF based algorithms under the background of the DDDS of maritime pirate activity.

The proposed data assimilation algorithm uses two kinds of system models to describe the real system: one is Markov transition density based simulation model and the other is the measurement likelihood based measurement model. Simulation model is used to describe the state transition of the studied system, and the measurement is used to model the sensors. Both of these models are constructed based on the general data model. The RFS based representation of data (including measurements and states) is the foundation of RFS based data assimilation algorithm and the essential difference between the proposed algorithm and the traditional PF based algorithms.

3.1. RFS Based General Data Model. The standard PF based data assimilation assumes precise measurements and measurement models. But the reality is that there is uncertainty in both measurements and measurement models. The uncertainty in the measurements can come from many aspects, including uncertainty due to multiple agents, uncertainty due to unreliability, uncertainty due to false measurements, uncertainty due to deception, uncertainty due to evasive action, and uncertainty due to unknown correlations. The details can be found in [33, 37]. The challenges caused by the uncertainties in the data model have inhibited the maturation of DDDS as a unified, systematic, scientifically founded simulation paradigm. For example, the synthetic aperture radar (SAR) images used in the maritime simulation are statistically uncharacterized and difficult to model. The natural-language statements or rules drawn from knowledge bases are widely used in the maritime pirate activity, and they are unclear and cannot be statistically processed.

The limitations of the standard PF based data assimilation algorithms are caused by the fact that they employ vectors to represent the states and measurements. Clearly, the vector based representation cannot represent all occurrences of multiple agent states and, more importantly, does not admit a meaningful and mathematically consistent notion of estimation error. The commonly used vector based assimilation algorithm is summarized in Figure 3. We can find that it should assume that the dimension and elements order in each vector are equal. It also needs necessary operations outside of the Bayesian recursion to ensure the consistency of the vectors used for assimilation. The measurement processing step is necessary for vector based algorithms. The determination of newly observed measurement and missed measurement is through vector augmentation or truncation which are very computationally intensive and irreversible. It is also a disadvantage of these algorithms, because it makes the computation complex and unstable. Because of the fundamental inconsistency of the vector based representation, stacking individual measurements into a large vector is not a satisfactory representation.

Here we use RFS to represent the multiple agent states and measurements, because the RFS can represent all possible occurrences of multiple agent states, and distance between sets is a well understood concept. The RFS based data assimilation framework is summarized in Figure 4; it enables the joint estimation of errors in cardinality and assimilated states, because it jointly propagates both the estimate of the cardinality of agents and their states. The data association process is automatically dealt with within the FISST framework, and consequently there is no need for data association and measurement management.

In the real operational context of maritime, the number of measurements is a random variable, and there is no information on which is the measurement of the state. Actually, the number of measurements is not fixed and the ordering of measurements is not relevant; the measurement is better to be modeled as a RFS. RFS based data model generalizes the state space model for simulation; it takes into account a more realistic situation where the random number of agents

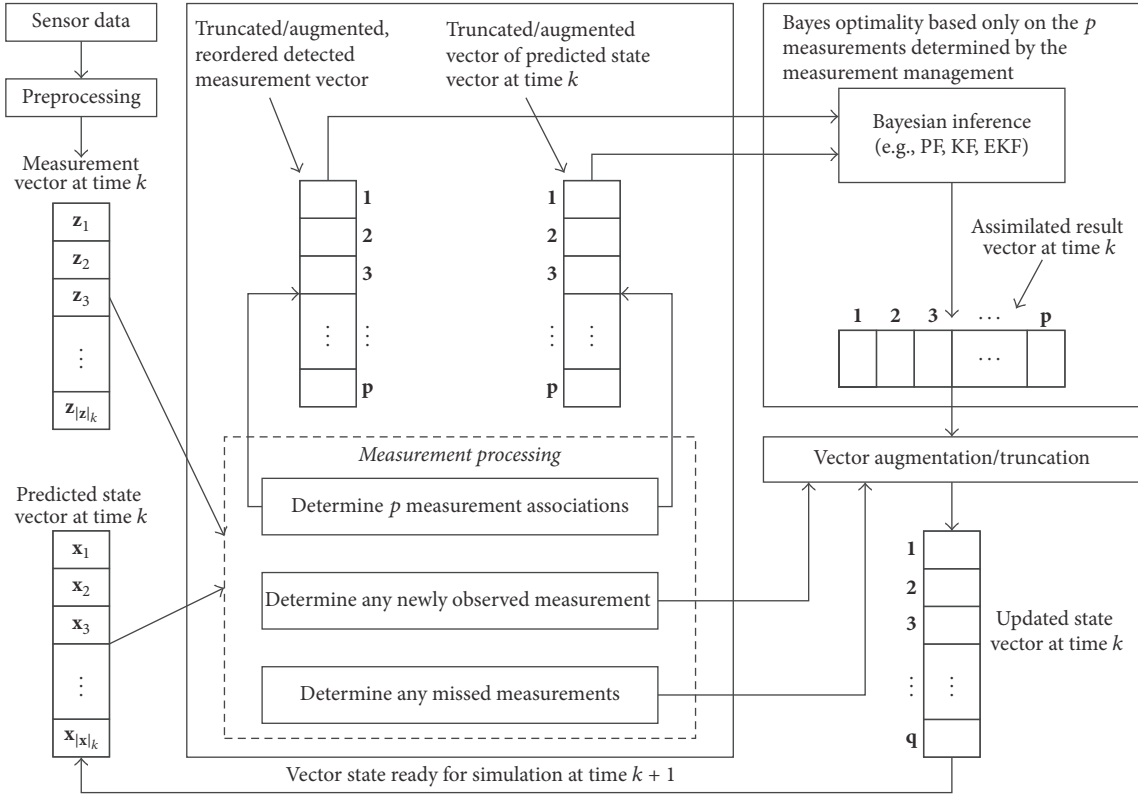


FIGURE 3: Vector based data assimilation and its mathematical restrictions for a single prediction/update cycle.

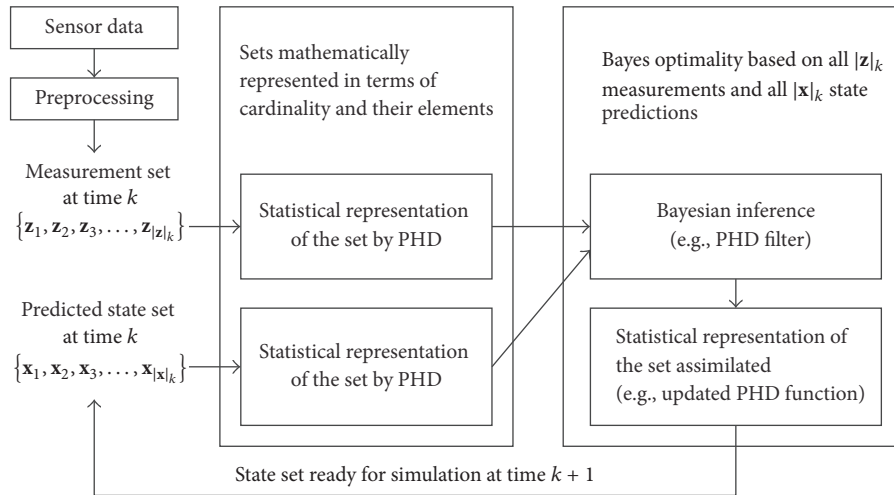


FIGURE 4: RFS based data assimilation for a single prediction/update cycle.

and measurements, detection uncertainty, false alarms, and association uncertainty are all taken into consideration.

A RFS $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n\}$ is a finite set-valued random variable that can be illustrated by a discrete probability distribution $\rho(n) = \Pr\{|\mathbf{X}| = n\}$ and a family of symmetric joint probability densities $f_n(\mathbf{x}_1, \dots, \mathbf{x}_n)$ [33, 38]. The joint probability densities conditioned on cardinality n describe the probability distribution of its elements over the state

space. The discrete probability distribution describes the cardinality of RFS \mathbf{X} and is defined as follows:

$$\begin{aligned} \rho_{\mathbf{X}}(n) &= \Pr\{|\mathbf{X}| = n\} \\ &= \frac{1}{n!} \int f_{\mathbf{X}}(\{\mathbf{y}_1, \dots, \mathbf{y}_n\}) d\mathbf{y}_1 \cdots d\mathbf{y}_n. \end{aligned} \quad (1)$$

The relationship between the distribution of RFS \mathbf{X} and its cardinality is

$$f(\{\mathbf{x}_1, \dots, \mathbf{x}_n\}) = n! \cdot \rho(n) \cdot f_n(\mathbf{x}_1, \dots, \mathbf{x}_n). \quad (2)$$

We assimilate measurements with a RFS variable formal measurement model that describes sensors' behaviors and a RFS variable formal simulation model that models the pirates' behaviors. This means that we use the RFS formulation to treat the collection of individual pirate states as a random finite set-valued state and the collection of individual measurements as a random finite set-valued measurement.

3.2. RFS Based Measurement Model. We propose a general detection type measurement model based on the following assumptions as shown in Figure 5:

- (i) A single sensor with likelihood function $g_{k+1}(\mathbf{z} | \mathbf{x}) = g_{W_{k+1}}(\mathbf{z} - h_{k+1}(\mathbf{x}))$ observes a scene involving an unknown number of pirate agents; here $h_{k+1}(\mathbf{x})$ is the deterministic generation of measurement from state \mathbf{x} .
- (ii) A single pirate agent with state \mathbf{x} generates either a single measurement which means an agent detection, occurring with probability $p_D(\mathbf{x})$, or a missed detection which means no measurement at all, occurring with probability $1 - p_D(\mathbf{x})$.
- (iii) The false alarm process \mathbf{C} is Poisson-distributed in time with expected value λ and distributed in space according to an arbitrary density $c(\mathbf{z})$, and $c_{k|k}(\mathbf{Z}_k) = e^{-\lambda} \lambda_{k|k}^m c_{k|k}(\mathbf{z}_1) \cdots c_{k|k}(\mathbf{z}_m)$ is the probability that a set $\mathbf{Z}_k = \{\mathbf{z}_1, \dots, \mathbf{z}_m\}$ of clutter observations will be generated.

For the given predicted multiple agent states $\mathbf{X}_k = \{\mathbf{x}_1, \dots, \mathbf{x}_n\}$, the random measurement set collected by the sensor will have the form $\mathbf{Z}_k = \Gamma_k(\mathbf{X}_k) \cup \mathbf{C}_k$, where $\Gamma_k(\mathbf{X}_k)$ is the agent detection set and it has the form $\Gamma_k(\mathbf{X}_k) = \Gamma_k(\mathbf{x}_1) \cup \cdots \cup \Gamma_k(\mathbf{x}_n)$; $\Gamma_k(\mathbf{x}_j)$ is the detection set for state \mathbf{x}_j ; and \mathbf{C}_k is Poisson with mean value λ and spatial distribution $c(\mathbf{z})$. We assume that $\Gamma_k(\mathbf{x}_1), \dots, \Gamma_k(\mathbf{x}_n); \mathbf{C}_k$ is statistically independent.

Any given agent can generate either a single measurement or no measurement at all. Consequently, $\Gamma_k(\mathbf{x}_i)$ must have the form $\Gamma_k(\mathbf{x}_i) = \emptyset^{p_D(\mathbf{x}_i)} \cap \{\mathbf{Z}_i\}$. Here, $\mathbf{Z}_i = h(\mathbf{x}_i, \mathbf{W}_i)$ is the sensor-noise model associated with i th state \mathbf{x}_i ; and \emptyset^p is the discrete random subset of the baseline measurement space \mathcal{O}_0 . Detection uncertainty is modeled as Bernoulli RFS as follows:

$$\Pr(\emptyset^p = \mathbf{T}) = \begin{cases} 1 - p_D, & \text{if } \mathbf{T} = \emptyset, \\ p_D \cdot g(\mathbf{Z}_i), & \text{if } \mathbf{T} = \mathcal{O}_0, \\ 0, & \text{if otherwise.} \end{cases} \quad (3)$$

The fact that the false alarm process \mathbf{C}_k is Poisson means that it must have the form $\mathbf{C}_k = \{\mathbf{C}_1, \dots, \mathbf{C}_M\}$, where $M = |\mathbf{C}_k|$ is a random nonnegative integer with probability distribution $p_M(m) = e^{-\lambda} \lambda^m / m!$; $\mathbf{C}_1, \dots, \mathbf{C}_M$ are

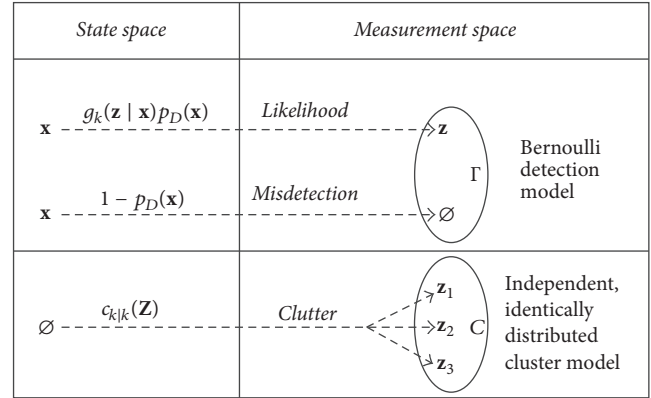


FIGURE 5: The major assumptions of RFS based measurement model.

independent, and they are identically distributed random measurement vectors with density $c(\mathbf{z}) = g_C(\mathbf{z})$.

The likelihood $g_k(\mathbf{Z}_k | \mathbf{X}_k)$ of multiple agents describes the sensor measurement for multiple agents and characterizes the detections, clutters, and agent generated observations. For the state set \mathbf{X}_k , the probability density distribution of receiving the measurement set \mathbf{Z}_k is described by the true likelihood function. The true likelihood function for arbitrary number of agents with missed detections and clutters is

$$\begin{aligned} g_k(\mathbf{Z}_k | \mathbf{X}_k) &= g_k((\Gamma_k(\mathbf{X}_k) \cup \mathbf{C}_k) | \mathbf{X}_k) \\ &= \sum_{W \subseteq \mathbf{Z}_k} p_{\Gamma_k(\mathbf{X}_k)}(W | \mathbf{X}_k) p_{\mathbf{C}_k}(\mathbf{Z}_k - W) \\ &= e^\lambda g_C(\mathbf{Z}_k) \cdot g_k(\emptyset | \mathbf{X}_k) \\ &\quad \cdot \sum_{\theta} \prod_{i: \theta(i) > 0} \frac{p_D(\mathbf{x}_i) \cdot g_k(\mathbf{z}_{\theta(i)} | \mathbf{x}_i)}{(1 - p_D(\mathbf{x}_i)) \cdot \lambda c(\mathbf{z}_{\theta(i)})}; \end{aligned} \quad (4)$$

here, $g_C(\mathbf{Z}_k) = e^{-\lambda} \prod_{\mathbf{z} \in \mathbf{Z}_k} \lambda c(\mathbf{z})$ and $g_k(\emptyset | \mathbf{X}_k) = e^{-\lambda} \prod_{\mathbf{x} \in \mathbf{X}_k} (1 - p_D(\mathbf{x}))$; the summation is taken over all association hypotheses $\theta: \{1, \dots, n\} \rightarrow \{0, 1, \dots, m\}$.

3.3. RFS Based Simulation Model. In the agent based simulation of maritime pirate activity, the agents number can be constantly varying because agents can appear and disappear. For example, completely new agents can enter a scene spontaneously, as when a pirate vessel or military vessel suddenly appears in the piracy zones. Existing agents can give rise to new agents through spawning, as when a pirate mother ship acts as a floating platform from which smaller pirate vessels are launched and when an apparently single agent at a distance resolves into multiple, closely spaced agents and so on. Agents can likewise leave a scene, as when disappearing from the piracy zones or behind some other occlusion, or they can be damaged or destroyed. All of these instances of multiple agent dynamics should be explicitly modeled by using RFS. If multiple agents are related to birth, death, spawning, merging, and so on, the standard PF based assimilation algorithms are inapplicable, because they fail to accurately estimate the number of agents. In DDDS of maritime

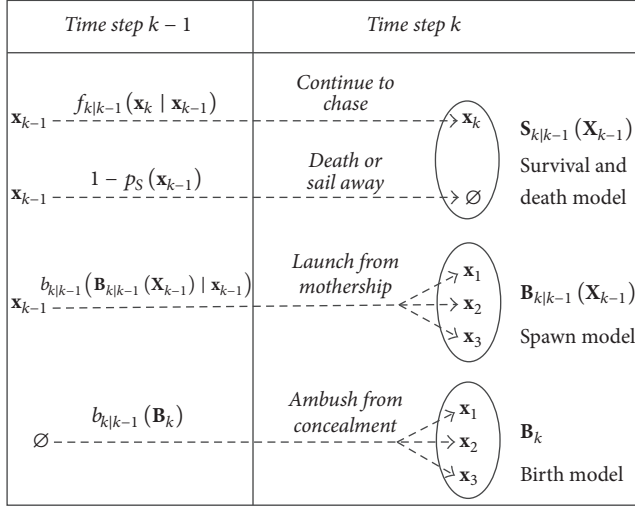


FIGURE 6: A summary of RFS based simulation model of pirate agents.

pirate activity, we use the unified representation of pirate's states to get a complete description of the simulation model.

We model the time evolution of the multiple agent states (e.g., the states of pirate vessels) by employing RFS. We take the agent birth, death, spawn, and motion into consideration. The RFS based simulation model is based on the following assumptions as shown in Figure 6:

- (i) The likelihood that a single agent will have state vector \mathbf{x}_k at time step k if it had state vector \mathbf{x}_{k-1} at time step $k-1$ is described by a Markov transition density $f_{k|k-1}(\mathbf{x}_k | \mathbf{x}_{k-1})$ corresponding to a single agent simulation model $\mathbf{X}_{k|k-1} = \varphi_{k-1}(\mathbf{x}_{k-1}, \mathbf{V}_{k-1})$. Hereafter this is assumed for notational simplicity to be additive: $\mathbf{X}_{k|k-1} = \varphi_{k-1}(\mathbf{x}_{k-1}) + \mathbf{V}_{k-1}$.
- (ii) A single agent with state \mathbf{x}_{k-1} at time step $k-1$ has a probability $p_S(\mathbf{x}_k) \stackrel{\text{abbr.}}{=} p_S^{k|k-1}(\mathbf{x}_{k-1})$ of surviving into time step k .
- (iii) A single pirate agent with state \mathbf{x}_{k-1} at time step $k-1$ has a probability density $b(\mathbf{B}_{k|k-1}(\mathbf{X}_{k-1}) | \mathbf{x}_{k-1}) \stackrel{\text{abbr.}}{=} b_{k|k-1}(\mathbf{B}_{k|k-1}(\mathbf{X}_{k-1}) | \mathbf{x}_{k-1})$ of spawning a set $\mathbf{B}_{k|k-1}(\mathbf{X}_{k-1})$ of new agents at time step k , as when new pirate vessels launch from a mother ship.
- (iv) The probability density that new agents with state set \mathbf{B}_k will spontaneously appear at time k is $b(\mathbf{B}_k) \stackrel{\text{abbr.}}{=} b_{k|k-1}(\mathbf{B}_k)$, as when some pirate vessels ambush from concealments or enter the zone for the first time.

The multiple agent Markov density $f_{k|k-1}(\mathbf{X}_k | \mathbf{X}_{k-1})$ characterizes the time evolution of the multiple agent states and takes the agent births, deaths, and motions into consideration. Here we will give the true multiple agent Markov density function $f_{k|k-1}(\mathbf{X}_k | \mathbf{X}_{k-1})$ for the proposed simulation model, where $\mathbf{X}_k = \{\mathbf{x}_1, \dots, \mathbf{x}_n\}$ and

$\mathbf{X}_{k-1} = \{\mathbf{x}'_1, \dots, \mathbf{x}'_{n_{k-1}}\}$. The proposed representation of agents' states has the mathematical form

$$\mathbf{X}_k = \mathbf{S}_{k|k-1}(\mathbf{X}_{k-1}) \cup \mathbf{B}_{k|k-1}(\mathbf{X}_{k-1}) \cup \mathbf{B}_k, \quad (5)$$

where \mathbf{X}_k is the predicted state set, $\mathbf{S}_{k|k-1}(\mathbf{X}_{k-1})$ are persisting agents that are continuing to chase the merchant, $\mathbf{B}_{k|k-1}(\mathbf{X}_{k-1})$ are spawned agents that are launched from the mother ship, \mathbf{B}_k are spontaneous agents that are ambushing from concealments or enter the designated zone for the first time, $\mathbf{S}_{k|k-1}(\mathbf{X}_{k-1})$ have the form $\mathbf{S}_{k|k-1}(\mathbf{X}_{k-1}) = \mathbf{S}_{k|k-1}(\mathbf{x}'_1) \cup \dots \cup \mathbf{S}_{k|k-1}(\mathbf{x}'_{n_{k-1}})$, and $\mathbf{S}_{k|k-1}(\mathbf{x}'_i)$ is the prediction of \mathbf{x}'_i . $\mathbf{B}_{k|k-1}(\mathbf{X}_{k-1})$ have the form $\mathbf{B}_{k|k-1}(\mathbf{X}_{k-1}) = \mathbf{B}_{k|k-1}(\mathbf{x}'_1) \cup \dots \cup \mathbf{B}_{k|k-1}(\mathbf{x}'_{n_{k-1}})$; $\mathbf{B}_{k|k-1}(\mathbf{x}'_i)$ is spawned by \mathbf{x}'_i which means that it is launched from the mother ship which has the state of \mathbf{x}'_i . We also assume that $\mathbf{S}_{k|k-1}(\mathbf{x}'_1), \dots, \mathbf{S}_{k|k-1}(\mathbf{x}'_{n_{k-1}})$, $\mathbf{B}_{k|k-1}(\mathbf{x}'_1), \dots, \mathbf{B}_{k|k-1}(\mathbf{x}'_{n_{k-1}})$, and \mathbf{B}_k are statistically independent.

The RFS based simulation model is identical in mathematical form to the RFS based measurement model. The corresponding true multiple agent Markov density is

$$\begin{aligned} & f_{k|k-1}(\mathbf{X}_k | \mathbf{X}_{k-1}) \\ &= f_{k|k-1}((\mathbf{S}_{k|k-1}(\mathbf{X}_{k-1}) \cup \mathbf{B}_{k|k-1}(\mathbf{X}_{k-1}) \cup \mathbf{B}_k) | \\ & \mathbf{X}_{k-1}) = \sum_{W_1 \cap W_2 = \emptyset, W_2 \cup W_2 \subseteq \mathbf{X}_k} p_{\mathbf{S}_{k|k-1}}(W_1 | \mathbf{X}_{k-1}) \\ & \cdot p_{\mathbf{B}_{k|k-1}}(W_2 | \mathbf{X}_{k-1}) p_{\mathbf{B}_k}(\mathbf{X}_k - W_1 - W_2) \\ &= e^{\mu(\mathbf{X}_{k-1})} f_{\mathbf{B}_{k|k-1}(\mathbf{X}_{k-1})}(\mathbf{X}_k) \cdot f_{k|k-1}(\emptyset | \mathbf{X}_{k-1}) \\ & \cdot \sum_{\theta} \prod_{i: \theta(i) > 0} \frac{p_S(\mathbf{x}'_i) \cdot f_{k|k-1}(\mathbf{x}_{\theta(i)} | \mathbf{x}'_i)}{(1 - p_S(\mathbf{x}'_i)) \cdot \mu(\mathbf{X}_{k-1}) b(\mathbf{x}_{\theta(i)} | \mathbf{X}_{k-1})}. \end{aligned} \quad (6)$$

The summation is taken over all association hypotheses $\theta: \{1, \dots, n_{k-1}\} \rightarrow \{0, 1, \dots, n_k\}$, where

$$\begin{aligned} & f_{k|k-1}(\emptyset | \mathbf{X}_{k-1}) = e^{-\mu(\mathbf{X}_{k-1})} \prod_{\mathbf{x}_{k-1} \in \mathbf{X}_{k-1}} (1 - p_S(\mathbf{x}_{k-1})), \\ & \mu(\mathbf{X}_{k-1}) = \mu_0 + \mu_1(\mathbf{x}') + \dots + \mu_{n_{k-1}}(\mathbf{x}'), \\ & b(\mathbf{x} | \mathbf{X}_{k-1}) \\ &= \frac{\mu_0 b_0(\mathbf{x}) + \mu_1(\mathbf{x}') b(\mathbf{x} | \mathbf{x}'_1) + \dots + \mu_{n_{k-1}}(\mathbf{x}') b(\mathbf{x} | \mathbf{x}'_{n_{k-1}})}{\mu_0 + \mu_1(\mathbf{x}') + \dots + \mu_{n_{k-1}}(\mathbf{x}')}. \end{aligned} \quad (7)$$

Here, μ_0 is the expected number of spontaneously generated new agents and $b_0(\mathbf{x})$ is their physical distribution. Also, $\mu(\mathbf{x}'_i)$ is the expected number of new agents spawned by an agent with previous state \mathbf{x}'_i , and $b(\mathbf{x} | \mathbf{x}'_i)$ is their physical distribution.

The calculation of the true Markov density is impractical. Here, we use another new variable to deal with the complex computation. The difficulty of RFS based Bayesian inference for multiple agents is its computational complexity. To solve this problem, Mahler proposed the PHD (probability hypothesis density) filter [34]. The PHD $D_{k|k}(\mathbf{x} | \mathbf{Z}_k)$ of the posterior

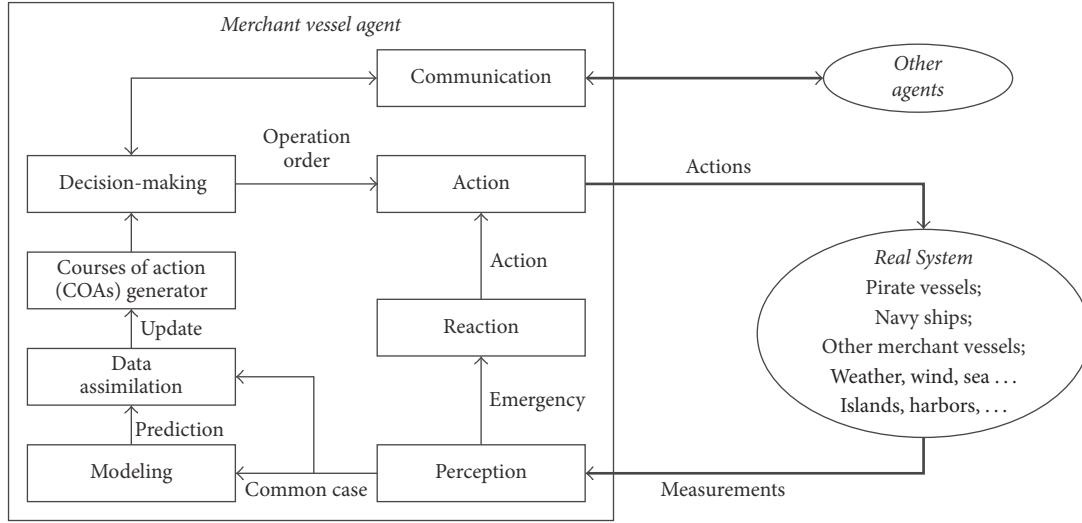


FIGURE 7: Interactions between merchant vessel agent and real system.

probability density $f_{k|k}(\mathbf{X}_k | \mathbf{Z}_k)$ is a probability density function defined on the single agent state $\mathbf{x} \in \mathbf{X}_0$ as follows:

$$D_{k|k} \stackrel{\text{abbr.}}{=} D_{k|k}(\mathbf{x} | \mathbf{Z}_k) \triangleq \int f_{k|k}(\{\mathbf{x}\} \cup \mathbf{X}_k | \mathbf{Z}_k) \delta \mathbf{X}_k. \quad (8)$$

In point theory, $D_{k|k}(\mathbf{x})$ is defined as the intensity density [39]. It is not a probability density. $D_{k|k}(\mathbf{x})$ represents the density of expected number of points at \mathbf{x} . Given any region \mathbf{S} of single pirate agent state space \mathbf{X}_0 , the integral $\int_{\mathbf{S}} D_{k|k}(\mathbf{x}) d\mathbf{x} = v_{k|k}$ is the expected number of pirate agents in \mathbf{S} , where $v_{k|k} \in \mathbb{R}$ is the expected number of objects in the state space \mathbf{S} . In particular, if $\mathbf{S} = \mathbf{X}_0$ is the entire state space, then $N_{k|k} \triangleq \int D_{k|k}(\mathbf{x}) d\mathbf{x}$ is the total expected number of pirate agents in the scene.

Here, we assume that then PHD of the RFS \mathbf{B}_k for the new born pirate agents is a Gaussian mixture defined as follows:

$$\gamma_k(\mathbf{x}) = \sum_{i=1}^{J_{\gamma,k}} w_{\gamma,k}^{(i)} N(\mathbf{x}; \mathbf{m}_{\gamma,k}^{(i)}, \mathbf{P}_{\gamma,k}^{(i)}), \quad (9)$$

where $w_{\gamma,k}^{(i)}$, $\mathbf{m}_{\gamma,k}^{(i)}$, and $\mathbf{P}_{\gamma,k}^{(i)}$ are the weights, means, and covariance matrices of the mixture birth PHD.

4. Agent Model of Vessels

Multiple agent based simulation models are more suitable for characterizing the complex systems. They can capture the properties of reasoning processes. The agent based modeling methods are perfect for the simulation of maritime pirate activity. Because there are a lot of priorities, objectives, constraints, and complex scheduling dependencies in the actual maritime pirate activities, the agent based simulation has been applied to solve these problems successfully.

Here, the agent based simulation models are used to represent the behaviors of the real pirate vessels. The pirate vessels are characterized by autonomous agents. These agent

models describe the movement and interactions of different kinds of pirate vessels. If there are no interactions between the vessel agents, each vessel agent pursues its individual goals. But just like the real pirate vessels, the pirate vessel agents can also interact with other kinds of vessels. And these interactions are difficult to model by using the mathematical models or DEVS.

The data assimilation algorithm is implemented in the merchant vessel agent model. By assimilating the measurements from the real system, the merchant vessel agent can interact with the real system more conveniently and effectively. The interactions between the merchant agent and the real system are shown in Figure 7.

5. RFS Based Data Assimilation Algorithm

We propose using a new RFS based data assimilation algorithm by combining the basic ideas of Gaussian mixture approximation, Cardinalized Probability Hypothesis Density (CPHD) filter, and the unscented Kalman filter [40] together. Unlike the standard unscented Kalman filter, the proposed algorithm represents the states by RFS rather than by vector.

Let $p_k(\mathbf{X} | \mathbf{Z}_{1:k})$ denote the multiple agent posterior probability density. Since we have got the RFS based measurement model and simulation model, we can use these two models to get the optimal Bayes filter. The optimal Bayesian inference is captured by the recursion as follows:

$$\begin{aligned} & p_{k|k-1}(\mathbf{X}_k | \mathbf{Z}_{1:k-1}) \\ &= \int f_{k|k-1}(\mathbf{X}_k | \mathbf{X}) p_{k-1}(\mathbf{X} | \mathbf{Z}_{1:k-1}) \mu_S(d\mathbf{X}), \\ & p_k(\mathbf{X}_k | \mathbf{Z}_{1:k}) \\ &= \frac{g_k(\mathbf{Z}_k | \mathbf{X}_k) p_{k|k-1}(\mathbf{X}_k | \mathbf{Z}_{1:k-1})}{\int g_k(\mathbf{Z}_k | \mathbf{X}) p_{k|k-1}(\mathbf{X} | \mathbf{Z}_{1:k-1}) \mu_S(d\mathbf{X})}, \end{aligned} \quad (10)$$

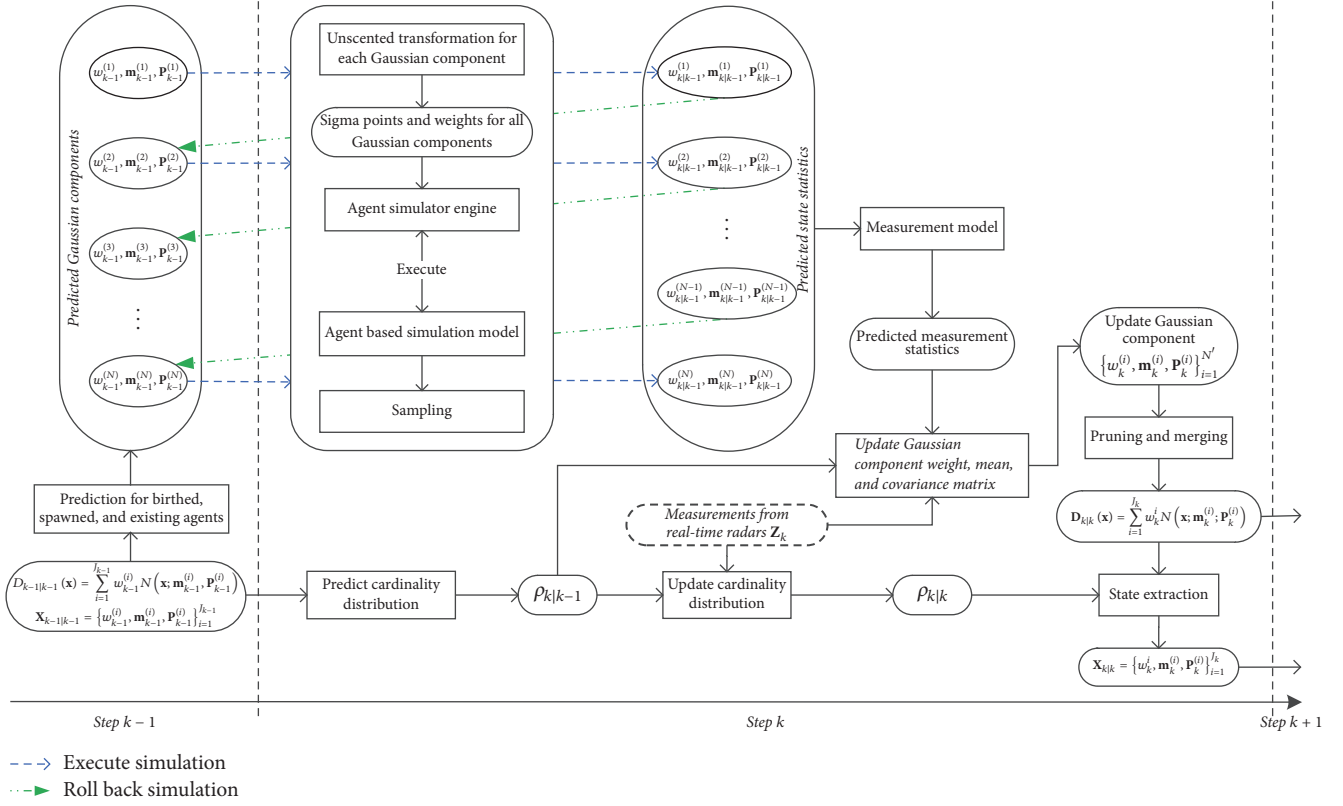


FIGURE 8: Gaussian mixture based implementation of RFS based data assimilation algorithm.

where μ_S is the reference measure on all finite subsets of the measurement space [41].

The CPHD filter is introduced by Mahler in [42], and it propagates both the PHD and the discrete distribution of the pirates' number. The discrete distribution of pirate numbers can be arbitrary.

Let $D_{k|k}(\mathbf{x} | \mathbf{Z}_{1:k}) \stackrel{\text{abbr.}}{=} D_{k|k}(\mathbf{x})$ denote the PHD associated with the posterior multiple agent states, and let $\rho_{k|k}(n | \mathbf{Z}_{1:n}) \stackrel{\text{abbr.}}{=} \rho_{k|k}(n)$ denote the posterior distribution of pirates' number at time k . Also, let $\rho_{\Gamma,k|k-1}(n) \stackrel{\text{abbr.}}{=} \rho_{\Gamma}(n)$ denote the distribution of new agents' number at time k . The proposed data assimilation algorithm propagates both the cardinality distribution $\rho_{k|k}(n)$ and the PHD $D_{k|k}(\mathbf{x})$.

The main idea of the proposed data assimilation algorithm is shown in Figure 8. It is a rollback based algorithm. In this algorithm, the Gaussian components are recursively generated by the rollback based sampling step. The rollback operation means reversing or undoing something. The rollback in this paper has its natural meaning which means restoring the simulation to a previous state. The agent based simulation system is used in the prediction step of the data assimilation algorithm. The rollback strategy is used in the simulation running during the prediction step. In the prediction step, the agent based simulation system is the parallel system of the real system. The update step is the central step of the data assimilation algorithm. The update step uses the

predicted measurement statistics, real-time measurements, and predicted cardinality as inputs.

5.1. Predict with RFS Based Simulation Model. If the number of pirate agents that continue to attack the merchant agent is j , the predicted cardinality of these survived agents will be

$$\begin{aligned} \rho_{S,k|k-1}(j) &= \sum_{\ell=j}^{\infty} \frac{C_{\ell}^j \langle p_S, D_{k-1|k-1} \rangle^j \langle 1 - p_S, D_{k-1|k-1} \rangle^{\ell-j}}{\langle 1, D_{k-1|k-1} \rangle} \rho_{k-1|k-1}(\ell) \\ &= \sum_{\ell=j}^{\infty} \frac{\ell!}{j! (\ell-j)!} p_S^j (1-p_S)^{\ell-j} \rho_{k-1|k-1}(\ell). \end{aligned} \quad (11)$$

Then the predicted distribution of pirates' number can be characterized by the following convolution:

$$\begin{aligned} \rho_{k|k-1}(n) &= \sum_{j=0}^n \rho_{\Gamma,k}(n-j) \rho_{S,k|k-1}[D_{k-1|k-1}, \rho_{k-1|k-1}](j) \\ &= \sum_{j=0}^n \rho_{S,k|k-1}(j) \rho_{\Gamma}(n-j). \end{aligned} \quad (12)$$

Based on the RFS based simulation model of pirate agent movement, the predicted PHD is given as follows:

$$D_{k|k-1}(\mathbf{x}) = \gamma_{k|k-1}(\mathbf{x}) + p_S \int f_{k|k-1}(\mathbf{x} | \mathbf{x}') D_{k-1|k-1}(\mathbf{x}') d\mathbf{x}', \quad (13)$$

where $\gamma_{k|k-1}(\mathbf{x})$ denotes the PHD of the RFS of new agents appearing between time $k-1$ and time k .

Since the Markov transition density of the single vessel agent is nonlinear, here we use the unscented transformation to approximate the moments of $f_{k|k-1}(\mathbf{x} | \mathbf{x}')$ since we know it can be transformed into a linear-Gaussian model by using the unscented transformation.

Under the assumption that the posterior PHD at time $k-1$ is the Gaussian mixture denoted by $D_{k-1|k-1}(\mathbf{x}) = \sum_{i=1}^{J_{k-1}} w_{k-1}^{(i)} N(\mathbf{x}; \mathbf{m}_{k-1}^{(i)}, \mathbf{P}_{k-1}^{(i)})$, the predicted PHD $D_{k|k-1}(\mathbf{x})$ is also a Gaussian mixture under the assumption $D_{k|k-1}(\mathbf{x}) = D_{S,k|k-1}(\mathbf{x}) + \gamma_k(\mathbf{x})$, where $\gamma_k(\mathbf{x})$ is given as in (9) and the predicted PHD of survival agents is

$$\begin{aligned} D_{S,k|k-1}(\mathbf{x}) &= p_S \int f_{k|k-1}(\mathbf{x} | \mathbf{x}') D_{k-1|k-1}(\mathbf{x}') d\mathbf{x}' \\ &= p_S \sum_{j=1}^{J_{k-1}} w_{k-1}^{(j)} N(\mathbf{x}; \mathbf{m}_{S,k|k-1}^{(j)}, \mathbf{P}_{S,k|k-1}^{(j)}). \end{aligned} \quad (14)$$

For each Gaussian component, we should compute the predicted mean value $\mathbf{m}_{S,k|k-1}^{(j)}$ and covariance $\mathbf{P}_{S,k|k-1}^{(j)}$ at time k by using the unscented transformation and the simulation model. For the i th Gaussian component, its predicted mean value $\mathbf{m}_{S,k|k-1}^{(i)}$ and predicted covariance $\mathbf{P}_{S,k|k-1}^{(i)}$ are computed by the following steps. Here, we assume that the mean of the i th Gaussian component is $\boldsymbol{\mu}_{k-1}^{(i)}$ and its covariance is $\mathbf{C}_{k-1}^{(i)}$.

Step 1. Choose the set of sigma points and their corresponding weights denoted by $\{\mathbf{x}_{k-1}^{(l)}, u^{(l)}\}_{l=0}^L$ to approximate mean $\boldsymbol{\mu}_k^{(i)}$ and covariance $\mathbf{C}_k^{(i)}$ as introduced in [43].

Step 2. Run the agent based simulation system to simulation time t_k and get the transformed sigma points denoted by $\mathbf{x}_{k|k-1}^{(l)} := \varphi_k(\mathbf{x}_{k-1}^{(l)}, \mathbf{v}_{k-1}^{(l)})$, $l = 0, \dots, L$.

Step 3. Compute the predicted state statistics:

$$\begin{aligned} \mathbf{m}_{k|k-1}^{(i)} &= \sum_{l=0}^L u^{(l)} \mathbf{x}_{k|k-1}^{(l)}, \\ \mathbf{P}_{k|k-1}^{(i)} &= \sum_{l=0}^L u^{(l)} (\mathbf{x}_{k|k-1}^{(l)} - \mathbf{m}_{k|k-1}^{(i)}) (\mathbf{x}_{k|k-1}^{(l)} - \mathbf{m}_{k|k-1}^{(i)})^T. \end{aligned} \quad (15)$$

Step 4. Choose the sigma points $\mathbf{x}_{k|k-1}^{(1)}, \dots, \mathbf{x}_{k|k-1}^{(L)}$ and their corresponding weights $u^{(1)}, \dots, u^{(L)}$ to approximate mean $\mathbf{m}_{k|k-1}^{(i)}$ and covariance $\mathbf{P}_{k|k-1}^{(i)}$.

Step 5. Generate the transformed sigma points $\mathbf{z}_{k|k-1}^{(l)} := h_k(\mathbf{x}_{k|k-1}^{(l)}, \boldsymbol{\varepsilon}_k^{(l)})$, $l = 0, \dots, L$.

Step 6. Calculate the predicted measurements and their corresponding characteristic:

$$\begin{aligned} \boldsymbol{\eta}_{k|k-1}^{(i)} &= \sum_{l=0}^L u^{(l)} \mathbf{z}_{k|k-1}^{(l)}, \\ \mathbf{S}_k^{(i)} &= \sum_{l=0}^L u^{(l)} (\mathbf{z}_{k|k-1}^{(l)} - \boldsymbol{\eta}_{k|k-1}^{(i)}) (\mathbf{z}_{k|k-1}^{(l)} - \boldsymbol{\eta}_{k|k-1}^{(i)})^T, \\ \mathbf{G}_k^{(i)} &= \sum_{l=0}^L u^{(l)} (\mathbf{x}_{k|k-1}^{(l)} - \mathbf{m}_{k|k-1}^{(i)}) (\mathbf{z}_{k|k-1}^{(l)} - \boldsymbol{\eta}_{k|k-1}^{(i)})^T. \end{aligned} \quad (16)$$

Step 7. Compute the Kalman gain and posterior covariance matrix:

$$\begin{aligned} \mathbf{K}_k^{(i)} &= \mathbf{G}_k^{(i)} [\mathbf{S}_k^{(i)}]^{-1}, \\ \mathbf{P}_{k|k}^{(i)} &= \mathbf{P}_{k|k-1}^{(i)} - \mathbf{G}_k^{(i)} [\mathbf{S}_k^{(i)}]^{-1} [\mathbf{G}_k^{(i)}]^T. \end{aligned} \quad (17)$$

5.2. Update with Online Measurements. Without measurements, the covariance matrix grows exponentially over time. Assimilating the real-time measurements can be regarded as calibrating or estimating the real-time state of the simulation. The update step of the proposed data assimilation algorithm is depending on the Bayesian inference. The updated cardinality is as follows:

$$\rho_{k|k}(n) = \frac{\Upsilon_{k,D_{k|k-1};\mathbf{Z}_k}^0(n) \rho_{k|k-1}(n)}{\langle \Upsilon_{k,D_{k|k-1};\mathbf{Z}_k}^0, \rho_{k|k-1} \rangle}. \quad (18)$$

Under the assumption that the predicted PHD $D_{k|k-1}(\mathbf{x})$ at time k is the Gaussian mixture denoted by $D_{k|k-1}(\mathbf{x}) = \sum_{i=1}^{J_{k|k-1}} w_{k|k-1}^{(i)} N(\mathbf{x}; \mathbf{m}_{k|k-1}^{(i)}, \mathbf{P}_{k|k-1}^{(i)})$, the updated PHD $D_{k|k}(\mathbf{x})$ can also be represented by the Gaussian mixture as follows:

$$\begin{aligned} D_{k|k}(\mathbf{x}) &= \frac{g_k(\mathbf{Z}_k | \mathbf{X}_k) D_{k|k-1}(\mathbf{x})}{\int g_k(\mathbf{Z}_k | \mathbf{X}_k) D_{k|k-1}(\mathbf{x}) d\mathbf{x}} \\ &\equiv \left[\sum_{\mathbf{z} \in \mathbf{Z}_k} \frac{p_D g_k(\mathbf{z} | \mathbf{x})}{\int p_D g_k(\mathbf{z} | \mathbf{x}) D_{k|k-1}(\mathbf{x}) d\mathbf{x} + c(\mathbf{z})} + 1 \right. \\ &\quad \left. - p_D \right] D_{k|k-1}(\mathbf{x}) \\ &= \left[\sum_{\mathbf{z} \in \mathbf{Z}_k} \frac{\langle \Upsilon_{k,D_{k|k-1};\mathbf{Z}_k \setminus \{\mathbf{z}\}}^1, \rho_{k|k-1} \rangle p_D \langle 1, c_k \rangle g_k(\mathbf{z} | \mathbf{x})}{\langle \Upsilon_{k,D_{k|k-1};\mathbf{Z}_k}^0, \rho_{k|k-1} \rangle c_k(\mathbf{z})} \right. \end{aligned}$$

$$\begin{aligned}
& + (1 - p_D) \left[\frac{\langle \Upsilon_{k, D_{k|k-1}; \mathbf{z}_k}^1, \rho_{k|k-1} \rangle}{\langle \Upsilon_{k, D_{k|k-1}; \mathbf{z}_k}^0, \rho_{k|k-1} \rangle} \right] D_{k|k-1}(\mathbf{x}) = (1 \\
& - p_D) \frac{\langle \Upsilon_{k, D_{k|k-1}; \mathbf{z}_k}^1, \rho_{k|k-1} \rangle}{\langle \Upsilon_{k, D_{k|k-1}; \mathbf{z}_k}^0, \rho_{k|k-1} \rangle} D_{k|k-1}(\mathbf{x}) \\
& + \sum_{\mathbf{z} \in \mathbf{Z}_k} \sum_{j=1}^{J_{k|k-1}} w_k^{(j)}(\mathbf{z}) N(\mathbf{x}; \mathbf{m}_k^{(j)}(\mathbf{z}), \mathbf{P}_k^{(j)}),
\end{aligned} \tag{19}$$

where

$$\begin{aligned}
w_{k|k-1} &= \left[w_{k|k-1}^{(1)}, \dots, w_{k|k-1}^{(J_{k|k-1})} \right]^T, \\
q_k(\mathbf{z}) &= \left[q_k^{(1)}(\mathbf{z}), \dots, q_k^{(J_{k|k-1})}(\mathbf{z}) \right]^T, \\
q_k^{(j)}(\mathbf{z}) &= N(\mathbf{z}; \boldsymbol{\eta}_{k|k-1}^{(j)}, \mathbf{S}_{k|k-1}^{(j)}), \\
w_k^{(j)} &= p_D w_{k|k-1}^{(j)} q_k^{(j)}(\mathbf{z}) \frac{\langle \Upsilon_{k, D_{k|k-1}; \mathbf{z}_k \setminus \{\mathbf{z}\}}^1, \rho_{k|k-1} \rangle}{\langle \Upsilon_{k, D_{k|k-1}; \mathbf{z}_k}^0, \rho_{k|k-1} \rangle} \frac{g_k(\mathbf{z} | \mathbf{x})}{c(\mathbf{z})}, \\
\mathbf{m}_k^{(j)} &= \mathbf{m}_{k|k-1}^{(j)} + \mathbf{K}_k^{(j)}(\mathbf{z} - \boldsymbol{\eta}_{k|k-1}^{(j)}).
\end{aligned} \tag{20}$$

$\langle \rho_1, \rho_2 \rangle$ denotes the inner product defined by $\langle \rho_1, \rho_2 \rangle = \sum_{n=0}^{\infty} \rho_1(n) \rho_2(n)$; $g_k(\cdot | \mathbf{x})$ is the single pirate agent measurement likelihood density at time k based on state \mathbf{x} ; $c_k(\cdot)$ is the PHD of clutter measurements at time k . The sequence $\Upsilon_{k, D, \mathbf{Z}^{(n)}}^u(n)$ is defined for $u \in \{0, 1\}$ as follows:

$$\begin{aligned}
\Upsilon_{k, D, \mathbf{Z}^{(n)}}^u(n) &= \sum_{j=0}^{\min(|\mathbf{Z}|, n)} (|\mathbf{Z}| - j)! \rho_{c, k}(|\mathbf{Z}| - j) \\
&\cdot P_{j+u}^n (1 - p_D) \left[\int D(\mathbf{x}) d\mathbf{x} \right]^{-(j+u)} \\
&\cdot e_j(\Xi_k(D, \mathbf{Z})).
\end{aligned} \tag{21}$$

The term $\rho_{c, k}(n)$ in (21) is the distribution of the numbers of clutters at time k . The other terms are given as follows:

$$\begin{aligned}
\Xi_k(D, \mathbf{Z}) &= \left\{ \frac{p_D \int D(\mathbf{x}) g_k(\mathbf{z} | \mathbf{x}) d\mathbf{x}}{c(\mathbf{z})} : \mathbf{z} \in \mathbf{Z} \right\}, \\
e_j(\mathbf{Z}) &= \sum_{\mathbf{W} \subseteq \mathbf{Z}, |\mathbf{W}|=j} \left(\prod_{\xi \in \mathbf{W}} \xi \right), \\
P_\ell^n &= \frac{n!}{(n - \ell)!},
\end{aligned} \tag{22}$$

where $e_j(\mathbf{Z})$ is the elementary symmetric function of order j for the random finite set \mathbf{Z} ; the details can be found in [37].

Let \hat{n} be the integer nearest in value to $\rho_{k|k}(n)$. Then the \hat{n} highest peaks of $D_{k|k}(\mathbf{y}, 0)$ are corresponding to the states of the agents. That is, let $\hat{\mathbf{y}}_1, \dots, \hat{\mathbf{y}}_{\hat{n}}$ be such that $D_{k|k}(\hat{\mathbf{y}}_1, 0), \dots, D_{k|k}(\hat{\mathbf{y}}_{\hat{n}}, 0)$ are the \hat{n} largest values of $D_{k|k}(\cdot, 0)$. Then $\hat{\mathbf{Y}}_{k|k} = \{\hat{\mathbf{y}}_1, \dots, \hat{\mathbf{y}}_{\hat{n}}\}$ is an estimate of the multiple agent states. The assimilated agent number is $\hat{n}_{k|k} = \arg \sup \rho_{k|k}(n | \mathbf{Z}_k)$ by using MAP estimation.

6. Experiments

6.1. Experimental Design. We take advantage of the identical-twin experiment introduced in [31, 32] to verify the proposed data assimilation algorithm in ideal situations. In the identical-twin experiment, the simulation whose results are considered as real states is first executed with corresponding measurements which are collected during the process. And the measurements here are treated as the real measurements, since they are collected from the real states. Accordingly, we assimilate the measurements into DDDS by using the proposed data assimilation algorithm and then check whether the assimilated results are close to the real system states or not.

We utilize three terms in the experiments, real pirate movement, assimilated pirate movement, and simulated pirate movement, to represent the results of the experiments. A real pirate movement is the simulated pirate movement from which the online measurements are obtained. A simulated pirate movement is the simulation states based on some erroneous parameters, for example, imprecise turn rate, which aims to comply with the fact that pirate movement simulation usually relies on imprecise parameters compared to real pirate movement. Finally, an assimilated pirate movement is the data assimilation enhanced simulation states based on the same erroneous parameters as in the simulated pirate movement. The goal of these experiments is to verify that an assimilated pirate movement creates more accurate simulation states by assimilating online measurements from the real pirate movement, even if it uses the erroneous parameters in the simulated pirate movement.

6.2. Experiments on Single Pirate Vessel in Clutter. The state space for the pirate vessel is four-dimensional with two position dimensions in the normal (\mathbf{x}, \mathbf{y}) Cartesian coordinate system and two velocity coordinates, $(\dot{\mathbf{x}}, \dot{\mathbf{y}})$. We establish the 2D horizontal turn motion model relying on ship kinematics. We use a widely used ship motion model [44]. The dynamics equation is as follows:

$$\begin{aligned}
\mathbf{x}_k &= \mathbf{x}_{k-1} + T v_{k-1} \sin \phi_{k-1}, \\
\mathbf{y}_k &= \mathbf{y}_{k-1} + T v_{k-1} \cos \phi_{k-1}, \\
\phi_k &= \phi_{k-1} \\
&\quad + T v_{k-1} \left[\Omega_{k-1} + \frac{1}{2} (\Omega_{k-1} - \Omega_0) T v_{k-1} \tau e^{T v_{k-1} \tau} \right],
\end{aligned}$$

$$\begin{aligned}\Omega_k &= \Omega_{k-1} e^{T v_{k-1} \tau} + \Omega_0 (1 - e^{T v_{k-1} \tau}), \\ v_{k-1} &= K \cdot v_0 = v_0 (1 + 1.9 \Omega_{k-1}^2 L^2)^{-1};\end{aligned}\quad (23)$$

here, $\tau = (-p/2 + \sqrt{p^2/4 - q})/L$, $\Omega_0 = \Omega/v_0$, and (\mathbf{x}, \mathbf{y}) , Ω , and ϕ are ship position, velocity vector turn rate, and heading, respectively. T is the sampling interval.

The simulation model $\mathbf{x}_k = \varphi_k(\mathbf{x}_{k-1}, v_{k-1})$ of the vessel motion is nonlinear, where φ_k is the nonlinear state transition functions and v_{k-1} is an independent zero-mean Gaussian process noise; and Q_{k-1} is the covariance matrix. The vessel simulation model can be reformulated as

$$\begin{aligned}\mathbf{x}_k &= \varphi_{k-1}(\mathbf{x}_{k-1}) + G \cdot \dot{\phi}_{k-1}, \\ \dot{\phi}_k &= \dot{\phi}_{k-1} + T \cdot u_{k-1},\end{aligned}\quad (24)$$

where $G = \begin{bmatrix} T^2/2 & T & 0 & 0 \\ 0 & 0 & T^2/2 & T \end{bmatrix}^T$, $\dot{\phi}_{k-1} \sim N(\cdot; 0, \sigma_{\dot{\phi}}^2 I)$, $u_{k-1} \sim N(\cdot; 0, \sigma_u^2 I)$, $T = 1$ s, $\sigma_{\dot{\phi}} = 15$ m/s², and $\sigma_u = \pi/180$ rad/s. The probability of survival is $p_{S,k}(\mathbf{x}) = 0.99$. The probability of detection is $p_{D,k}(\mathbf{x}) = 0.98$. The speed of the pirate vessel is 25–30 knots.

The sensor measurement model in the experiments is as follows:

$$\mathbf{z}_k = \begin{bmatrix} \arctan\left(\frac{\mathbf{x}_k}{\mathbf{y}_k}\right) \\ \sqrt{\mathbf{x}_k^2 + \mathbf{y}_k^2} \end{bmatrix} + \boldsymbol{\varepsilon}_k, \quad (25)$$

where measurement noise $\boldsymbol{\varepsilon}_k \sim N(\cdot; 0, \mathbf{R}_k)$ with $\mathbf{R}_k = \text{diag}([\sigma_{\theta}^2, \sigma_r^2]^T)$; in the simulation experiments, we assume that $\sigma_{\theta} = 2(\pi/180)$ rad, and $\sigma_r = 10$ m.

The false detection obeys the Poisson RFS with PHD $\lambda_c = 3.2 \times 10^{-3}$ (rad·m)⁻¹ over the disc $[0, 2\pi]$ rad \times $[0, 2000]$ m.

In our experiments, the differences between a real pirate movement and a simulated one are caused by the imprecise turn rate $\dot{\phi}$. We use the imprecise turn rate as the erroneous parameters. The real turn rate is $\pi/60 \approx 0.0524$ rad/s with random variances added every second. Our first experiment case uses the real turn rate to run the agent based simulation system and collects the measurements at every time step. Our second experiment case introduces errors to the turn rate and makes the other configurations exactly the same as the first experiment. The erroneous turn rate is $0.0524 + 0.01/3 \approx 0.05573$ rad/s with the same random variances in the first experiment. Our third experiment case uses the erroneous turn rate as the second experiment and the other configurations remain the same as the other two experiments. In this experiment case, the measurements collected in the first experiment are used as well as the proposed data assimilation algorithm.

We assume that the merchant vessel is stationary at a particular location, which is the center in the Cartesian coordinates. In this way, we can get the pirate's states. The states of the pirate vessel generated by the three experiments are shown in Figure 9.

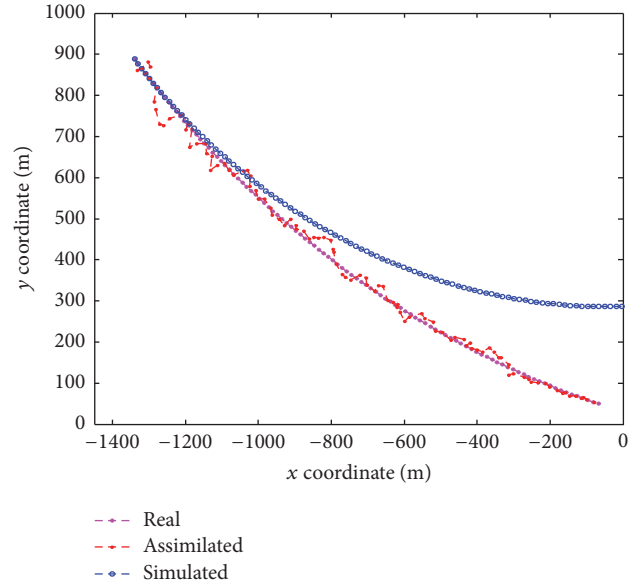


FIGURE 9: States of identical-twin experiments on single pirate vessel in clutter.

This set of experiments verifies the effectiveness of data assimilation algorithm when the turn rate has errors. From Figure 9, we know that the real pirate vessel movement and the simulated one have large deviations due to the turn rate errors. By assimilating the measurements, the assimilated pirate vessel movement is obtained. From this figure, it is observed that the assimilated pirate vessel movement is much closer to the real one than the simulated one. Specifically, due to the erroneous turn rate, the simulated pirate states gradually deviate from the real pirate states. Using the proposed data assimilation algorithm, the assimilated pirate states overcome this problem and match the real pirate states with minor differences.

Figure 10(a) shows the PHD of positions and Figure 10(b) shows the contours of the marginal PHD in all time steps. We can learn that the PHD is gradually improved as the simulation time evolves. We can find that, in the initial phase, the uncertainty of the PHD is large due to the limited and imprecise prior information, and as the simulation time evolves, the uncertainty gradually decreases. This phenomenon is caused by the fact that the knowledge about the PHD (related to the posterior probability density) is gradually accumulated over time, and the uncertainty about the mean of mixture Gaussian distribution of PHD gradually decreases at the same time. This result proves that the proposed data assimilation algorithm can make the uncertainty of the estimated results be controlled in a limited range and it can overcome the uncertainty of the measurements including clutters, false alarms, missed detections, and noises.

To capture the average performance, we run 1000 Monte Carlo trials for each experiment with the same setup. Figure 11 shows the Monte Carlo average of the mean squared error at every step for the assimilated pirate states and the simulated ones. We can find that the mean squared errors between the real pirate states and the assimilated ones are smaller

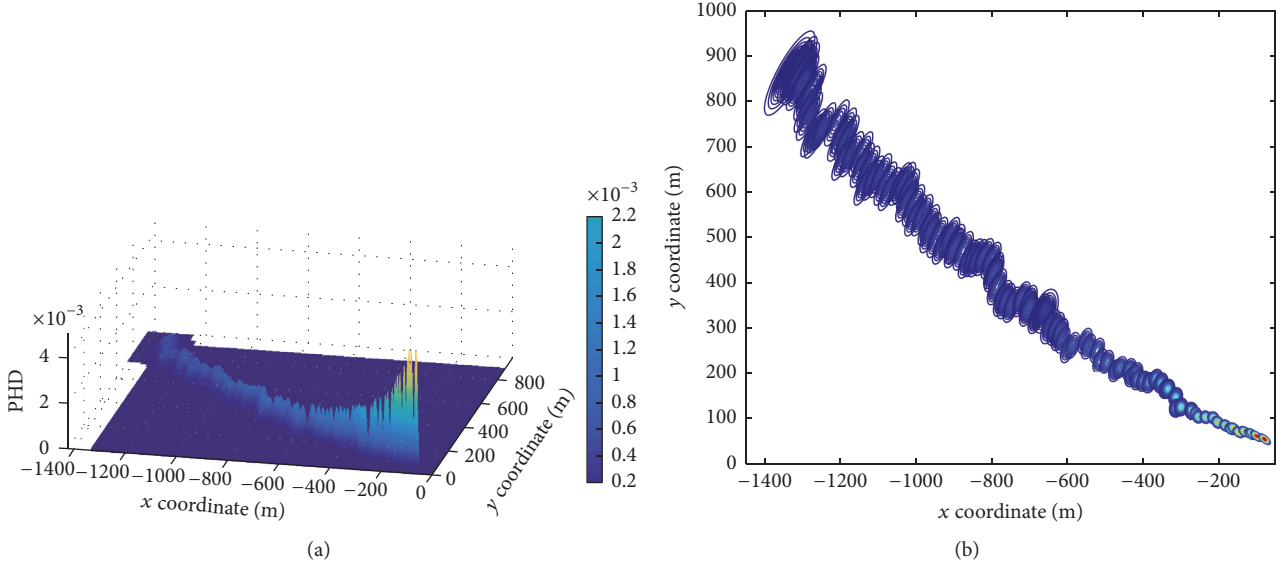


FIGURE 10: PHD of single pirate agent versus time: (a) surface plot of the distribution of PHD; (b) contours of the marginal PHD.

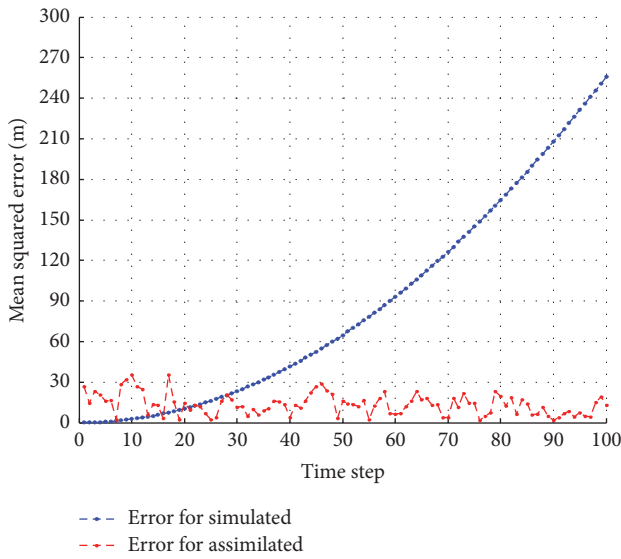


FIGURE 11: 1000 MC run average mean squared errors versus time for assimilated and simulated results.

than those of the simulated ones. As time evolves, the mean squared errors between the real pirate states and the simulated ones gradually increase, while the mean squared errors between the real ones and the assimilated ones are stable and limited. This figure reflects the differences between using data assimilation and not using it.

At each time step, the data assimilation algorithm assimilated the measurements by using the RFS based Gaussian mixture. To see the data assimilation process more clearly, we give the assimilation result at a single time step. Here we analyze the result at time step 57. To see the distribution of the PHD more intuitively, we give the surface plot of the distribution of PHD. From Figure 12, we can know that the assimilated states of the pirate vessel are the states having the

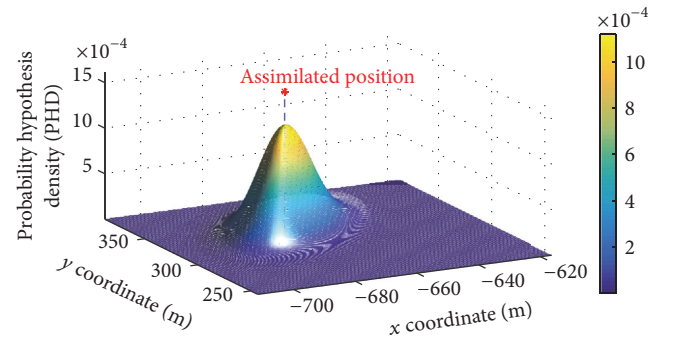


FIGURE 12: Surface plot of the distribution of assimilated PHD at time step 57.

largest PHD. Since there is only one pirate vessel in the experiment, there is one two-dimensional Gaussian distribution in the xy plane.

Figure 13 shows the real pirate vessel state with cluttered measurements and the simulated one and the assimilated one at time step 57. From this figure, we can find that the assimilated state is much closer to the real one than the simulated one even though the measurements are heavily influenced by the clutters and noises. We also plot the mean value of the measurements. We can find that neither the mean value nor the measurements are closer to the real state than the assimilated one. This proves that the direct using of measurements without data assimilation is not applicable.

6.3. Experiments on Multiple Pirate Vessels. Now consider a nonlinear bearing and range scenario with a total of 5 pirate vessels and a mothership. The birth process follows a Poisson RFS with PHD:

$$\gamma_k(\mathbf{x}) = \sum_{i=1}^4 w_\gamma^{(i)} N(\mathbf{x}; \mathbf{m}_\gamma^{(i)}, \mathbf{P}_\gamma), \quad (26)$$

TABLE I: Configurations of different pirate vessels.

Pirate vessels	Real turn rate (rad/s)	Biased turn rate (rad/s)	Start time (s)	Stop time (s)
Pirate 1	0.0058	0.0092	1	100
Pirate 2	0.0058	0.0092	1	80
Pirate 3	0.0087	0.0137	1	90
Pirate 4	0.0087	0.0137	30	100
Pirate 5	0.0058	0.0092	35	100
Mothership	0.0044	0.0069	1	50

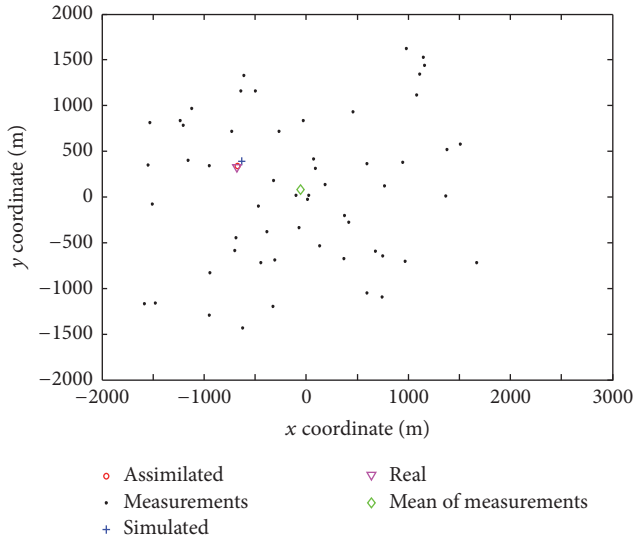


FIGURE 13: The real state, assimilated state, simulated state, and corresponding measurements at time step 57.

where $w_\gamma^{(1)} = w_\gamma^{(2)} = 0.03$ and $w_\gamma^{(3)} = w_\gamma^{(4)} = 0.02$, $\mathbf{m}_\gamma^{(1)} = [-1000, 0, 1500, 0; 0]^T$, $\mathbf{m}_\gamma^{(2)} = [-1000, 0, -1000, 0; 0]^T$, $\mathbf{m}_\gamma^{(3)} = [1000, 0, -1500, 0; 0]^T$, $\mathbf{m}_\gamma^{(4)} = [1000, 0, 500, 0; 0]^T$, and $\mathbf{P}_\gamma = \text{diag}([50, 50, 50, 50; 6(\pi/180)]^T)^2$.

Like in Section 6.2, we carry out three experiments with the same measurement model and simulation model. The configurations of the three experiments are identical except the turn rate. The real turn rates and erroneous ones of all 5 pirate agents and a mother ship are shown in Table 1. To show the ability of assimilating varying agents' number of the proposed data assimilation algorithm, we make the pirate vessel agents' number vary by setting different start and stop time.

We assume that the merchant vessel is stationary at a particular location as a center. Thus, we can get the states in the Cartesian coordinates with the location of merchant vessels. Pirate vessels' states are shown in Figure 14.

From Figure 14, it is not difficult to find that, by assimilating the measurements, the assimilated pirate vessel states are closer to the real ones than the simulated ones.

Figure 15(a) shows the PHD of positions and Figure 15(b) shows the contours of the marginal PHD in all time steps. It can be learned that the PHD of all agents is gradually improved as the simulation time evolves. We can learn that,

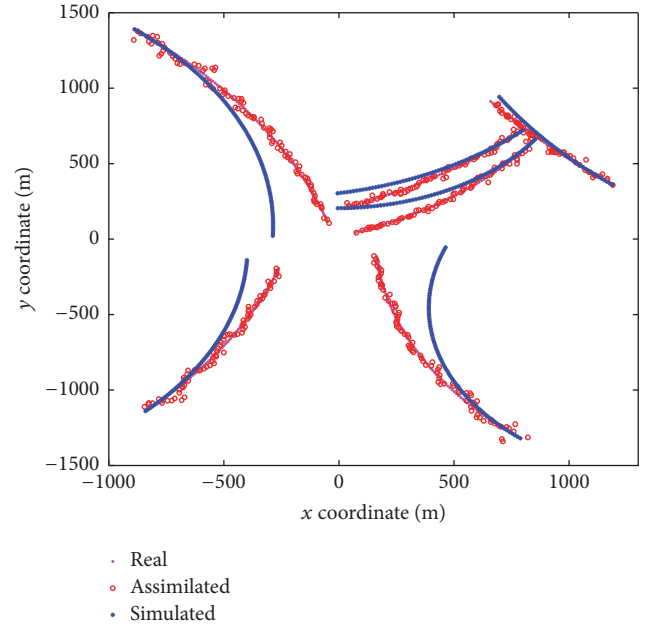


FIGURE 14: States of identical-twin experiments on multiple pirates.

in the initial phase, the uncertainty of the PHD is large due to the limited and imprecise prior information, and as the simulation time evolves, the uncertainty of PHD gradually decreases. This phenomenon is consistent with the experiment for the single pirate agent. From this result, we can know that the varying agents' number will not affect the ability of the proposed algorithm to control the uncertainty of estimated results in a limited range and overcome the uncertainty of measurements.

Unlike the analysis of single pirate vessel, the analysis of the multiple pirate vessels cannot use the distance error directly. Here we use the new metric which is proposed by Schuhmacher et al. for sets, named as the optimal subpattern assignment (OSPA) metric [45]. The OSPA metric is employed extensively as a performance evaluation criterion here.

To capture the average performance, we also run 1000 Monte Carlo trials for each experiment with the same setup. Figure 16 shows the Monte Carlo average of the OSPA distance, and we can find that the Monte Carlo average of the OSPA distance of simulated states gradually increases, while the OSPA distance of assimilated states is limited to a smaller range. This result reflects the fact that the multiple agent simulation using data assimilation is far more accurate than those not using it.

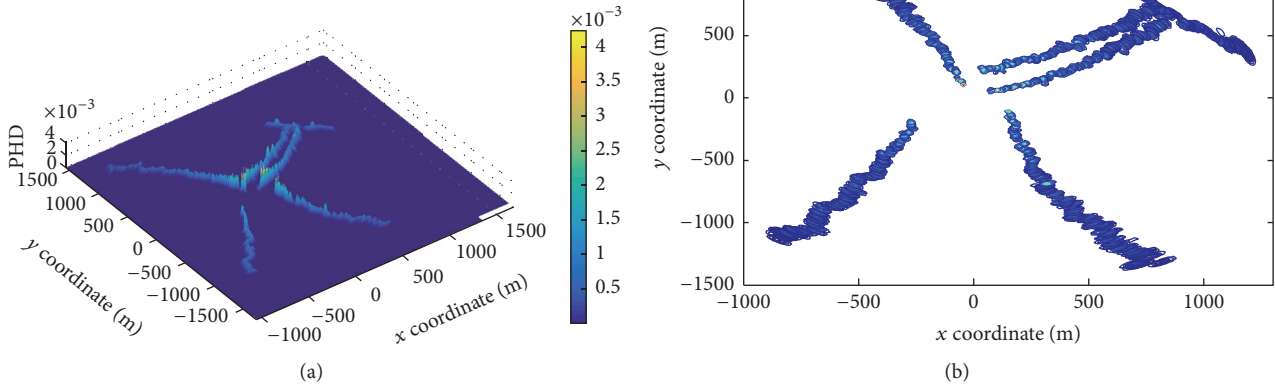


FIGURE 15: PHD of multiple pirate agents versus time: (a) surface plot of the distribution of PHD; (b) contours of the marginal PHD.

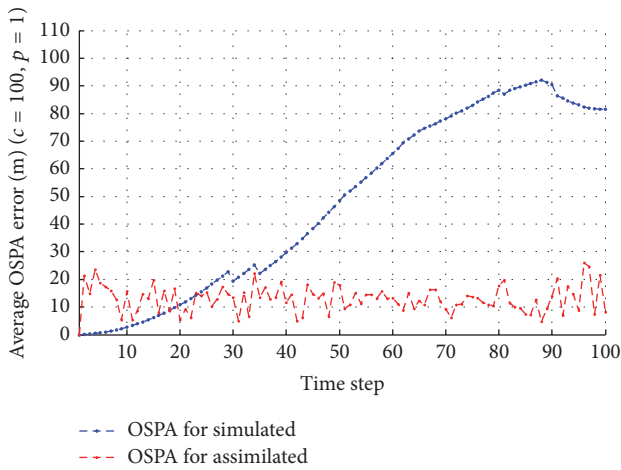


FIGURE 16: 1000 MC run average OSPA error versus time for simulated states and assimilated states.

Figure 17 shows the agents' number in the real experiment and the assimilated agents' number in the assimilated experiment. As the assimilated agents' number is estimated by rounding off the mean of updated agents' cardinality, a few estimation errors are caused. But mostly the assimilated numbers are consistent with the real agents' numbers. This figure proves that the proposed data assimilation algorithm not only can assimilate the simulation states of the agents but also can estimate the varying agents' number in the simulation. It is difficult for the standard PF based data assimilation algorithms to estimate the agents' number at every time step.

To describe the working mechanism of the proposed data assimilation algorithm more clearly, we give the assimilated results at a single time step. Here, we analyze time step 48 at which all the six pirate vessels are concurrent. To see the distribution of the PHD more intuitively, we give the surface plot of the distribution of PHD. From Figure 18, we can know that the assimilated states of the pirate vessels are those with the largest PHD, and the number of the two-dimensional Gaussian distributions equals the number of pirate vessels.

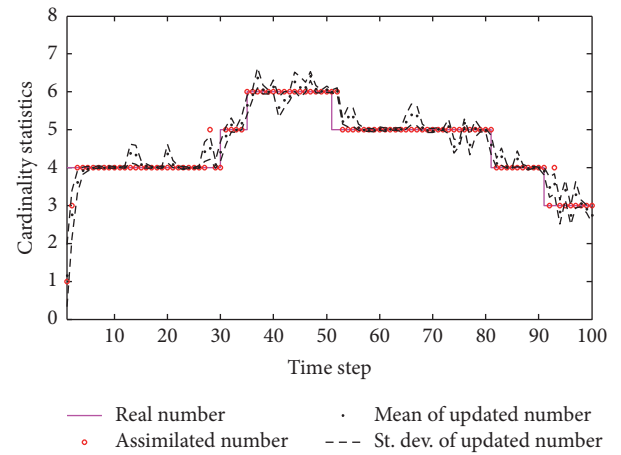


FIGURE 17: True versus assimilated cardinality over time.

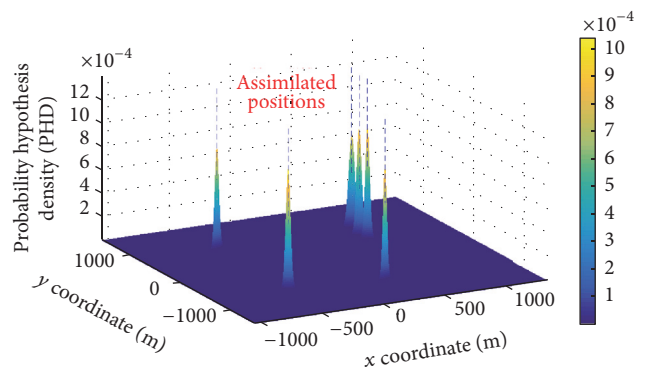


FIGURE 18: Surface plot of the distribution of resulting PHD at time step 48.

Figure 19 shows the real pirate vessels' states, the simulated ones, the assimilated ones, the clutters, and noises. We can see that the assimilated pirate vessels' states are closer to the real ones than the simulated ones. It can be seen that the assimilated states are not affected by the disorganized distribution of measurements.

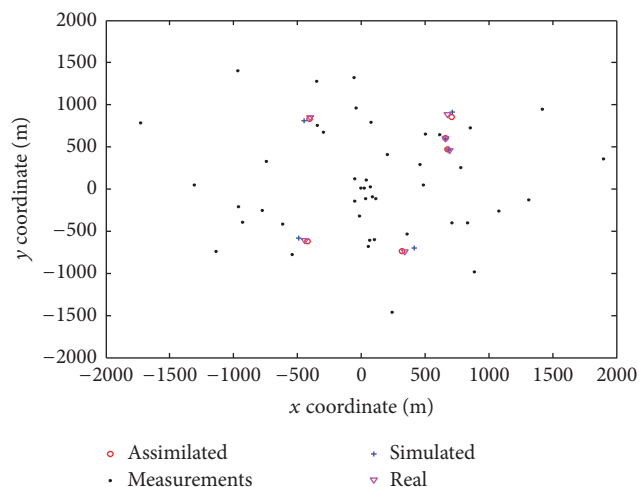


FIGURE 19: The real states, assimilated states, simulated states, and corresponding measurements at time step 48.

7. Conclusions

In this paper, a RFS based data assimilation algorithm is proposed for the agent based DDDS of maritime pirate activity. The main limitations of the traditional PF based and Kalman filter based data assimilation algorithms have been analyzed; therefore by employing RFS these limitations can be effectively overcome. This algorithm bears the ability to meet the demands of practical applications. Additionally, we designed the RFS based measurement model and simulation model for the maritime sensors and vessel agents. We implemented the RFS based data assimilation algorithm by using RFS based Gaussian mixture. We tested the effectiveness of the proposed algorithm by using the identical-twin experiments of maritime pirate activity. The experiment results prove that the proposed data assimilation algorithm can assimilate the input measurements and improve the accuracy of simulation results to a certain extent.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

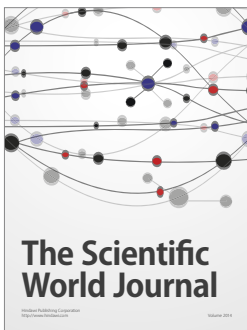
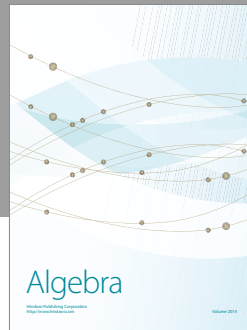
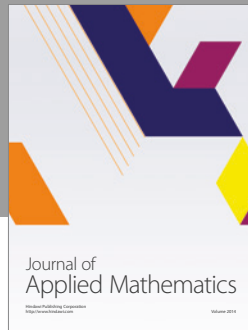
Acknowledgments

The current work has been supported by the National Natural Science Foundation of China (Grant 61374185). The authors have obtained the patient guidance and great assistance from Professor Kedi Huang.

References

- [1] D. Earnest and S. Yetiv, "Economic globalization and notional insecurity: vulnerabilities in the global intermodal shipping network," in *Proceedings of the 50th Annual Convention "Exploring the Past, Anticipating the Future"*, pp. 1–28, 2009.
- [2] A. Shortland and M. Vothknecht, "Combating "maritime terrorism" off the coast of Somalia," *European Journal of Political Economy*, vol. 27, no. 1, pp. S133–S151, 2011.
- [3] U. Daxecker and B. Prins, "Insurgents of the Sea: Institutional and Economic Opportunities for Maritime Piracy," *Journal of Conflict Resolution*, vol. 57, no. 6, pp. 940–965, 2013.
- [4] J. V. Hastings, "Geographies of state failure and sophistication in maritime piracy hijackings," *Political Geography*, vol. 28, no. 4, pp. 213–223, 2009.
- [5] E. Marchione and S. D. Johnson, "Spatial, temporal and spatio-temporal patterns of maritime piracy," *Journal of Research in Crime and Delinquency*, vol. 50, no. 4, pp. 504–524, 2013.
- [6] R. A. Waraich, M. D. Galus, C. Dobler, M. Balmer, G. Andersson, and K. W. Axhausen, "Plug-in hybrid electric vehicles and smart grids: investigations based on a microsimulation," *Transportation Research Part C: Emerging Technologies*, vol. 28, pp. 74–86, 2013.
- [7] J. J. Tang, S. Alam, C. Lokan, and H. A. Abbass, "A multi-objective approach for Dynamic Airspace Sectorization using agent based and geometric models," *Transportation Research Part C: Emerging Technologies*, vol. 21, no. 1, pp. 89–121, 2012.
- [8] S. Bourdon, Y. Gauthier, and J. Greiss, "MATRICS: a maritime traffic simulation," Tech. Rep., Defence RD Canada, 2007.
- [9] I. Norstad, K. Fagerholt, and G. Laporte, "Tramp ship routing and scheduling with speed optimization," *Transportation Research Part C: Emerging Technologies*, vol. 19, no. 5, pp. 853–865, 2011.
- [10] A. Bruzzone, M. Massei, F. Madeo, F. Tarone, and M. Gunal, "Simulating maritime asymmetric scenarios for testing different C2 maturity levels," in *Proceedings of the in*, pp. 12–23, 2011.
- [11] L. Sloomaker, *Countering Piracy with The Next-Generation Piracy Performance Surface Model [Master, thesis]*, California, Calif, USA, 2011.
- [12] J. Hansen, L. Hsu, J. Dykes et al., "Information domination: dynamically coupling METOC and INTEL for improved guidance for piracy interdiction," in *NRL Review, Naval Research laboratory*, pp. 109–115, 2011.
- [13] T. Tsilis, *Counter-Piracy Escort Operations in the Gulf of Aden [Master, thesis]*, California, Calif, USA, 2011.
- [14] J. Decraene, M. Anderson, and M. Y. H. Low, "Maritime counter-piracy study using agent-based simulations," in *Proceedings of the 2010 Spring Simulation Multiconference, SpringSim'10*, April 2010.
- [15] M. Jakob, O. Vaněk, and M. Pěchouček, "Using agents to improve international maritime transport security," *IEEE Intelligent Systems*, vol. 26, no. 1, pp. 90–95, 2011.
- [16] O. Vaněk, M. Jakob, O. Hrstka, and M. Pěchouček, "Agent-based model of maritime traffic in piracy-affected waters," *Transportation Research Part C: Emerging Technologies*, vol. 36, pp. 157–176, 2013.
- [17] F. Darema, "DDDAS, A key driver for large-scale-big-data and large-scale-big-computing," *Procedia Computer Science*, vol. 51, p. 2463, 2015.
- [18] Y. Li, Z. Liu, H. Zhu, J. Song, and Y. Liang, "Real-time flood simulations using CA model driven by dynamic observation data," *International Journal of Geographical Information Science*, vol. 29, pp. 1–13, 2014.
- [19] X. Hu, "Dynamic data driven simulation," *SCS MS Magazine II*, vol. 1, pp. 16–22, 2011.
- [20] F. Bouttier and P. Courtier, "Data assimilation concepts and methods," *Training course notes of ECMWF*, 1999.
- [21] A. C. Lorenc, S. P. Ballard, R. S. Bell et al., "The Met. Office global three-dimensional variational data assimilation scheme," *Quarterly Journal of the Royal Meteorological Society*, vol. 126, no. 570, pp. 2991–3012, 2000.

- [22] E. Bélanger and A. Vincent, "Data assimilation (4D-VAR) to forecast flood in shallow-waters with sediment erosion," *Journal of Hydrology*, vol. 300, no. 1-4, pp. 114–125, 2005.
- [23] R. H. Reichle, J. P. Walker, R. D. Koster, and P. R. Houser, "Extended versus ensemble Kalman filtering for land data assimilation," *Journal of Hydrometeorology*, vol. 3, no. 6, pp. 728–740, 2002.
- [24] G. Evensen, "The Ensemble Kalman Filter: theoretical formulation and practical implementation," *Ocean Dynamics*, vol. 53, no. 4, pp. 343–367, 2003.
- [25] H. Moradkhani, K.-L. Hsu, H. Gupta, and S. Sorooshian, "Uncertainty assessment of hydrologic model states and parameters: sequential data assimilation using the particle filter," *Water Resources Research*, vol. 41, no. 5, pp. 1–17, 2005.
- [26] N. G. Plant and K. T. Holland, "Prediction and assimilation of surf-zone processes using a Bayesian network. Part I: Forward models," *Coastal Engineering Journal*, vol. 58, no. 1, pp. 119–130, 2011.
- [27] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Journal of Fluids Engineering*, vol. 82, no. 1, pp. 35–45, 1960.
- [28] G. Evensen, *Data assimilation: the ensemble kalman filter*, Springer, Berlin, Germany, 2nd edition, 2009.
- [29] X. Xie, A. Verbraeck, and F. Gu, "Data assimilation in discrete event simulations—a rollback based sequential Monte Carlo approach," in *Proceedings of the 2016 TMS/DEVS Symposium on Theory of Modeling and Simulation, TMS/DEVS 2016, Part of the 2016 Spring Simulation Multiconference, SpringSim 2016*, April 2016.
- [30] P. Wu, H. Xue, and X. Hu, "Particle filter based traffic data assimilation with sensor informed proposal distribution," in *Proceedings of the Spring Simulation Conference*, Alexandria, VA, USA, 2015.
- [31] H. Xue, F. Gu, and X. Hu, "Data assimilation using sequential Monte Carlo methods in wildfire spread simulation," *ACM Transactions on Modeling and Computer Simulation (TOMACS)*, vol. 22, no. 4, article no. 23, 2012.
- [32] H. Xue and X. Hu, "An effective proposal distribution for sequential Monte Carlo methods-based wildfire data assimilation," in *Proceedings of the 2013 43rd Winter Simulation Conference - Simulation: Making Decisions in a Complex World, WSC 2013*, pp. 1938–1949, Washington, DC, USA, December 2013.
- [33] I. R. Goodman, R. P. Mahler, and H. T. Nguyen, *Mathematics of data fusion*, vol. 37 of *Theory and Decision Library. Series B: Mathematical and Statistical Methods*, Kluwer Academic Publishers Group, Dordrecht, Netherlands, 1997.
- [34] R. P. S. Mahler, "Multitarget bayes filtering via first-order multitarget moments," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 39, no. 4, pp. 1152–1178, 2003.
- [35] J. Goutsias, R. Mahler, and H. Nguyen, *Random Sets Theory and Applications*, Springer, New York, NY, USA, 1997.
- [36] P. Ranald and S. Mahler, "Global integrated data fusion," in *Proceedings of the 7th Nat. Symp. on Sensor Fusion*, vol. 1, pp. 187–199, Global integrated data fusion, 1994.
- [37] P. S. Ranald, *Statistical Multisource-Multitarget Information Fusion*, Artech House, Norwood, MA, USA, 2007.
- [38] D. Stoyan, W. S. Kendall, and J. Mecke, *Stochastic geometry and its applications*, Wiley Series in Probability and Mathematical Statistics: Applied Probability and Statistics, John Wiley & Sons, Ltd., Chichester, New York, NY, USA, 1995.
- [39] D. J. Daley and D. Vere-Jones, *An introduction to the theory of point processes*, Springer Series in Statistics, Springer, New York, NY, USA, 1988.
- [40] S. J. Julier and J. K. Uhlmann, "Unscented filtering and nonlinear estimation," *Proceedings of the IEEE*, vol. 92, no. 3, pp. 401–422, 2004.
- [41] B.-N. Vo, S. Singh, and A. Doucet, "Sequential Monte Carlo methods for multi-target filtering with random finite sets," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 41, no. 4, pp. 1224–1245, 2005.
- [42] R. P. S. Mahler, "PHD filters of higher order in target number," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 43, no. 4, pp. 1523–1543, 2007.
- [43] S. Julier, J. Uhlmann, and H. F. Durrant-Whyte, "A new method for the nonlinear transformation of means and covariances in filters and estimators," *Institute of Electrical and Electronics Engineers Transactions on Automatic Control*, vol. 45, no. 3, pp. 477–482, 2000.
- [44] R. Pershitz, *Ship's maneuverability and control*. Leningrad, USSR: Sudostroenie, 1973.
- [45] D. Schuhmacher, B.-T. Vo, and B.-N. Vo, "A consistent metric for performance evaluation of multi-object filters," *IEEE Transactions on Signal Processing*, vol. 56, no. 8, part 1, pp. 3447–3457, 2008.



Hindawi

Submit your manuscripts at
<https://www.hindawi.com>

