

Communication

# Cooperative-PHD Tracking Based on Distributed Sensors for Naval Surveillance Area

Kleberon Meireles de Lima \*  and Ramon Romankevicius Costa 

PEE/COPPE—Department of Electrical Engineering, Federal University of Rio de Janeiro, Cidade Universitária, Centro de Tecnologia, Bloco H, Rio de Janeiro 21941-972, RJ, Brazil; ramon@coep.ufrj.br

\* Correspondence: kleberonmlima@outlook.com

**Abstract:** Brazil has an extensive coastline and Exclusive Economic Zone (EEZ) area, which are of high economic and strategic importance. A Maritime Surveillance System becomes necessary to provide information and data to proper authorities, and target tracking is crucial. This paper focuses on a multitarget tracking application to a large-scale maritime surveillance system. The system is composed of a sensor network distributed over an area of interest. Due to the limited computational capabilities of nodes, the sensors send their tracking data to a central station, which is responsible for gathering and processing information obtained by the distributed components. The local Multitarget Tracking (MTT) algorithm employs a random finite set approach, which adopts a Gaussian mixture Probability Hypothesis Density (PHD) filter. The proposed data fusion scheme, utilized in the central station, consists of an additional step of prune & merge to the original GM PHD filter algorithm, which is the main contribution of this work. Through simulations, this study illustrates the performance of the proposed algorithm with a network composed of two stationary sensors providing the data. This solution yields a better tracking performance when compared to individual trackers, which is attested by the Optimal Subpattern Assignment (OSPA) metric and its location and cardinality components. The presented results illustrate the overall performance improvement attained by the proposed solution. Moreover, they also stress the need to resort to a distributed sensor network to tackle real problems related to extended targets.



**Citation:** Lima, K.M.d.; Costa, R.R. Cooperative-PHD Tracking Based on Distributed Sensors for Naval Surveillance Area. *Sensors* **2022**, *22*, 729. <https://doi.org/10.3390/s22030729>

Academic Editors: Andrzej Stateczny

Received: 13 November 2021

Accepted: 22 December 2021

Published: 19 January 2022

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

**Keywords:** surveillance systems; tracking; PHD filter

## 1. Introduction

In 2015, the Brazilian Navy presented the strategic program, so-called the Management System of Amazônia Azul (SisGAAz). Its main objective is to monitor the surveillance of Brazilian jurisdictional waters and the international areas of responsibility for Search and Rescue (SAR) operations [1].

The SCUA (freely translated to the Unified Situational Awareness System) implements the SisGAAz's first step in the gradual development of this Brazilian program. This system is a C4ISR (Command, Control, Communications, Computers, Intelligence, Surveillance, and Reconnaissance) that integrates multisensor and multisource data to either evaluate situational awareness or support decision-making activities for the maritime area of interest. In the context of situational awareness, multitarget tracking is a crucial task.

As for the problem of tracking multiple targets with an unknown number of targets, random finite sets have attracted significant attention since they were introduced in the 1990s by Mahler [2]. Since this new approach was introduced, several filters for its implementation have been proposed. In the 2000s, real-data implementations were emerging in areas as diverse as underwater active acoustics and air-to-ground GMTI (Ground Moving Target Indicator) detection and tracking [3].

Although the Probability Hypothesis Density (PHD) filter is already widely known, real-time implementation in systems that demand high performance and availability remains an issue. Over the years, several works have proposed ways to overcome these

concerns. According to Gao et al. [4], although there is no data association problem in this filter, it can hardly be real-time implemented on a serial processor.

Different approaches have tried to overcome processing limitations or speed up processing. Zheng et al. [5] have proposed an efficient event-driven data particle PHD filter for real-time multi-target tracking. In order to reduce processing, the proposed method distinguishes the survival measurements, spontaneous birth measurements, and clutters for weight computation, which are steps with high computational complexity. With the same objective, Li et al. [6] present an algorithmic framework for parallel SMC-PHD filtering based on multiple processing. This algorithm fully parallels all Sequential Monte Carlo (SMC) steps.

Few papers address the parallelization and optimization of the implementation of SMC PHD filters. Li et al. [7] has proposed a novel Cardinality Consensus (CC) scheme to PHD filter for multitarget tracking in decentralized sensor networks with severely constrained communication. This scheme has a parallel filtering-communication mode as it performs communication parallel to filtering operations, which leads to data exchanges that require only a tiny amount of time. Li et al. [8] present details of GM PHD on the CC scheme, which is the first distributed GM filter based on the parallel filtering-communication mode, according to the authors.

In distributed multitargeting tracking applications, the fusion process of data received from sensors is another concern. Li et al. [9] propose a suitable real-time data fusion algorithm called Generalized Covariance Intersection (GCI). They, see [10], present another solution for this issue. This work introduces Fixed-nodes PHD (FN-PHD) fusion, which can achieve similar performance as PHD, but at much lower communication and computational costs. Firstly, radars track targets by a PHD filter. Then the target state closest to the fusion node is chosen, and all states among radars based on GCI are predicted and fused.

The SCUA applies parallelism to solve complex problems concerning data fusion in real time. The Brazilian Navy system has already implemented the PPTS (Pair of Plots in Two Stages) algorithm. This approach is a graph-based method proposed in [11], which is PP (Pair of Plot [12]) based. In 2015, Mello et al. proposed the implementation based on the model Gamma [13]. The proposed solution was the first parallel implementation of the PPTS method, which employed three Gamma models, where two of them exploited the resources of a parallel hardware environment (using MPI protocol and GPU).

This manuscript deals with the application of the GM PHD algorithm in a distributed sensor network, which can be: Radars or cameras installed along the coast or area of interest; radars on patrol or larger ships; and sensors installed in smaller boats. In such a system, unlike those presented in other applications [14–16], each node has its processing and tracking solution. Furthermore, through a communication link, they send their tracking data to a central station responsible for gathering and processing the information obtained by the distributed components.

The main contribution of this work is to propose an additional step of ‘prune & merge’ to the original GM PHD filter algorithm. This approach includes an additional step expecting to overcome the limitations of sensors and the original PHD algorithm. The proposed scheme presented good results attested by the OSPA metric in the simulations carried out. Furthermore, the target occlusion is overcome, which is one of the main weaknesses of the GM PHD algorithm.

The paper is organized as follows: Section 2 presents related works; Section 3 defines the MTT problem, presents the mathematical framework for RFS-based MTT, and presents the PHD filter definition; Section 4 illustrates a distributed network tracking system application, describes methods, presents the simulated scenarios and modeling, and analyzes the results; and Section 5 exposes the conclusions.

## 2. Related Works

Many algorithms and methods for tracking targets are available in the literature. Recursive filtering techniques under a Bayesian estimation framework are commonly used to solve the tracking problem using a stochastic approach. These algorithms and methods

include extensions of the well-known Kalman Filter (KF): The Extended Kalman filter (EKF), Interactive Multiple-Model (IMM) filtering, and Variable-Structure IMM (VS-IMM) filtering, besides others filtering techniques, that include grid-based approaches, as well as nonparametric particle techniques [17].

The Kalman filter is the optimal estimator for stochastic white noise linear systems. When dealing with a nonlinear estimation problem, this technique has substantial performance limitations and no convergence guarantee. Extended and unscented KF variants circumvent the nonlinearity issues applying approximations. These algorithms are most effective when facing unimodal problems. The performance of Kalman Filter-based methods significantly decreases when applied to a multimodal problem, even in the most uncomplicated cases. It is precisely the problem considered in a maritime surveillance region: Nonlinear models and multimodal distributions.

The Multitarget Tracking (MTT) problem is multimodal. An established way to overcome parametric methods limitations is achieved with the Multi-Hypothesis Kalman Filter (MHKF) or Particle Filter (PF)-based tracking. The objective of MTT is to jointly estimate, at each observation time, the number of targets and their trajectories from sensor data. Even at a conceptual level, MTT is a nontrivial extension of single-target tracking. Indeed, MTT is far more complex in both theory and practice [18]. According to [19], if the assumptions of the Kalman filter or grid-based filters hold, then no other algorithm can out-perform them. However, the assumptions in typical real scenarios do not hold, and approximate techniques are essentially employed. As mentioned by [20], the problems found in these applications favor and justify the use of MTT algorithms. The study by Jinan and Raveendran [21] presents a particle realization applied to MTT using the coordinate turn model to characterize targets dynamics in a two-dimensional maneuver representation. The survey by Wang et al. [22] introduces some advances in algorithms for multitarget tracking.

While the Kalman Filter is concerned with tracking a unique object under a single target problem, there are complementary steps of data association and management that compose multitarget tracking. Conventional MTT techniques typically employ divide-and-conquer approaches that partition a multitarget problem into a family of parallel single-target problems [23]. Given that each KF instance corresponds to a unique target, the data association issue arises as a consequence. If a wrong association occurs (i.e., some observation is associated with a wrong track), the system cannot recover from this error. When dealing with crowded scenes, it is not straightforward to assign an observation to a particular track [20]. One strategy to deal with this problem in Multiple-Hypothesis Tracking (MHT) is to delay data association decisions by keeping multiple hypotheses active until data association ambiguities are solved [24]. Data association is a process in which each measurement is hypothesized to have been originated from a known target, a new target, or clutter. This process is the vital point of MTT, and it becomes more complex with multiple targets, where the tracks compete for measurement [25]. For an introduction to conventional algorithms of data association in MTT, see [26]. In [27], the data association problem is treated as an optimization problem, and two methods are presented, one using a neural network and the other, an evolutionary algorithm.

Although these MTT techniques are well established, they present the problem of data association as an extra and even more complex difficulty. The lack of an optimal global solution for estimating target states and the absence of a Bayesian filter without some intermediate heuristic method exemplify such complexity. Alternatively, seeking to cope with this brittleness, Mahler [2] proposes an MTT algorithm based on the theory of Random Finite Sets (RFS). The indicated approach unifies into a single probabilistic procedure that contains detection, correlation, tracking, and classification. Finite-set Statistics (FISST) is the mathematical framework that supports the RFS approach, see [23,28,29] for a more detailed review.

Notwithstanding all RFS theoretical advantages, the RFS-based recursive Bayes filter, even in single-target problems, is so computationally challenging that it requires an approximate solution to make it implementable. For this purpose, Mahler [30] presents the PHD filter, which gives the number of expected targets when integrated over a region

in target state space. Considering its recursive nature, by propagating PHD's first-order moment statistics, it becomes computationally attractive. In their study of the PHD filter, Vo and Ma [31] propose an analytical solution for the PHD recursion of targets with linear dynamics. Clark and Bell [32] analyze convergence for Sequential Monte Carlo (SMC) approximation using a particle filter, while Clark and Vo [33] consider the Gaussian Mixture (GM) realizations under linear or nonlinear stochastic processes.

There are significant applications in the literature for the mentioned methods that demonstrate their use to multitarget tracking under a high clutter environment as [34–37] for SMC, and [38–42] for GM implementations. Different works [43–45] present the Cardinalized version (CPHD), which is a generalization of the PHD recursion, propagating jointly the posterior intensity and the posterior cardinality distribution, whilst PHD recursion propagates only the intensity. Mahler [46,47] introduces more advances in PHD approximations.

Regarding naval applications, Pace [14] implements SMC and GM PHD filter for a 3D radar applied to realistic three-dimensional aerial and naval scenarios. This work illustrates and compares their performance in detecting, initiating, and terminating tracks with clutter presence. Also related to the same type of application, Do et al. [16] propose a method for online tracking multiple targets for a naval area using a Generalized Labeled Multi-Bernoulli (GLMB) filter.

There is some research on surveillance and distributed sensors based on PHD filtering. However, few papers deal with maritime surveillance and distributed processing. Laneuville and Houssineau [48] consider the problem of multitarget tracking with passive data, obtained by geographically-distributed cameras using a GM-based algorithm. Battistelli et al. [49] treat CPHD applied to distributed multitarget tracking over a sensor network of heterogeneous and geographically-dispersed nodes with sensing, communication, and processing capabilities. Pailhas et al. [15] attempt Multiple-Input Multiple-Output (MIMO) sonar systems for area surveillance, especially to a harbor environment, with a restricted area located close to a traffic area, protecting it from underwater intrusion. Gulmezoglu et al. [50] investigate the use of GM PHD filters for multiple people tracking using a network of radar sensors in an indoor environment. Dias and Bruno [51] introduce a new cooperative particle filter algorithm for tracking emitters using Received-Signal Strength (RSS) measurements considering the communication's cost. They propose two different solutions for reducing communication overhead with a modest degradation in performance. Concerning maritime surveillance, these cited applications have in common the following aspects: A central sensor fusion, stationary sensors, and no local PHD computation.

### 3. Background

This section provides minimal background to readers unfamiliar with the problem of tracking multiple targets, Bayesian filters, or the RFS approach.

#### 3.1. Problem Statement

To formulate the tracking problem under a RFS framework, it is necessary to define some fundamental issues. According to Bar-Shalom and Li [52], estimation is the process of selecting a point from a continuous space—the “best choice” in line with some criteria; tracking is the state estimation of a moving object based on remote measurements, using one or more sensors at fixed locations or on moving platforms; and filtering is the current state estimation of a dynamic system—the reason for the use of the word “filter” is that the process for obtaining the “best estimate” from noisy data amounts to “filtering out” the noise.

In general, the objective of MTT is to jointly estimate, at each observation time, the number of targets and their trajectories from sensor data. Even at a conceptual level, MTT is a non-trivial extension of single-target tracking [18]. This problem has a two-fold objective: (1) Estimate the random number of targets that are present in the area of interest and (2) estimate the random state vector of each target [49].

Different from traditional MTT vector-based techniques that divide the problem into decoupled single-target problems, RFS tracking uses a state vector collection, treating the elements as random variables as well as the number of elements in that collection itself. Mathematically, the objective is to obtain the posterior multitarget joint probability density functions:

$$p(\mathbf{X}_k, \mathbf{Z}_k), \quad (1)$$

where  $\mathbf{X}_k$  and  $\mathbf{Z}_k$  are, respectively, the state and observations random finite sets, which are defined in the next section, at the instant  $k$ .

### 3.2. RFS Fundamentals

This section presents concepts and mathematical tools necessary to contextualize RFS-based tracking and Bayesian filtering. For a more detailed review, see [23,29].

A random finite set is a convenient probabilistic model for the representation of multiple stochastic dynamic systems (objects) and sensor measurements. Suppose, at the discrete-time  $k$ , there are  $n_k$  objects with states  $\mathbf{x}_{k,1}, \dots, \mathbf{x}_{k,n_k}$  taking values in the state space  $\mathcal{X} \subseteq \mathcal{R}^{n_x}$ . Both the number of dynamic objects,  $n_k$  and their individual states in  $\mathcal{X}$  are random and time-varying. The multi-object state at instant  $k$ , represented by a finite set.

$$\mathbf{X}_k = \{\mathbf{x}_{k,1}, \dots, \mathbf{x}_{k,n_k}\} \in \mathcal{F}(\mathcal{X}), \quad (2)$$

can be modeled as a random finite set on  $\mathcal{X}$ . Here  $\mathcal{F}(\mathcal{X})$  is the set of finite subsets of  $\mathcal{X}$  [53].

Analogously, suppose that  $\mathbf{Z}_k$  is a measurement set from an observation process, which contains  $m_k$  elements reported at time  $k$ , then:

$$\mathbf{Z}_k = \{\mathbf{z}_{k,1}, \dots, \mathbf{z}_{k,m_k}\} \in \mathcal{F}(\mathcal{Z}), \quad (3)$$

is the RFS model on the observation space  $\mathcal{Z} \subseteq \mathcal{R}^{n_z}$ .

The cardinality (number of elements) and the individual state for a random finite collection are random variables that take values as unordered finite sets. These RFS characteristics provide this technique with the capacity to perform data association automatically and multitarget state estimation jointly.

The cardinality is a random variable and is modeled by a discrete distribution [54], according to:

$$\rho(n) = Pr\{|X| = n\}, \quad (4)$$

where  $n \in \mathbb{N}$ , and  $Pr$  denotes the probability.

A random finite set is a finite-set valued random variable. Thereof, it has the usual probabilistic descriptors of a random variable in the sense of Finite Set Statistics (FISST), such as the Probability Density Function (PDF) and its statistical moments. The PDF of an RFS variable  $\mathbf{X}$  is denoted  $f(\mathbf{X})$  and uniquely defined by the cardinality  $\rho(n)$  and symmetric joint distributions  $f_n(\mathbf{x}_1, \dots, \mathbf{x}_n)$ . Mathematically, the FISST PDF definition is [54]:

$$f(\mathbf{X}) = f(\{\mathbf{x}_1, \dots, \mathbf{x}_n\}) = n! \cdot \rho(n) \cdot f_n(\mathbf{x}_1, \dots, \mathbf{x}_n). \quad (5)$$

The probability distribution of the cardinality itself is obtained according to:

$$\rho(n) = \frac{1}{n!} \int f(\{\mathbf{x}_1, \dots, \mathbf{x}_n\}) d\mathbf{x}_1 \dots d\mathbf{x}_n. \quad (6)$$

A central concept in FISST is the *set integral*. Given  $f(\mathbf{X})$  is a random variables distribution over a random set, the *set integral* of  $f(\mathbf{X})$  is defined by:

$$\int f(\mathbf{X}) \delta \mathbf{X} = \sum_{n=0}^{\infty} \frac{1}{n!} \int f(\{\mathbf{x}_1, \dots, \mathbf{x}_n\}) d\mathbf{x}_1 \dots d\mathbf{x}_n. \quad (7)$$



This integral characterizes the sum over the set cardinality, integrating all possible sets according to the number of elements. Besides, the term  $1/n!$  considers the permutations of a set of size  $n$ .

The *intensity function* (also known as the probability hypothesis density or PHD) of an RFS  $\mathbf{X}$  is defined as its first statistical moment [54]:

$$D(\mathbf{x}) = \mathbb{E}\{\delta_{\mathbf{X}}(\mathbf{x})\} = \int \delta_{\mathbf{X}}(\mathbf{x})f(\mathbf{X})\delta\mathbf{X}, \quad (8)$$

where  $\delta_{\mathbf{w}}(\mathbf{x})$  is the standard Dirac delta function concentrated at  $\mathbf{w}$  and  $\delta_{\mathbf{X}}(\mathbf{x})$  is the set Dirac delta function, which is defined according to:

$$\delta_{\mathbf{X}}(\mathbf{x}) = \sum_{\mathbf{w} \in \mathbf{X}} \delta_{\mathbf{w}}(\mathbf{x}). \quad (9)$$

It is important to indicate that the PHD is not a probability density. It is uniquely characterized by the following property. Given a region  $S$  of single-target state space, the integral is defined by:

$$\hat{m} = \int_S D(\mathbf{x})d\mathbf{x}, \quad (10)$$

and is the expected number of targets in  $S$ , i.e., equal to unity. In particular, if  $S$  is the entire state space (the multitarget state space), then the integral is the total expected number of targets in the scene [23]. An intuitive interpretation of this function considers it as the density of the expected number of targets occurring at a given point. Furthermore, the mass is seen as the density's integral over the volume of space. Consequently, it is the expected number of targets. Additionally, the peaks of the intensity function indicate locations with a relatively high concentration of the expected number of targets, in other words, locations with a high probability of feature existence [55].

The Poisson RFS is a vital probability distribution type for this study. It is the unique RFS completely specified by its intensity function. Its name comes from the Poisson point process. If the RFS is Poisson, i.e., the number of points is Poisson distributed and the points themselves are IID (Independently and Identically Distributed), then the probability density of  $\mathbf{X}$  can be constructed exactly from the PHD [55]. Summarily, its cardinality distribution, multi-object PDF, and intensity function follow, respectively, Equations (11)–(13).

$$\rho(n) = \frac{e^{-\lambda}\lambda^n}{n!}, \quad n = 0, 1, 2, \dots \quad (11)$$

where  $\lambda > 0$  is the expected cardinality of  $\mathbf{X}$ .

$$f(\mathbf{X}) = e^{-\lambda} \prod_{\mathbf{x} \in \mathbf{X}} \lambda p(\mathbf{x}). \quad (12)$$

$$D(\mathbf{X}) = \lambda p(\mathbf{x}). \quad (13)$$

In this context,  $p(\mathbf{x})$  refers to the Probability Density Function (PDF) of a random variable  $\mathbf{x}$ , which can be distributed as Gaussian or Uniform, for example.

### 3.3. PHD Filter Definition

A simple approach to set-based estimation is to exploit the physical intuition of the first moment of an RFS, known as its PHD or intensity function [55]. This approach is consistent with the multitarget equivalent of expected value—the expectation of an RFS. As stated previously, the PHD function corresponds to a mass density, and the mass corresponds to the expected value of targets in some state-space region  $S \subseteq \mathcal{X}$ . For this reason, it is possible to link the PHD function and Bayesian framework.

In order to obtain a definition for the PHD function, there are two intuitive ways. The first one would be to obtain the PHD as a mathematical expectation of a point process

density, as previously presented in Equation (8). The second one would be to treat the PHD as the limit of an occupancy grid probability.

As stated by Erdinc et al. [56], the surveillance region can be partitioned into bins, and it is assumed that each bin has the same volume. Furthermore, these bins are sufficiently small so that each bin is potentially occupied by at most one target [56]. Based on this, if  $m_i$  denote the centre and  $B(m_i)$  the region defined by the boundaries of the  $i$ th grid cell, then the expected number of features over the region  $S_J = \cup_{i \in J} B(\mathbf{x}_i)$  is given by:

$$\mathbb{E}\{|\mathcal{X} \cap S_J|\} = \sum_{i \in J} P^{occ}(B(\mathbf{x}_i)) = \sum_{i \in J} D(\mathbf{x}_i)\lambda(B(\mathbf{x}_i)), \tag{14}$$

where  $\mathcal{X} \cap S_J$  is the intersection between the state space  $\mathcal{X}$  and surveillance area  $S_J$ ,  $P^{occ}(B(\mathbf{x}_i))$  denote the occupancy probability,  $\lambda(B(\mathbf{x}_i))$  is the area of the  $i$ th grid cell, and:

$$D(\mathbf{x}_i) = \frac{P^{occ}(B(\mathbf{x}_i))}{\lambda(B(\mathbf{x}_i))} \tag{15}$$

is the intensity function. As the grid cell area tends to zero (or for an infinitesimally small cell),  $B(\mathbf{x}_i) \rightarrow d\mathbf{x}_i$ , the sum then becomes an integral, and the expected number of features in  $S$  becomes:

$$\mathbb{E}\{|\mathcal{X} \cap S_J|\} = \int_S D(\mathbf{x})d\mathbf{x}. \tag{16}$$

$D(\mathbf{x})$  defines the PHD and it can be interpreted as the occupancy probability density at the point  $\mathbf{x}$  [55]. The main objective for MTT RFS-based is propagating the intensity function to estimate targets state recursively based on the Bayesian framework.

Bayes theorem allows computing the probability of an event based on prior knowledge of conditions related to the event and some new evidence. This concept is the basis of Bayesian filters, which estimate the current state of a dynamic system in a probabilistic way. This process consists of two steps: Prediction and update, according to:

$$p(\mathbf{x}_k | \mathbf{z}_{1:k-1}) = \int p(\mathbf{x}_k | \mathbf{x}_{k-1})p(\mathbf{x}_{k-1} | \mathbf{z}_{1:k-1})d\mathbf{x}_{k-1}, \tag{17}$$

and

$$p(\mathbf{x}_k | \mathbf{z}_{1:k}) = \frac{p(\mathbf{z}_k | \mathbf{x}_k)p(\mathbf{x}_k | \mathbf{z}_{1:k-1})}{\int p(\mathbf{z}_k | \mathbf{x}_k)p(\mathbf{x}_k | \mathbf{z}_{1:k-1})d\mathbf{x}_k}, \tag{18}$$

where  $\mathbf{x}$  is the state vector, and  $\mathbf{z}$  is the observation vector, both in the conventional sense. Equations (17) and (18) are the recursive algorithm presentation of the Bayes Theorem.

Because of the Markov assumption, the posterior probability of the state is only determined based on the prior distribution, i.e., it is conditionally independent of the other earlier states. This premise is the basis for recursive estimation.

The presented filter has only conceptual importance. Because of the integral in Equation (17), it is computationally tractable only for a few discrete cases. However, all Bayesian recursive filters, such as Kalman Filter, are derived from it.

The recursive RFS filter is defined by Equations (19) and (20), similarly to the Bayesian filter one. Given the multitarget state  $\mathbf{X}_k$ , and measurement  $\mathbf{Z}_k$  in the random finite set framework, suppose that at time  $k - 1$  the posterior FISST PDF of the multi-object state,  $f_{k-1}(\mathbf{X}_{k-1} | \mathbf{Z}_{1:k-1})$  is known. Here  $\mathbf{Z}_{1:k-1} \equiv \mathbf{Z}_1, \dots, \mathbf{Z}_{k-1}$  is the sequence of all previous measurements. Then, respectively, the predicted and updated posterior multi-object densities is expressed as follows [53]:

$$f_{k|k-1}(\mathbf{X}_k | \mathbf{Z}_{1:k-1}) = \int \Pi_{k|k-1}(\mathbf{X}_k | \mathbf{X}')f_{k|k-1}(\mathbf{X}' | \mathbf{Z}_{1:k-1})\delta\mathbf{X}', \tag{19}$$

$$f_k(\mathbf{X}_k | \mathbf{Z}_{1:k}) = \frac{\varphi_k(\mathbf{Z}_k | \mathbf{X}_k)f_{k|k-1}(\mathbf{X}_k | \mathbf{Z}_{1:k-1})}{\int \varphi_k(\mathbf{Z}_k | \mathbf{X})f_{k|k-1}(\mathbf{X} | \mathbf{Z}_{1:k-1})\delta\mathbf{X}'} \tag{20}$$

where  $\mathbf{X}'$  is the set of targets previously observed,  $\varphi_k(\mathbf{Z}_k|\mathbf{X}_k)$  is the observation likelihood function, and  $\Pi_{k|k-1}(\mathbf{X}_k|\mathbf{X}')$  is the FISST transitional density (representing the multitarget motion model in the RFS sense). The posterior density is the estimate of a set of points from the multi-object filter at each time step. It is a target state estimate collection (unlabeled and unordered). Similar to the integral present in the Bayes filter, computing the exact multi-object posterior density is numerically intractable because of the set integrals. All the practical algorithms are approximations, and the PHD filter is one of these algorithms.

Mahler [30] derives a recursive Bayes filter for a PHD filter that accounts for multiple sensors, nonconstant probability of detection, Poisson false alarms, as well as the appearance, spawning, and disappearance of targets. The PHD is a best-fit approximation of the multitarget posterior in an information-theoretic sense and propagates the first-order statistical moment of the multitarget posterior.

For a Poisson point process, as described previously, the prediction and update recursive filter equations are [54]:

$$D_{k|k-1}(\mathbf{x}) = b_k(\mathbf{x}) + p_S \int \pi_{k|k-1}(\mathbf{x}|\mathbf{x}') D_{k-1}(\mathbf{x}') d\mathbf{x}', \quad (21)$$

$$D_k(\mathbf{x}) = D_{k|k-1}(\mathbf{x}) \times \left[ 1 - p_D + \sum_{\mathbf{z} \in \mathbf{Z}_k} \frac{p_D g_k(\mathbf{z}|\mathbf{x})}{c_k(\mathbf{z}) + \int p_D g_k(\mathbf{z}|\mathbf{x}) D_{k|k-1}(\mathbf{x}) d\mathbf{x}} \right], \quad (22)$$

respectively, where  $\mathbf{x}$  is the state of a single object (a random vector),  $\mathbf{z}$  is a measurement of a single object (a random vector),  $b_k(\mathbf{x})$  is the PHD of the births at instant  $k$ ,  $p_S$  is the probability that a target still exists at time  $k$ , and  $\pi_{k|k-1}$  is the target transition function,  $D_{k-1}$  is the prior intensity,  $D_k$  is the posterior intensity,  $p_D$  is the probability that a target is detected at time  $k$ ,  $c_k(\mathbf{z})$  is the clutter PHD at time  $k$ ,  $g_k(\mathbf{z}|\mathbf{x})$  is the target observation model. For simplicity, the probabilities of detection  $p_S$  and  $p_D$  presented here are independent of the state.

For computational implementations, see [38,39] for approximations based on Gaussian Mixture and [37,57] for particle methods.

This work implements a Gaussian Mixture realization. The GM is a weighted sum of Gaussians, according to:

$$\text{GM}(x) = \sum_{i=1}^{J_k} w_k^{(i)} \mathcal{N}(x; m_k^{(i)}, P_k^{(i)}), \quad (23)$$

where  $w_k^{(i)}$  is the weight,  $m_k^{(i)}$  is the mean, and  $P_k^{(i)}$  is the covariance matrix of the  $i$ th component from total  $J_k$ .

For practical implementations, this algorithm needs complementary steps when compared to the canonical form of the Bayesian sequence. The GM PHD filter suffers from computational problems associated with the increasing number of Gaussian components as time progresses. This issue implies that the number of components in the posterior intensities increases without bound [38].

The prune & merge methods are the two most implemented methods to solve these issues in practical implementations. The first one consists of reaping components of the Gaussian mixture with values below a limit of  $T$ . Combined with the pruning operation, the merge method is applied to prevent unbounded growth. This method consists of clustering GM components in a region defined by the Mahalanobis distance  $L$  and delimited by threshold  $U$  according to:

$$L := \left\{ i \in I = w_k^{(i)} > T \mid (m_k^{(i)} - m_k^{(j)})^T (P_k^{(i)})^{-1} (m_k^{(i)} - m_k^{(j)}) \leq U \right\}, \quad (24)$$

where  $j$  refers to the component with the highest weight in set  $I$ . This distance measures how some points are distant from the mean, reflecting the sample dispersion considering the covariance matrix. An intuitive interpretation is that the more the data is correlated, the shorter the distance between them will be, forming a cluster.



The last step is the state extraction, which eliminates GM components with weak weights after the prune & merge step. A better alternative is to select the means of the Gaussians that have weights greater than an *Extraction threshold*  $E$  [38]. It is important to note that this step does not interfere with the filter performance.

Algorithm 1 presents the relationship between the filter equations and the execution steps for the GM PHD approximation .

---

**Algorithm 1** *GMPHD*( $D(x_{k-1})$ )

---

for Every instant  $k$

do  
 Prediction based on Equation (21)  
 Update based on Equation (22)  
 Prune and Merge based on Equation (24)  
**comment:** Extraction based on:  
 Set  $\hat{X}_k := \emptyset$   
 for  $i \leftarrow 1$  to  $J_k$   
 do { if  $w_k^{(i)} > E$   
     then  $\hat{X}_k := [\hat{X}_k, m_k^{(i)}]$   
 }  
 return ( $D(x_k), \hat{X}_k$ )

---

### 3.4. Optimal Subpattern Assignment (OSPA) Metric

The concept of error between a reference quantity and its estimated value plays a fundamental role in any filtering problem [58]. Unlike the idea of miss-distance in single-object tracking systems, such as the error between actual and estimated state, there is no direct way to measure this error in the multi-object case. As stated by Schuhmacher et al. [58], a satisfactory multi-object miss-distance needs to capture the “difference” between two sets of vectors, namely the reference multi-object state and the estimated multi-object state, in a mathematically consistent yet physically meaningful manner.

In short, the OSPA metric is the “distance” between a set of tracks and the known truths. This metric contains two error measures between those sets: Localization error component (accounts for state estimation) and cardinality error component (a benchmark for the number of missed targets). There is a third component out of the scope of this work called the labeling error component, which accounts for an incorrect assignment.

For two finite sets  $X$  and  $Y$  with respective  $m$  and  $n$  cardinalities, for  $m \leq n$ , the OSPA distance metric is defined according to [58]:

$$\bar{d}_p^{(c)}(X, Y) := \left( \frac{1}{n} \left( \min_{\pi \in \Pi} \sum_{i=1}^m (d^{(c)}(x_i, y_{\pi(i)})^p + c^p(n-m)) \right) \right)^{\frac{1}{p}}, \quad (25)$$

where  $d^{(c)}(x, y) := \min(c, d(x, y))$  is the cutoff distance between two elements of  $X$  and  $Y$  with  $c > 0$  being the cutoff parameter,  $\Pi_n$  represents the set of permutations of length  $m$  with elements taken from  $\{1, 2, \dots, n\}$ , and  $1 \leq p < \infty$ .

The cutoff parameter  $c$  determines the relative weighting of the cardinality error component against the base distance error component. Larger values of  $c$  tend to emphasize cardinality errors and vice versa. The order parameter  $p$  controls the penalty assigned to “outlier” estimates (which are not close to the ground truth tracks). A higher value of  $p$  increases sensitivity to outliers [59].

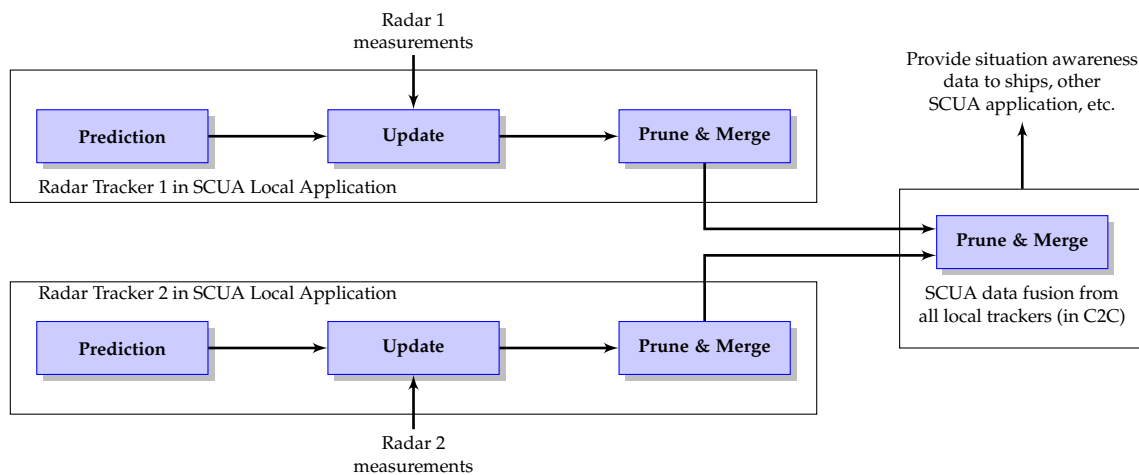
## 4. Application for Tracking Maritime Surveillance

The SCUA integrates multisensor and multisource data to evaluate situational awareness and support decision-making activities for the maritime area of interest. Both types of sensors, shipboard or stationary, can belong to the distributed network. Therefore, different devices such as cameras, radars, and sonars can integrate the system.

For small ships, computers (sometimes IoT devices or tablets) have a limited processing capacity. System components exchange data via communication links such as 4G, LoRaWan, or satellite. As a result of processing capacity and bandwidth limitations, the tracker should have a low computational load. Allied to this, it should produce good results in a low clutter environment with a high probability of detection.

Each network member has a local application, which runs SCUA with several features and services, with the tracker being one of those. SCUA has parallel processes and computing (based on OpenCL), depending on the hardware used in the node. The parallelization method is not concerning in this manuscript.

Figure 1 presents the implementation scheme for PHD filtering in local trackers and the data fusion in Central Station (C&C), for the case illustrated in this work. The simulations presented in this section use this same scheme.



**Figure 1.** Probability Hypothesis Density ( PHD) filtering and data fusion implementation scheme (example of two sensors available).

#### 4.1. Surveillance Scenario

Due to confidentiality issues, only simulations based on Matlab Sensor Fusion<sup>®</sup> objects are used to illustrate the results of the application. In this simulation scenario, the distributed network is composed of two stationary sensors (2D radars). Both radars stare into the harbor, covering a 90-degree azimuth sector. The sensors are diagonally positioned in the surveillance area in opposite directions, according to Table 1. Each of them has a tracker responsible for monitoring targets in their respective FoV and providing the posterior intensity distribution.

This scenario was defined according to the main aspects: Availability and ease of access to data. There were no more sensors available at the collection and analysis of actual data to prepare the simulation. Furthermore, data obtained from sensors installed on ships could expose sensitive performance data. Given Brazilian regulations, this additional data would require significant effort to consult competent authorities. Despite this data limitation, this scenario proved appropriate for this work.

**Table 1.** Radars and Central Station in simulation.

Entity	Coordinates [km]	Angular Sector Covered by FoV [rad]
Radar 1	(7.38, -0.236)	$[0, -\pi/2]$
Radar 2	(-4, 9)	$[-\pi/2, \pi/2]$
Central Station (Tower)	(0, 0)	Not Applicable

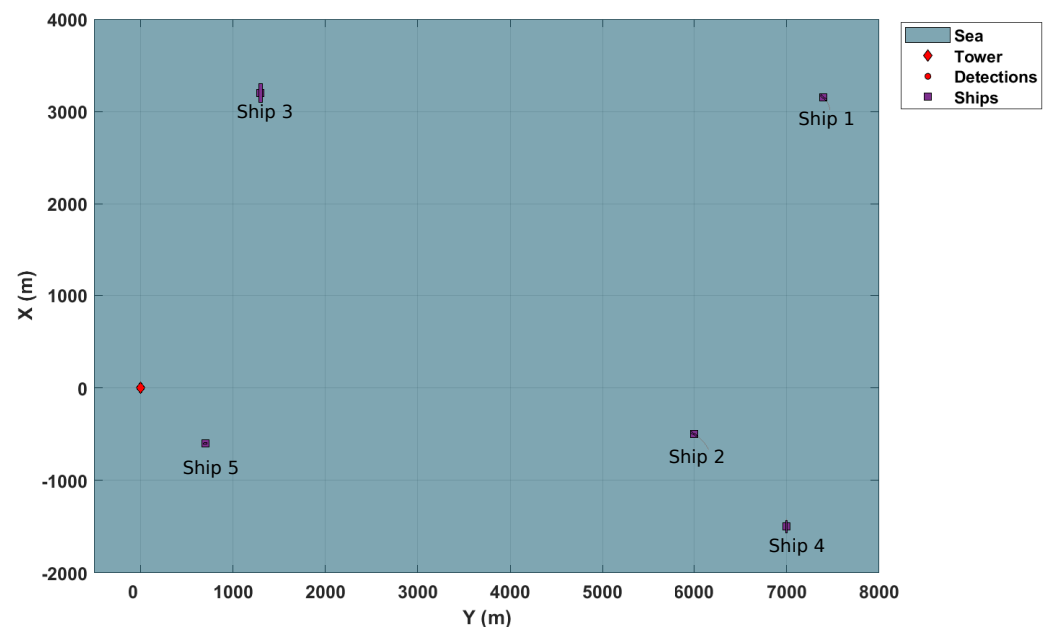
The effects of latency and bandwidth limitations in communication are not considered in the simulations. Due to the distances involved, the effect of the earth's curvature in the observations is also not considered.

Additionally, there are vessels typically found in the maritime area of interest in simulation. There are five vessels in the harbor within the surveillance sector. Two of them are turning at 20 and 30 knots. The others are traveling with a constant heading of 10, 12, and 6 knots. Table 2 resumes information concerning the initial states of the ships.

**Table 2.** Ships initial states and type of movement.

Ship	Initial Speed [knot]	Initial Orientation [Deg]	Initial Coordinates [km]	Type of Movement
1	20	130	(3.15, 7.4)	Circular with radius 200 m
2	30	120	(−0.5, 6)	Circular with radius 400 m
3	10	0	(3.2, 1.3)	Constant heading
4	12	0	(−1.5, 7)	Constant heading
5	6	90	(−0.6, 0.7)	Constant heading

Figure 2 illustrates the surveillance scenario described here.



**Figure 2.** Maritime surveillance scenario.

#### 4.2. Dynamic Modeling

Each of these five vessels in the simulation scenario has a kinematic state vector denoted by:

$$\mathbf{x} = [x \ \dot{x} \ y \ \dot{y} \ \omega]^T, \quad (26)$$

where  $(x, y)$  corresponds to position coordinates,  $(\dot{x}, \dot{y})$  corresponds to velocities,  $\omega$  is the constant turn-rate, and  $T$  denotes the transpose operation.

Another crucial dynamic model is the one used by the filter itself. Despite the problem of multitarget tracking, the dynamic model of a single target remains fundamental to establish the representative dynamics of each individualized target. Therefore, each target needs a motion model for the filter prediction step. The choice of this model is directly linked to the filter performance.

Tracking targets with coordinated turn motion is highly dependent on the models and algorithms [60]. Since the targets are possibly small, fast, and easy-to-maneuver vessels,

the nonlinear nearly constant turn model is the best choice [61]. For a single target, the state dynamics is given by [38]:

$$\mathbf{x}_k = F(\omega_{k-1})\mathbf{x}_{k-1} + Gw_{k-1} \quad (27)$$

$$\omega_k = \omega_{k-1} + \Delta u_{k-1}, \quad (28)$$

where  $\mathbf{x}_k = [x_k \ y_k \ \dot{x}_k \ \dot{y}_k]^T$  corresponds to 2D position and velocities coordinates at instant  $k$ ,  $\omega$  stands for the turn rate in time  $k$ ,  $w = [w_x \ w_y]^T$  is a zero-mean Gaussian process noise with  $\sigma_w \in \mathbb{R}^{2 \times 2}$  covariance,  $u$  is a random variable with zero-mean,  $\sigma_u$  is the covariance,

$$F(\omega) = \begin{bmatrix} 1 & 0 & \frac{\sin \omega \Delta}{\omega} & -\frac{1 - \cos \omega \Delta}{\omega} \\ 0 & 1 & \frac{1 - \cos \omega \Delta}{\omega} & \frac{\sin \omega \Delta}{\omega} \\ 0 & 0 & \cos \omega \Delta & -\sin \omega \Delta \\ 0 & 0 & \sin \omega \Delta & \cos \omega \Delta \end{bmatrix},$$

$$G = \begin{bmatrix} \frac{\Delta^2}{2} & 0 \\ 0 & \frac{\Delta^2}{2} \\ \Delta & 0 \\ 0 & \Delta \end{bmatrix},$$

and  $\Delta$  is the sample time.

For each target and stationary sensor, the observation model consists of azimuth and range measurements, according to:

$$z_k = \begin{bmatrix} \sqrt{(x_k - x_s)^2 + (y_k - y_s)^2} \\ \arctan\left(\frac{y_k - y_s}{x_k - x_s}\right) \end{bmatrix} + \epsilon_k \quad (29)$$

where  $(x_k, y_k)$  is the target position at instant  $k$ ,  $(x_s, y_s)$  is the sensor position, and  $\epsilon_k$  is a zero-mean Gaussian process noise with  $R_k \in \mathbb{R}^{2 \times 2}$  covariance.

#### 4.3. Filter Parameters

The Gaussian Mixture PHD filters used in the simulations are that described in Equations (21) and (22). Each of them enables us to estimate the positions of the ships individually, as described before. Since the predictive model is nonlinear, the filter is the EKF version (see [38] for details about the Jacobians). Table 3 presents the filter parameters used in simulations.

**Table 3.** Filter parameters used for tracking targets.

Filter Parameter	Value
Sensor maximum range	12 km
Distance resolution noise	25 m
Azimuth resolution noise	0.5°
Probability of Survival ( $p_S$ )	0.99
Probability of detection ( $p_D$ )	0.9
Sensor field-of-view (FoV)	90°
Clutter density	$2 \times 10^{-8}$
Prune threshold ( $T$ )	$10^{-6}$
Merge threshold ( $U$ )	25
Extraction threshold ( $E$ )	0.8
Maximum number of components	1000

For simulation purposes, there is no spawning, and the spontaneous birth RFS is Poisson, i.e., added uniformly inside the sensor's field of view, with a rate of  $10^{-5}$ . The births have a unitary initial weight, and an initial covariance matrix:

$$P_{birth} = \begin{bmatrix} 10^5 & 0 & 0 & 0 & 0 \\ 0 & 10^5 & 0 & 0 & 0 \\ 0 & 0 & 10^5 & 0 & 0 \\ 0 & 0 & 0 & 10^5 & 0 \\ 0 & 0 & 0 & 0 & 1000 \end{bmatrix}.$$

Finally, the clutter RFS follows the uniform Poisson model over the sensor FoV, the surveillance region being  $[-\pi/2, \pi/2]$  rad  $\times$   $[0, 12]$  km.

Since each radar independently estimates the coordinates of the targets, this information needs to be gathered. Then, the simulated central processing merges these distributions, obtaining the final solution in a distributed and cooperative way. Similarly to [62], the merging process is:

$$\text{Cooperative Tracking} = \text{merge}(\{D(\mathbf{x})^1, D(\mathbf{x})^2, \dots, D(\mathbf{x})^R\}), \quad (30)$$

where  $D(\mathbf{x})^i$  is the intensity estimated by  $i$ th distributed sensor, and  $R$  is the number of distributed sensors. Merge operation is a clustering technique. For RFS applications, the algorithms based on Mahalanobis distance are the most common method, which is detailed in [38].

Figure 3 resumes, in a block diagram, all the steps of the simulation setup (surveillance scenario, models, and filter parameters).

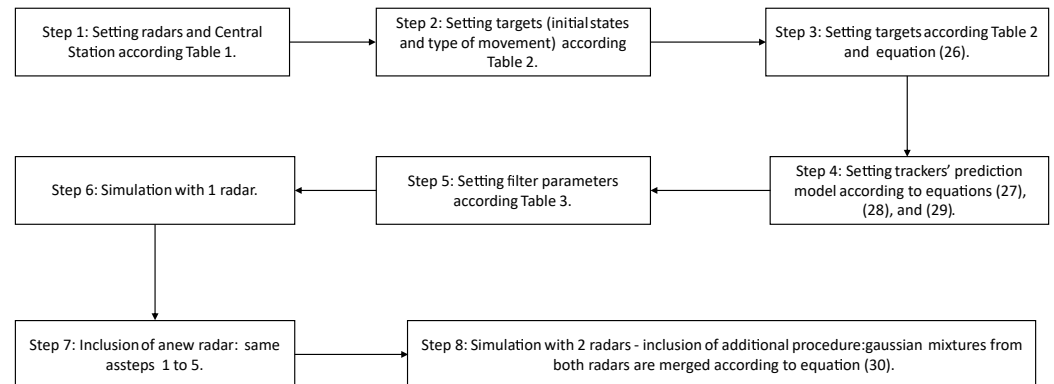


Figure 3. Block diagram of the proposed scheme.

#### 4.4. Results and Discussions

Based on OSPA metrics shown in Figure 10, the results presented in the simulations are satisfactory. Initially, to illustrate the performance achieved, the simulated tracking system uses only one radar, whose FoV is represented by the region delimited by red lines (as in the other figures). The tracker exhibits good results, i.e., it can track the targets observed by the sensor. Figures 4 and 5 present these results.



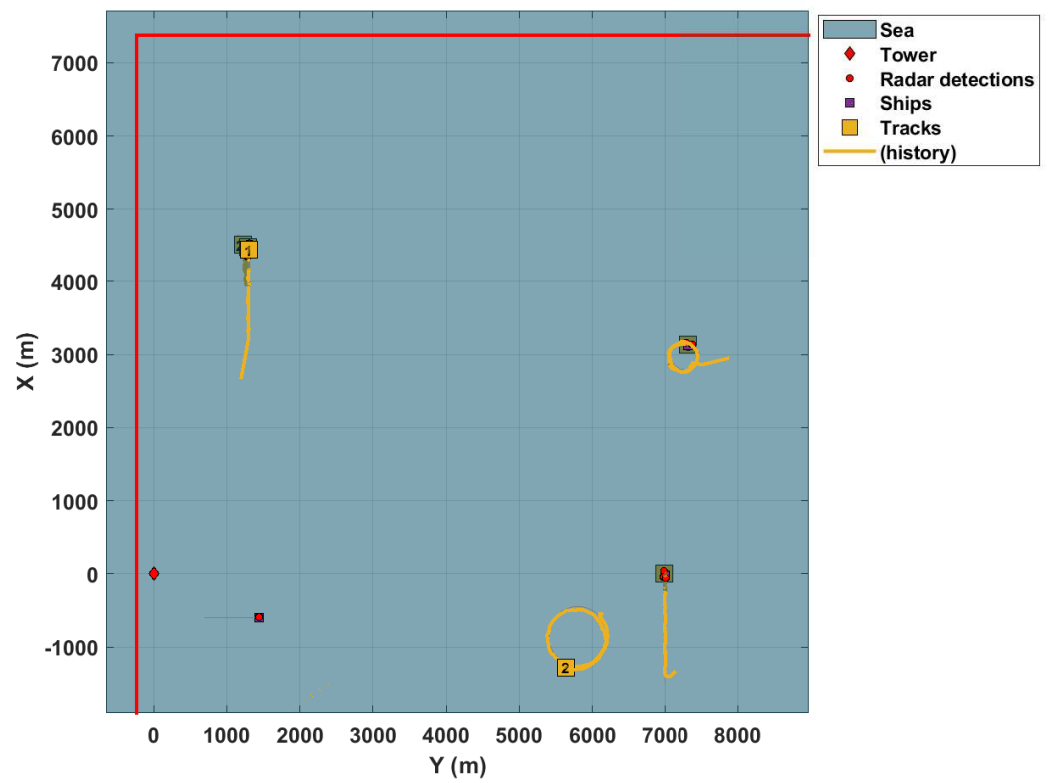


Figure 4. Tracking with one sensor.

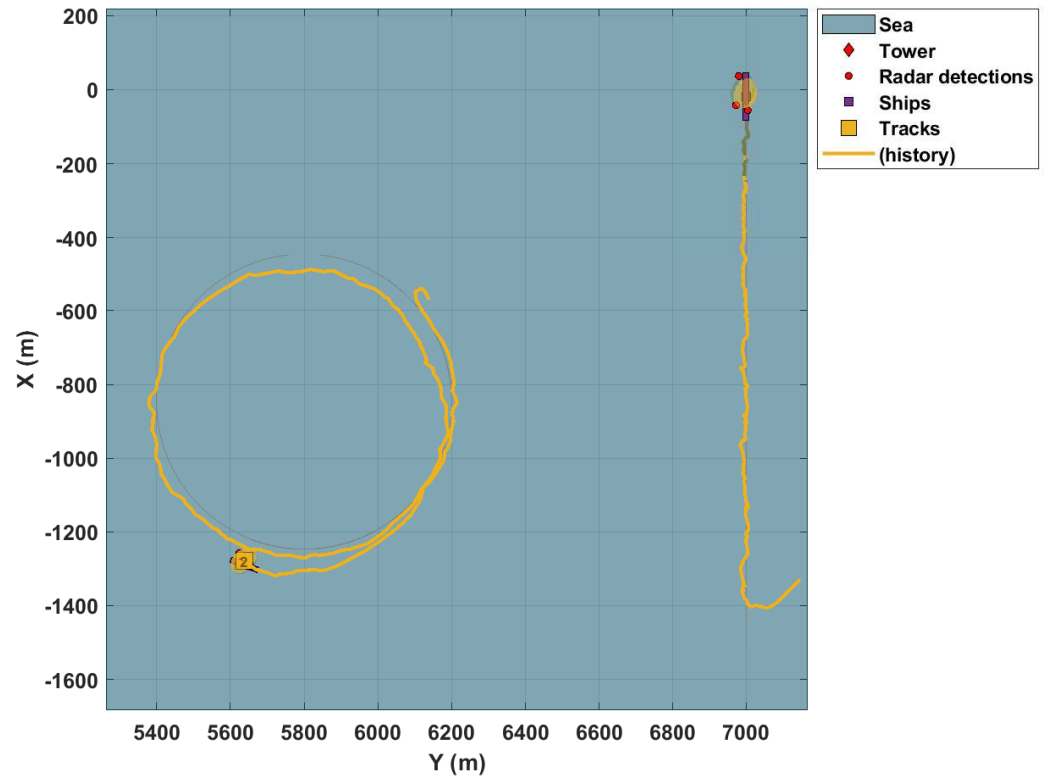


Figure 5. Tracking of two targets with different trajectories.

As stated before, these figures demonstrate the tracking system's performance with just one sensor. In addition, these figures show that the ships are tracked regardless of whether the movement performed is linear or circular, despite an initial transitory. It is noteworthy that the vessels present significant differences in their speed and movements. At the same time, one navigates at 12 knots, the other spins at 30 knots.

Figure 5 is an enlarged view of two targets with different movements, one linear and the other circular. This figure reinforces the tracking capacity of the system in different conditions of movement of the target. Furthermore, it attests the good choice of the dynamic turn-rate model for the targets. This choice is an essential factor for the success of the tracking system, as stated by [60].

Despite the successful choice for the prediction model, the figure illustrates an existing problem in real-life systems called target occlusion. Figure 6 shows this weakness when using only one radar. The larger vessel occludes the smaller one. Therefore, it is not detected by the tracking system. Given that the application aims to combat illegal activities usually carried out in the region by small vessels, it is essential to use sensors distributed throughout the surveillance area. This fact is further reinforced by the large dimensions of the surveillance area.

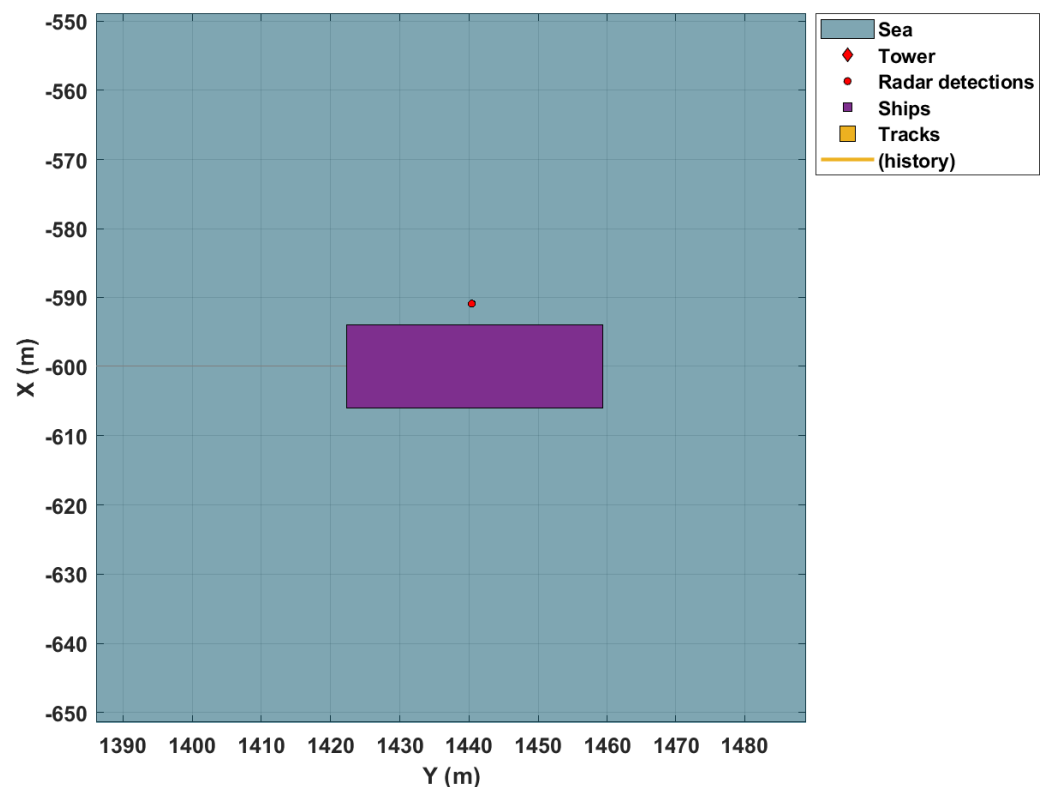


Figure 6. Target in occlusion.

Inserting a second radar prevents targets occlusion, as illustrated by Figure 7.

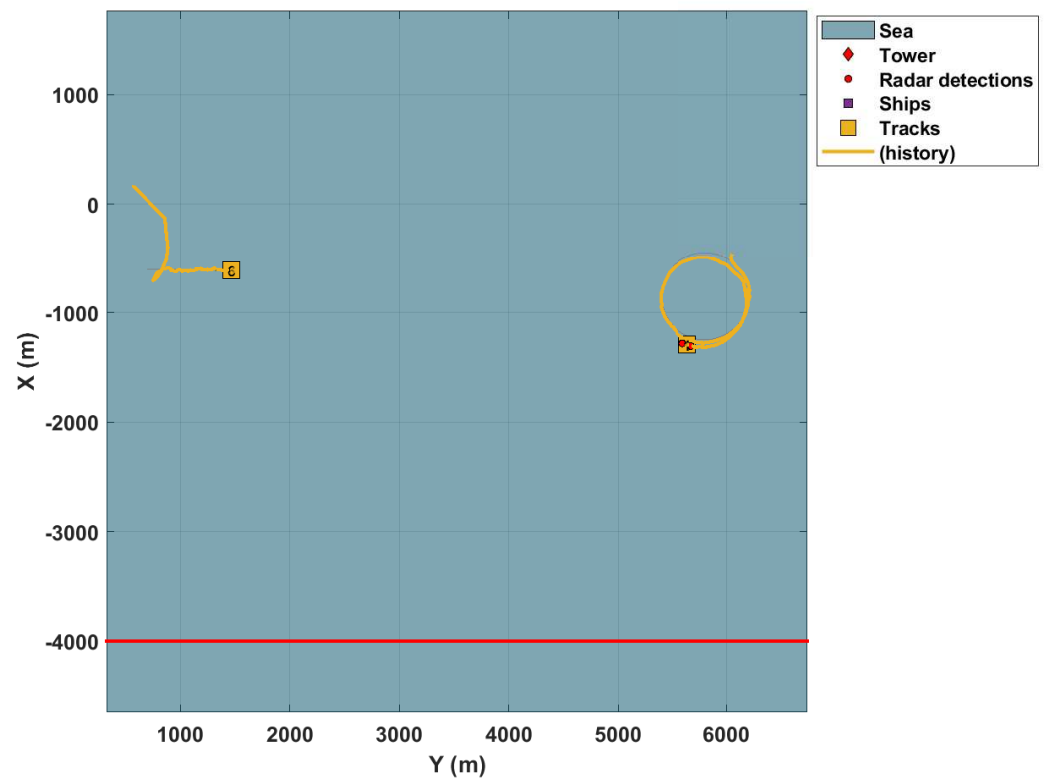


Figure 7. Second radar in the surveillance area.

Figures 8 and 9 present the results with combined trackers, i.e., two radars.

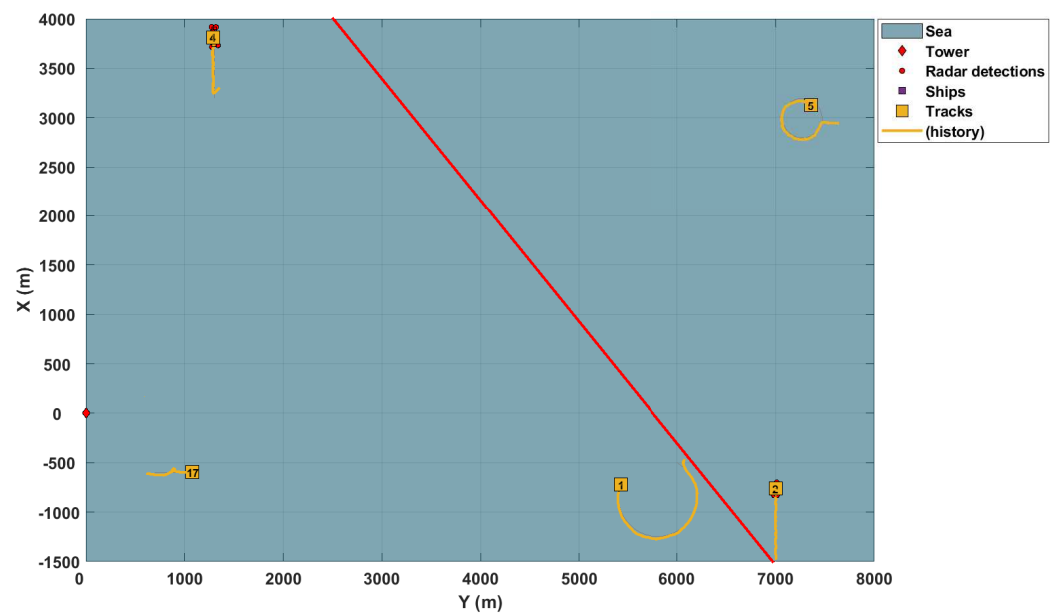


Figure 8. Complete tracking with merged PHDs.

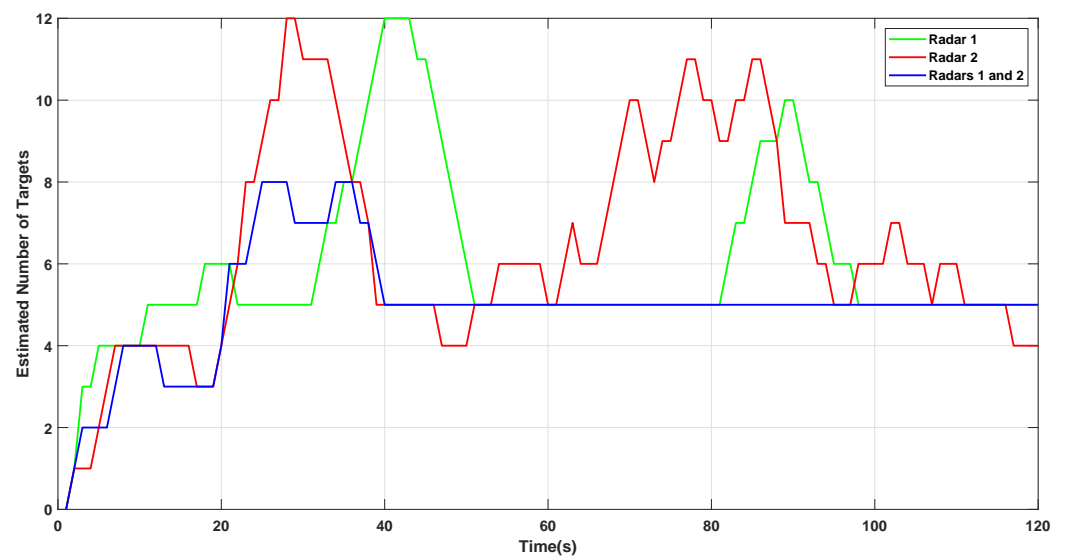


Figure 9. The number of estimated targets by individual trackers, and merged.

It is possible to observe that combining information from each tracker allows a faster convergence to the number of actual targets. In addition, it is more stable since the increase in data sources (sensors) in the system avoids targets occlusion compared to using only one radar.

Finally, Figure 10 presents the OSPA metric and its components, which corroborates the analyses based on the previous figures. The merged tracker is faster and more stable, as can be observed in the plots, in particular after a period of 40 s. The lower the metric value, the better the filter’s performance. Furthermore, the combined filter with data from both radars shows better results for localization and cardinality OSPA components. The location component is related to the position error, as can be seen in Figure 10, after 40 s, the value tends to zero. Similarly, at the same instant, the cardinality component becomes zero.

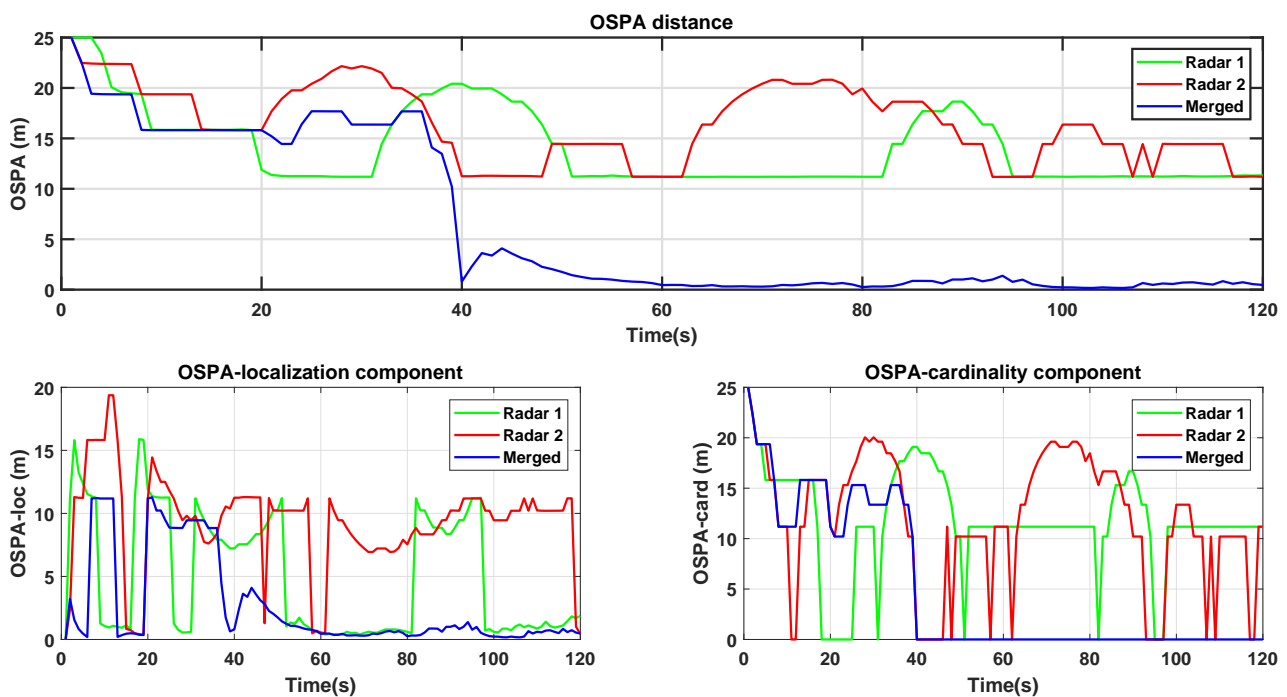


Figure 10. OSPA metrics.

The simulations illustrate the problem of target occlusion when they present different geometries and dimensions. Inserting a new sensor and the consequent increase in the sensor network prevents targets occlusion and improves the overall performance of the tracking system. These results reinforce the statement by Vo and Ma [38]: The efficiency and simplicity in the implementation of the Gaussian mixture PHD recursion also suggest a possible application to tracking in sensor networks.

The algorithm has performance limitations already demonstrated in other researches. The proposed architecture presents good results when applying an additional step to the original algorithm, which is the main contribution of this work: A low-complexity data fusion scheme. Additionally, this proposed scheme presents promising results, especially when dealing with the target occlusion problem.

## 5. Conclusions

This paper introduces a Maritime Surveillance System for the Brazilian Coast. The system has a complex network of distributed sensors and processing stations. The demanding requirements for its operation are one of the motivations for the present work. In this context, this work presents an algorithm to track an unknown number of marine targets from multiple sensors in a large-scale surveillance area. A merged version of the Gaussian Mixture PHD filter is applied to a realistic multitarget tracking scenario. The proposed filter shows promising results through simulations, even for targets with different sizes, movements, and speeds. The proposed data fusion scheme improves the overall performance of the tracking system and overcomes targets' occlusion. This fact further evidences, for the application, the need for a network of distributed sensors, given the system's mission and the extension of the area of interest. Additionally, it is possible to conclude that increasing the covered area using sensors of a smaller capacity also contributes to the economic aspect of this modular undertaking.

For future work, one of the focuses would be investigating different RFS-based filters, the capabilities of the extended algorithms, and different sensor types such as stationary passive sonars or shipboard radars. Moreover, the computational models used for parallelism and multiprocessing should be investigated. In particular, those based on Gamma models, once SCUA data fusion applications already use this approach. These future works will attest, corroborate the empirical results obtained in the simulations, and contribute to using these algorithms in a new computational paradigm.

**Author Contributions:** Conceptualization, K.M.d.L. and R.R.C.; methodology, K.M.d.L.; software, K.M.d.L.; validation, K.M.d.L. and R.R.C.; writing—original draft preparation, K.M.d.L.; writing—review and editing, K.M.d.L. and R.R.C. All authors have read and agreed to the published version of the manuscript.

**Funding:** This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 001.

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

## References

1. Plano Estratégico da Marinha. Strategic Programs. Available online: <https://www.marinha.mil.br/programas-estrategicos> (accessed on 2 April 2020).
2. Mahler, R.P.S. Random-set approach to data fusion. In Proceedings of the SPIE's International Symposium on Optical Engineering and Photonics in Aerospace Sensing, Orlando, FL, USA, 4–8 April 1994. [CrossRef]
3. Maher, R. A survey of PHD filter and CPHD filter implementations. In Proceedings of the Defense and Security Symposium, Orlando, FL, USA, 9–13 April 2007. [CrossRef]
4. Gao, L.; Tang, X.; Wei, P. Real-Time Implementation of Particle-PHD Filter Based on GPU. In Proceedings of the 2014 IEEE 17th International Conference on Computational Science and Engineering, Chengdu, China, 19–21 December 2014. [CrossRef]
5. Zheng, Y.; Shi, Z.; Lu, R.; Hong, S.; Shen, X. An Efficient Data-Driven Particle PHD Filter for Multitarget Tracking. *IEEE Trans. Ind. Inform.* **2013**, *9*, 2318–2326. [CrossRef]
6. Li, T.; Sun, S.; Bolić, M.; Corchado, J.M. Algorithm design for parallel implementation of the SMC-PHD filter. *Signal Process.* **2016**, *119*, 115–127. [CrossRef]



7. Li, T.; Hlawatsch, F.; Djuric, P.M. Cardinality-Consensus-Based PHD Filtering for Distributed Multitarget Tracking. *IEEE Signal Process. Lett.* **2019**, *26*, 49–53. [[CrossRef](#)]
8. Li, T.; Mallick, M.; Pan, Q. A Parallel Filtering-Communication-Based Cardinality Consensus Approach for Real-Time Distributed PHD Filtering. *IEEE Sens. J.* **2020**, *20*, 13824–13832. [[CrossRef](#)]
9. Li, G.; Yi, W.; Kong, L. A real-time fusion algorithm for asynchronous sensor system with PHD filter. In Proceedings of the 2017 International Conference on Control, Automation and Information Sciences (ICCAIS), Chiang Mai, Thailand, 31 October–1 November 2017. [[CrossRef](#)]
10. Li, G.; Yi, W.; Jiang, M.; Kong, L. Distributed fusion with PHD filter for multi-target tracking in asynchronous radar system. In Proceedings of the 2017 IEEE Radar Conference (RadarConf), Seattle, WA, USA, 8–12 May 2017. [[CrossRef](#)]
11. Barbosa, R. Fusão de Alvos Utilizando Grafos em Ambientes de Múltiplos Sensores. Ph.D. Thesis, COPPE/UFRJ, Rio de Janeiro, Brazil, 2012.
12. Livernet, F.; Cuilliere, O. Tracking based on graph theory applied to pairs of plots. In Proceedings of the 2010 IEEE Radar Conference, Arlington, VA, USA, 10–14 May 2010; pp. 90–95.
13. Mello, R.R.; de Almeida, R.H.P.; Franca, F.M.; Paillard, G.A. A Parallel Implementation of Data Fusion Algorithm Using Gamma. In Proceedings of the 2015 International Symposium on Computer Architecture and High Performance Computing Workshop (SBAC-PADW), Florianopolis, Brazil, 18–21 October 2015. [[CrossRef](#)]
14. Pace, M. Comparison of PHD based filters for the tracking of 3D aerial and naval scenarios. In Proceedings of the 2010 IEEE Radar Conference, Arlington, VA, USA, 10–14 May 2010. [[CrossRef](#)]
15. Pailhas, Y.; Houssineau, J.; Petillot, Y.R.; Clark, D.E. Tracking with MIMO sonar systems: Applications to harbour surveillance. *IET Radar Sonar Nav.* **2017**, *11*, 629–639. [[CrossRef](#)]
16. Do, C.T.; Nguyen, T.T.D.; Liu, W. Tracking Multiple Marine Ships via Multiple Sensors with Unknown Backgrounds. *Sensors* **2019**, *19*, 5025. [[CrossRef](#)]
17. Carthel, C.; Coraluppi, S.; Grignan, P. Multisensor tracking and fusion for maritime surveillance. In Proceedings of the 2007 10th International Conference on Information Fusion, Québec, QC, Canada, 9–12 July 2007.
18. Vo, B.; Mallick, M.; Bar-shalom, Y.; Coraluppi, S.; Osborne, R., III; Mahler, R.; Vo, B. Multitarget Tracking. *Wiley Encycl. Electr. Electron. Eng.* **2015**, *2015*, 1–15.
19. Arulampalam, M.; Maskell, S.; Gordon, N.; Clapp, T. A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking. *IEEE Trans. Signal Process.* **2002**, *50*, 174–188. [[CrossRef](#)]
20. Bazzani, L.; Bloisi, D.D.; Murino, V. A Comparison of Multi Hypothesis Kalman Filter and Particle Filter for Multi-Target Tracking. CVPR 2009. 2009. Available online: <https://www.researchgate.net/publication/228373867> (accessed on 3 April 2020).
21. Jinan, R.; Raveendran, T. Particle Filters for Multiple Target Tracking. *Procedia Technol.* **2016**, *24*, 980–987. [[CrossRef](#)]
22. Wang, X.; Li, T.; Sun, S.; Corchado, J. A Survey of Recent Advances in Particle Filters and Remaining Challenges for Multitarget Tracking. *Sensors* **2017**, *17*, 2707. [[CrossRef](#)]
23. Mahler, R.P.S. *Statistical Multisource Multitarget Information Fusion*; Artech House, Inc.: Boston, MA, USA, 2007.
24. Kim, C.; Li, F.; Ciptadi, A.; Rehg, J.M. Multiple Hypothesis Tracking Revisited. In Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 7–13 December 2015; pp. 4696–4704.
25. Wang, L.Y.; Zhao, Y.J.; Wang, W.X. Study on Theory of Multi-target Tracking and Data Association Algorithms in Phased Array Radar. In Proceedings of the 2006 6th International Conference on ITS Telecommunications, Chengdu, China, 21–23 June 2006; pp. 1232–1235. [[CrossRef](#)]
26. Liggins, M., II; Hall, D.; Llinas, J., Eds. *Handbook of Multisensor Data Fusion: Theory and Practice*, 2nd ed.; CRC Press: Boca Raton, FL, USA, 2009.
27. Zaveri, M.A.; Merchant, S.N.; Desai, U.B. Data Association for Multiple Target Tracking: An Optimization Approach. In *Neural Information Processing*; Springer: Berlin/Heidelberg, Germany, 2004; pp. 187–192. [[CrossRef](#)]
28. Mahler, R. “Statistics 101” for multisensor, multitarget data fusion. *IEEE Aerosp. Electron. Syst. Mag.* **2004**, *19*, 53–64. [[CrossRef](#)]
29. Goodman, I.R.; Mahler, R.P.S.; Nguyen, H.T. *Mathematics of Data Fusion*; Springer: Dordrecht, The Netherlands, 1997. [[CrossRef](#)]
30. Mahler, R. Multitarget bayes filtering via first-order multitarget moments. *IEEE Trans. Aerosp. Electron. Syst.* **2003**, *39*, 1152–1178. [[CrossRef](#)]
31. Vo, B.N.; Ma, W.K. A closed-form solution for the probability hypothesis density filter. In Proceedings of the 2005 7th International Conference on Information Fusion, Philadelphia, PA, USA, 25–28 July 2005. [[CrossRef](#)]
32. Clark, D.; Bell, J. Convergence results for the particle PHD filter. *IEEE Trans. Signal Process.* **2006**, *54*, 2652–2661. [[CrossRef](#)]
33. Clark, D.; Vo, B.N. Convergence Analysis of the Gaussian Mixture PHD Filter. *IEEE Trans. Signal Process.* **2007**, *55*, 1204–1212. [[CrossRef](#)]
34. Clark, D.; Vo, B.T.; Vo, B.N. Gaussian Particle Implementations of Probability Hypothesis Density Filters. In Proceedings of the 2007 IEEE Aerospace Conference, Big Sky, MT, USA, 3–10 March 2007. [[CrossRef](#)]
35. Clark, D.; Bell, J. Multi-target state estimation and track continuity for the particle PHD filter. *IEEE Trans. Aerosp. Electron. Syst.* **2007**, *43*, 1441–1453. [[CrossRef](#)]
36. Lin, L.; Bar-Shalom, Y.; Kirubarajan, T. Track labeling and PHD filter for multitarget tracking. *IEEE Trans. Aerosp. Electron. Syst.* **2006**, *42*, 778–795. [[CrossRef](#)]

37. Zajic, T.; Mahler, R.P.S. Particle-systems implementation of the PHD multitarget-tracking filter. In Proceedings of the Signal Processing, Sensor Fusion, and Target Recognition XII, Orlando, FL, USA, 21 April 2003. [\[CrossRef\]](#)
38. VO, B.; MA, W. The Gaussian Mixture Probability Hypothesis Density Filter. *IEEE Trans. Signal Process.* **2006**, *54*, 4091–4104. [\[CrossRef\]](#)
39. Clark, D.; Panta, K.; Vo, B.N. The GM-PHD Filter Multiple Target Tracker. In Proceedings of the 9th International Conference on Information Fusion, Florence, Italy, 10–13 July 2006. [\[CrossRef\]](#)
40. Clark, D.; Godsill, S. Group Target Tracking with the Gaussian Mixture Probability Hypothesis Density Filter. In Proceedings of the 2007 3rd International Conference on Intelligent Sensors, Sensor Networks and Information, Melbourne, Australia, 3–6 December 2007. [\[CrossRef\]](#)
41. Hao, Y.; Meng, F.; Zhou, W.; Sun, F.; Hu, A. Gaussian mixture probability hypothesis density filter algorithm for multi-target tracking. In Proceedings of the 2009 IEEE International Conference on Communications Technology and Applications, Beijing, China, 16–18 October 2009. [\[CrossRef\]](#)
42. Baisa, N.L.; Wallace, A. Development of a N-type GM-PHD filter for multiple target, multiple type visual tracking. *J. Vis. Commun. Image Represent.* **2019**, *59*, 257–271. [\[CrossRef\]](#)
43. Mahler, R. PHD filters of higher order in target number. *IEEE Trans. Aerosp. Electron. Syst.* **2007**, *43*, 1523–1543. [\[CrossRef\]](#)
44. Nannuru, S.; Blouin, S.; Coates, M.; Rabbat, M. Multisensor CPHD filter. *IEEE Trans. Aerosp. Electron. Syst.* **2016**, *52*, 1834–1854. [\[CrossRef\]](#)
45. Vo, B.T.; Vo, B.N.; Cantoni, A. Analytic Implementations of the Cardinalized Probability Hypothesis Density Filter. *IEEE Trans. Signal Process.* **2007**, *55*, 3553–3567. [\[CrossRef\]](#)
46. Mahler, R. A brief survey of advances in random-set fusion. In Proceedings of the 2015 International Conference on Control, Automation and Information Sciences, Changshu, China, 29–31 October 2015. [\[CrossRef\]](#)
47. Mahler, R. Exact Closed-Form Multitarget Bayes Filters. *Sensors* **2019**, *19*, 2818. [\[CrossRef\]](#)
48. Laneuville, D.; Houssineau, J. Passive multi target tracking with GM-PHD filter. In Proceedings of the 2010 13th International Conference on Information Fusion, Edinburgh, UK, 26–29 July 2010. [\[CrossRef\]](#)
49. Battistelli, G.; Chisci, L.; Fantacci, C.; Farina, A.; Graziano, A. Consensus CPHD Filter for Distributed Multitarget Tracking. *IEEE J. Sel. Top. Signal Process.* **2013**, *7*, 508–520. [\[CrossRef\]](#)
50. Gulmezoglu, B.; Guldogan, M.B.; Gezici, S. Multiperson Tracking with a Network of Ultrawideband Radar Sensors Based on Gaussian Mixture PHD Filters. *IEEE Sens. J.* **2015**, *15*, 2227–2237. [\[CrossRef\]](#)
51. Dias, S.S.; Bruno, M.G.S. Cooperative Target Tracking Using Decentralized Particle Filtering and RSS Sensors. *IEEE Trans. Signal Process.* **2013**, *61*, 3632–3646. [\[CrossRef\]](#)
52. Bar-Shalom, Y.; Li, X.R.; Kirubarajan, T. *Estimation with Applications to Tracking and Navigation*; John Wiley Sons: Hoboken, NJ, USA, 2001.
53. Ristic, B.; Beard, M.; Fantacci, C. An Overview of Particle Methods for Random Finite Set Models. *Inf. Fusion* **2016**, *31*, 110–126. [\[CrossRef\]](#)
54. Ristic, B. *Particle Filters for Random Set Models*; Springer: New York, NY, USA, 2013. [\[CrossRef\]](#)
55. Mullane, J.; Vo, B.N.; Adams, M.; Vo, B.T. *Random Finite Sets for Robot Mapping and SLAM: New Concepts in Autonomous Robotic Map Representations*; Springer Tracts in Advanced Robotics; Springer: Berlin/Heidelberg, Germany, 2011; Volume 72.
56. Erdinc, O.; Willett, P.; Bar-Shalom, Y. The Bin-Occupancy Filter and Its Connection to the PHD Filters. *IEEE Trans. Signal Process.* **2009**, *57*, 4232–4246. [\[CrossRef\]](#)
57. Sidenbladh, H. Multi-target particle filtering for the probability hypothesis density. In Proceedings of the Sixth International Conference of Information Fusion, Cairns, Australia, 8–11 July 2003. [\[CrossRef\]](#)
58. Schuhmacher, D.; Vo, B.T.; Vo, B.N. A Consistent Metric for Performance Evaluation of Multi-Object Filters. *IEEE Trans. Signal Process.* **2008**, *56*, 3447–3457. [\[CrossRef\]](#)
59. Ristic, B.; Vo, B.N.; Clark, D.; Vo, B.T. A Metric for Performance Evaluation of Multi-Target Tracking Algorithms. *IEEE Trans. Signal Process.* **2011**, *59*, 3452–3457. [\[CrossRef\]](#)
60. Yuan, X.; Lian, F.; Han, C. Models and Algorithms for Tracking Target with Coordinated Turn Motion. *Math. Probl. Eng.* **2014**, *2014*, 1–10. [\[CrossRef\]](#)
61. Li, X.R.; Jilkov, V. Survey of maneuvering target tracking. Part I: Dynamic models. *IEEE Trans. Aerosp. Electron. Syst.* **2003**, *39*, 1333–1364. [\[CrossRef\]](#)
62. Cano, P.; Ruiz-del Solar, J. Robust Tracking of Soccer Robots Using Random Finite Sets. *IEEE Intell. Syst.* **2017**, *32*, 22–29. [\[CrossRef\]](#)