Available online at www.sciencedirect.com

ScienceDirect

IFAC PapersOnLine 55-37 (2022) 333–338

A Trifacacking System for Dynamic Subset Targets using Probability Hypothesis Filtering ⋆

R.A. Thivanka Perera * Andrew Phillips ** Chengzhi Yuan *** Paolo Stegagno ****

* ECBE, ICRobots lab, University of Rhode Island, Kingston, RI, USA (e-mail: thiva@uri.edu).
** ECBE, ICRobots lab, University of Rhode Island, Kingston, RI, USA (e-mail: andrew_phillips@uri.edu).
*** MCISE, ICRobots lab, University of Rhode Island, Kingston, RI, USA (e-mail: cyuan@uri.edu).
**** ECBE, ICRobots lab, University of Rhode Island, Kingston, RI, USA (e-mail: pstegagno@uri.edu).

Abstract: In this paper we present a novel formulation of the multi-target tracking problem in which the target represents a variable shape and size subset of the state space. Due to the nonlinear nature of the system and the inherent multi-target nature of the problem, we provide a solution based on the particle implementation of the PHD filter. Testing in simulations yields promising results allowing a significant reduction of the errors with respect to measurements.

Keywords: Sensor fusion, mobile robotics, multi-target tracking

1. INTRODUCTION

Autonomous robotic systems are a novel approach for early-action response of natural or human-induced disasters. These incidents typically occur in remote, harsh environments which pose as a safety risk for human operators and involve great financial burden. Recent advances in autonomous robotic systems have granted opportunities for first responders to step away from the action and gather real time information, all while improving the mission's cost efficiency.

One notorious example of a high-risk, large-scale disaster is an oceanic crude oil spill. Autonomous robotic technology has been developed to provide a reliable and cost effective option for tracking spatiotemporal environmental phenomena (Pashna et al., 2020). Another environmental hurdle under investigation is the emergence of harmful cyanobacteria blooms. Cyanobacteria, also referred to as blue green algae, are microorganisms that have existed in water bodies such as lakes, rivers, and reservoirs since the beginning of life on Earth. The term "cyanobacteria blooms" refers to when cyanobacteria populations begin to multiply at a very quick pace. Cyanobacteria blooms occur as circumstance of an excess of nutrients entering a slow moving water environment and pose as a threat to the health of all organisms within its community (Carmichael, 1994; Falconer, 1999; RI-DEM; van Halderen et al., 1995). This paper was developed within the context of a larger project to improve the knowledge and techniques of tracking and predicting cyanobacteria blooms. Despite this research's specific application, the developed methodology discussed in this paper is applicable to a large variety of target tracking problems within and beyond the field of natural science.

Current efforts in UAV-assisted cyanobacteria bloom monitoring mainly focus on mere bloom identification. Since cyanobacteria blooming events may ensue for several days, weeks, or months, it is necessary to develop tracking techniques that identify and follow bloom populations over time. However, the underlying problem of tracking cyanobacteria blooms, or more in general pollutants, within a water body has very peculiar characteristics that call for ad-hoc methodologies. Although the problem is inherently related to multitarget tracking, with respect to canonical multitarget tracking problems there are definite differences. In canonical problems, a finite number of targets evolves in the state-space. In our case instead, entire subsets of the state space constitutes the targets. Furthermore, the evolution of these subsets includes not only translations or rotations, but also changes in shape and size. More practically, the large monitoring area require computationally efficient methodologies.

In this paper, we present a formalization of the novel subset-tracking problem to take into account these specific aspects. Furthermore, we propose a solution based on a particle implementation of the Probability Hypothesis Density (PHD) filter. In recent years, the PHD filter, particularly in its Gaussian Mixture (GM-PHD) implementation, has been employed in robotics mostly in localization or canonical multi-target tracking problems. In our case, the employment of the PHD filter is suggested by the inherent multi-target tracking nature of the problem, and its recursive filter formulation allows for the inclusion of

⋆ This work was supported by the National Science Foundation under Grant RII Track-2 FEC 1923004.

10.1016/j.ifacol.2022.11.206

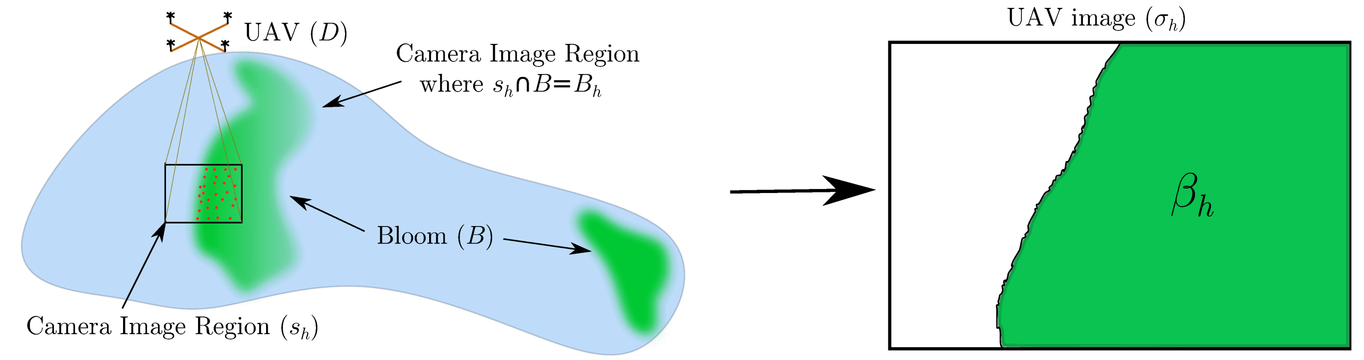


Fig. 1. (left) Depiction of lake region, cyanobacteria population, UAV and region that the camera image captures. (Right) UAV camera image of region Σ_n

system dynamics. An additional desirable characteristic that makes the PHD filter a good solution for this problem is its ability to cope with variable number of targets through the birth, death, and spawning probabilities that can be used to represent changes in shape and size of the tracked targets. These probabilities can also be linked to environmental factors that are known to increase or decrease the likelihood of these phenomena to occur, hence allowing for better tracking and prediction results.

The contribution of this paper is therefore twofold:

- we formalize a new instance of the multitarget tracking problem in which the targets are represented as variable shape and size subsets of the state-space;
- we provide a solution based on the particle implementation of the PHD filter detailing the role of each term.

The rest of this paper is organized as follows. In Section 2 we formulate the problem. In Section 3 we provide the necessary background on the PHD filter. In Section 4 we present our solution, that is tested in simulations presented in Section 5. Section 6 concludes the paper.

2. PROBLEM SETTINGS

The phenomena we consider in Figure 1 consists of a target set B moving on the surface of the water body L. Both L and B are defined as sets in $\mathbb{R}^2$ and are expressed in a world reference frame F^W. This simplification will help with the descriptions of equations later on in this paper. Furthermore, B is a subset of L such that $B \subseteq L$. The size and shape of the target set B is unknown and time variant. The evolution of B over time is driven by environmental factors over n time steps (Reynolds and Walsby, 1975). The generic time step will be indicated through the letter k, such that $k = 0, 1, \ldots, n-1$. In the case of cyanobacteria blooms, these time steps can be of several days or weeks, depending on the body of water that is under investigation. After each time step, noisy measurements of the average motion $W(k)$ of B are gathered as a pair $\bar{W}(k) = \{\bar{W}_s, \bar{W}_d\}$, expressing the average speed $\bar{W}_s$ and the average direction $\bar{W}_d$. Measurements of environmental factors $T(k)$ leading to the growth or decline of B are summarized with a single measurement $\bar{T}(k)$.

A multi-rotor UAV D is used to collect measurements of L at each time step. The UAV D gathers data along a predetermined path within the world frame F^W. D is equipped with an attached reference frame denoted as F^D. The position of D in F^W is denoted by the vector $D^W = [D^W_x \, D^W_y \, D^W_z]^T \in \mathbb{R}^3$. The orientation of D in F^W is denoted by the rotation matrix $R^W_D \in SO(3)$. In the following, when relevant we will indicate the reference frame of a quantity $*$ by using the right superscript $*^W$ or $*^D$. When the reference frame superscript is omitted, the reference frame is assumed to be F^W. Noisy measurements of the position and orientation of the UAV are available at all times through GPS and IMU modules, and are referred to as $\bar{D}^W$ and $\bar{R}^W_D$ respectively.

D is equipped with a camera that is oriented along the negative Z axis of frame F^D (i.e., towards the ground plane) and captures images of L. For simplicity, we assume that the camera's position in F^W corresponds to the position of UAV D^W. At each time step, the camera captures a set of images $\Sigma(k) = \{\sigma_1, \sigma_2, \ldots, \sigma_a\}$ within L. An arbitrary image σ_h is a set of pixels, where a specific pixel can be identified as $\rho_h = \{\varrho_{h_x}, \varrho_{h_y}\}$ for $x = \{0, 1, 2, \ldots, 639\}$ and $y = \{0, 1, 2, \ldots, 359\}$.

Let s_h be the subset of L represented in image σ_h, then the area surveyed by D during a flight is $S(k) = s_1 \cup s_2 \cup \ldots \cup s_a \subseteq L$. We further denote with $B_h = s_h \cap B$ the intersection of s_h and B, i.e., B_h is the subset of B that is represented in the image σ_h. Furthermore, the set of pixels within σ_h that correspond to B_h is referred to as β_h, with $\beta_h \subseteq \sigma_h$. We assume that through color extraction it is possible to identify β_h within σ_h, with a correct identification rate $p_b \in [0, 1]$. with respect to our application, this is an idealization of the sensor measurements. In fact, simple color extraction is unlikely to yield good results in identifying cyanobacteria in the water. However, several works have been done in the identification of algal blooms through satellite and UAV multi-spectral imagery (Richardson, 1996; Bostater Jr et al., 2010; Wu et al., 2019; Kislik et al., 2018).

Following from the definitions above, we define the following Subset tracking problem.

Problem 1: The Subset Tracking problem consists in computing an estimate $\hat{B}(k)$ of $B \subseteq L$ at all time steps $k = 0, 1, \ldots, n-1$ given the measurements $\Sigma(k)$, $\bar{W}(k)$, $\bar{T}(k)$.

3. BACKGROUND

This Section provides the necessary background on the PHD filter and is mostly based on Vo and Ma (2006); Junjie et al. (2015); Zajic and Mahler (2003); Vo et al. (2003). There are n targets living in a space $\mathcal{X}$, with n unknown and varies over time. The goal of the PHD filter is to compute an estimate of the PHD of the targets in $\mathcal{X}$. The PHD $f_k(x)$ at time k is defined as the function such that its integral over any subset $S \subseteq \mathcal{X}$ is the expected number of targets $N(S)$ in that subset, i.e., $N(S) = \int_S f_k(x)dx$.

The structure of the PHD filter is that of a recursive estimator composed of two main steps: the time update and the measurement update. The time update produces a prediction of the PHD $f_{k|k-1}(x)$ at time step k given the estimate $f_{k-1|k-1}(x)$ at time step $k-1$, through the time update equation:

$$f_{k|k-1} = b_{k|k-1}(x) + \int [P_S(x')f_{k|k-1}(x|x') + b_{k|k-1}(x|x')]f_{k-1|k-1}(x')dx' \quad (1)$$

where $b_{k|k-1}(x)$ is the probability that a new target appears in x between times $k-1$ and k, $P_S(x')$ is the probability that a target in x' at time $k-1$ will survive into step k, $f_{k|k-1}(x|x')$ is the probability density that a target in x' at time $k-1$ moves to x at time k, and $b_{k|k-1}(x|x')$ is the probability that a new target spawns in x at time k from a target in x' at time $k-1$. Note that both $f_{k-1|k-1}(x)$ and $f_{k|k-1}(x)$ are computed considering only the measurements up to time $k-1$.

Measurements Z_k are incorporated in the estimate with the measurements update to compute the posterior PHD:

$$f_{k|k}(x) = f_{k|k-1}(x) * \Bigg[1 - P_D(x) + \sum_{z\in Z_k} \frac{P_D(x)g(z|x)}{\lambda c(z) + \int P_D(x')g(z|x')f_{k|k-1}(x')dx'}\Bigg] \quad (2)$$

where $P_D(x)$ is the probability that a measurement is collected from a target with state x, $g(z|x)$ is the sensor likelihood function, and $\lambda c(z)$ expresses the probability that a given measurement z is a false positive.

The implementation of equations (1)-(2) can be done using a sequential Monte Carlo (SMC) method. The SMC-PHD method approximates all PHD functions $f_{k-1|k-1}(x)$, $f_{k|k-1}(x)$, and $f_{k|k}(x)$ through sums of $L_{*|*}$ weighted samples (or particles) in the form:

$$f_{*|*}(x) = \sum_{i=1}^{L_{*|*}} w^i_{*|*}\delta(x - x^i_{*|*}) \quad (3)$$

where $w^i_{*|*}$ is the weight of the i-th sample, and the Dirac delta function $\delta(x - x^i_{*|*})$ centered in $x^i_{*|*}$ is used to express the position of the i-th particle.

Introducing the SMC representation (3) in equation (1), the $L_{k-1|k-1}$ particles composing the posterior $f_{k-1|k-1}(x)$ can be propagated into time step k using the sampling property of the Dirac delta:

$$f_{k|k-1}(x) = \sum_{i=1}^{L_{k-1|k-1}} P^i_S w^i_{k-1|k-1}\delta(x - x^i_{k|k-1}) + \sum_{i=L_{k-1|k-1}+1}^{L_{k-1|k-1}+J_k} w_\gamma \delta(x - x^i_{k|k-1}) \quad (4)$$

where the birth target probability $b_{k|k-1}(x)$ is represented using the sum of J_k samples which are drawn from importance function, and assigned weight constant w_γ. In this step, the spawning probability $b_{k|k-1}(x|x')$ is assumed to be zero, and the new i-th particle state $x^i_{k|k-1}, i = 1, \dots, L_{k-1|k-1}$ is a random sample extracted from the distribution $f_{k|k-1}(x|^i_{k-1|k-1})$. Note that the number of particles composing $f_{k|k-1}(x)$ is therefore $L_{k|k-1} = L_{k|k-1} + J_k$.

Introducing the SMC-PHD representation (3) in equation (2), assuming $\lambda c(z) = 0$, the SMC-PHD filter measurement update equation becomes:

$$f_{k|k}(x) = \sum_{i=1}^{L_{k|k-1}+J_k} w^i_{k|k}\delta(x - x^i_{k|k}) \quad (5)$$

where

$$w^i_{k|k}(x) = (1 - P^i_D)w^i_{k|k-1} + \sum_{z\in Z_k} \frac{P^i_D w^i_{k|k-1} g(z|x^i_{k|k-1})}{\sum_{i=1}^{L_{k|k-1}+J_k} P^i_D w^i_{k|k-1} g(z|x^i_{k|k-1})} \quad (6)$$

where $P^i_D = P_D(X^i_{k|k-1})$. The equation (5) shows that only the weights are updated while the particle state remains unchanged.

The resampling of PHD posterior $f_{k|k}(x)$ to time step k is performed in two steps. In the first step, the particles with weight lower than pruning weight w^p is removed, effectively reducing the number of samples in the posterior $f_{k|k}$. Secondly, Each particle that has weight over one $\{w^i_{k|k} > 1\}$ is copied $N = w^i_{k|k-1}$ times with a new weight constant of one. The total number of particles are changed from $L_{k|k-1} + J_k$ to $L_{k|k}$ after the resampling.

4. METHODOLOGY

In this study we use the particle implementation of the PHD filter considering the nonlinear behavior of the targets, their variable shapes and sizes, and the lack of a proven model to predict the behavior of B. Furthermore, the particle PHD filter provides a computational edge over other PHD implementations, particularly during the measurement update where the only quantity that is updated is the particle weight. In comparison, during the GM-PHD filter measurement update, not only the mean value of each Gaussian component must be updated, but also the covariance matrix must be propagated in time.

The other side of the coin is the need for an additional step, the resampling, that however can be computed offline. Therefore, a generic target particle is only described through its position $x_{*|*} \in \mathbb{R}$ in F^w and the weight w^i_k. In summary, our methodology consists with three main steps that are iteratively executed at each time step: the time update, using the environmental readings between flights, the measurement update, using the images, and the resampling step.

4.1 Time Update

In the time update, the state of all components of the PHD is updated using a noisy measurement $\{\bar{W}_s,\ \bar{W}_d\}$ of the average motion $\{W_s,\ W_d\}$ and Gaussian random noise with zero mean and known covariance. Due to the particular definition of the average motion as a magnitude and an angle, the resulting update equations are as follows:

$$x^i_{k|k-1} = x^i_{k-1|k-1} + \bar{W}_s * \sin(\bar{W}_d) + N^i_x \tag{7}$$

$$y^i_{k|k-1} = y^i_{k-1|k-1} + \bar{W}_s * \cos(\bar{W}_d) + N^i_y \tag{8}$$

where N^i_x and N^i_y are noise samples extracted from a normal distribution $N^i = [N^i_x \ \ N^i_y]^T = \mathcal{N}(.;0;cov)$, $N^i \in \mathbb{R}^2$. Whenever a particle would escape L by applying (7)-(8), a reflection model is applied so that a the particle is reflected from L's boundary.

Other environmental factors, here summarized in the temperature, are used to estimate the variation of the weight of the particles. For example, the average measured temperature $\bar{T}$ between two UAV flights will be used to adjust the weights of PHD components as it would affect blooms to increase or decrease in size. We used 25 deg Celsius as the mid margin to adjust the weight.

$$w^i_{k|k-1} = w^i_{k-1|k-1} + (T - 25) * 0.01 \tag{9}$$

This is equivalent to selecting the survival probability in the PHD filter through the measurements of environmental factors.

The last step in the time update is the introduction of new components for birth targets. If no model is known (i.e., blooms can emerge anywhere in the water body), the new components can be uniformly distributed across L:

$$x^i_{k|k-1} = M^i \tag{10}$$

$$w^i_{k|k-1} = P_{new} \tag{11}$$

where $M_i \in L$ is a random sample from a uniform distribution on L, and P_{new} is the uniform probability of having a new target in any $x^i_{k|k-1} \in L$. In case a probability $P_{new}(x)$ of new targets is known (e.g., by the historic data), this can be utilized as $w^i_{k|k-1} = P_{new}(x)$.

4.2 Measurement Extraction

The measurement extraction is executed between the time update step and the measurement update step to extract measurements $Z = \bar{G}_h$ from the set of images $\Sigma(k)$ captured by D during a time step k. For each image σ_h, a color extraction algorithm (using OpenCV) identifies the set of pixels β_h within the image σ_h. Among β_h, a set of discrete points $\Gamma_h = \{\gamma_{h_1}, \gamma_{h_2}, \ldots, \gamma_{h_e}\}$ is selected, where $\Gamma_h \subseteq \beta_h$. The set Γ_h is the intersection of β_h and a lattice Ξ_h, where $\Gamma_h = \beta_h \cap \Xi_h$. The pixel locations of Ξ_h within σ_h are located so that each lattice point of Ξ_h corresponds to one meter of distance between points in s_h. This technique limits the number of measurements that are fed to the PHD filter.

Each pixel value γ_{h_j} is translated into a three-dimensional position vector in reference frame F^D. This position vector is referred to as $g^D_{h_j}$:

$$\bar{g}^D_{h_j} = \begin{bmatrix} \bar{g}^D_{h_j x} \\ \bar{g}^D_{h_j y} \\ \bar{g}^D_{h_j z} \end{bmatrix} = \begin{bmatrix} \dfrac{(\gamma^D_{h_j x} - \psi_v)\bar{D}^W_{h_x}}{\alpha_x f} \\ \dfrac{(\gamma^D_{h_j y} - \phi_v)\bar{D}^W_{h_y}}{\alpha_y f} \\ \bar{D}^W_{h_z} \end{bmatrix} \tag{12}$$

where ψ_v, ϕ_v, α_x, α_y, and f are constants regarding the camera parameters.

Next it is necessary to transform $\bar{g}^D_{h_j}$ into $\bar{g}^W_{h_j}$. This is done using the homogeneous transformation matrix T^W_D:

$$\bar{g}_{h_j} = \bar{g}^W_{h_j} = \bar{T}^W_D \bar{g}^D_{h_j} = \begin{bmatrix} \bar{R}^W_D & \bar{D}^W \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \bar{g}^D_{h_j} \\ 1 \end{bmatrix} \tag{13}$$

The set $\bar{G}_h = \bar{G}^W_h$ is the determined set of positions measurements of particles representing B in F^W, and is used in the measurement update of the PHD filter.

4.3 Measurement Update

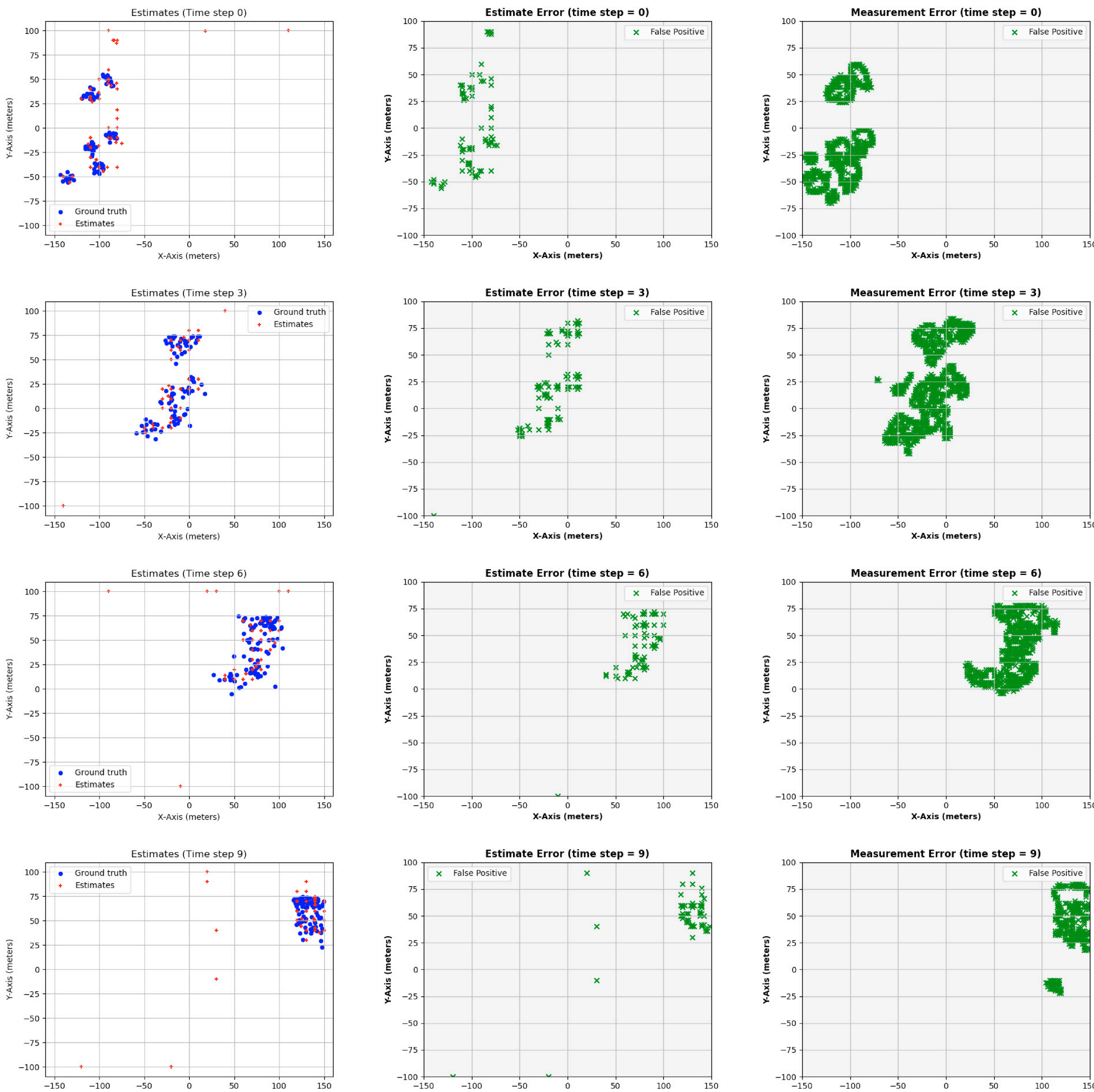
The measurement update is performed when the filter receives a set of measurements $Z = \bar{G}_h$ from measurement extraction algorithm. Note that each measurement update is divided into a sequence of a consecutive updates, with a equal to the number of images collected in a single flight. Theoretically, each $\bar{G}_h$ will be used to update all the particles in the posterior according to (5)-(6). However, the $P_D(x^i_{k|k-1})$ for each particles $x^i_{k|k-1}$ is computed as:

$$P_D(x^i_{k|k-1}) = \begin{cases} p_d & \text{if } x^i_{k|k-1} \in s_h \\ 0 & \text{otherwise} \end{cases} \tag{14}$$

where $0 < p_d \leq 1$ is a constant that reflects the probability that a measurement is collected from a particle that is within the field of view of the image σ_h. Therefore, P_D serves also the purpose of not updating the particles that are outside of s_h, and in order to drastically increase the computational efficiency of the algorithm, before a measurement update is performed only the particles such that $x^i_{k|k-1} \in s_h$ are selected for the update. Furthermore, the computed P_D values are substituted into equations (5)-(6) to perform the measurement update. In the same equations, $g(z|\delta(x - x^i_{k|k-1}))$ is assumed to be a Gaussian distribution and is computed as:

$$g(\bar{g}_{h_j}|x^i_{k|k-1}) = \mathcal{N}\{\bar{g}_{h_j}; x^i_{k|k-1}, C_{2x2}\} \tag{15}$$

where C_{2x2} is a suitable covariance matrix to represent target position measurement noise, and $z = \bar{g}_{h_j} \in \bar{G}_h$. With this definition we assume that the target position measurements are affected by zero-mean Gaussian noise. This is a realistic assumption as the UAV positioning measurements utilized in (13), as well as the application of the camera pinhole model in (12) introduce noise into the measurement extraction process.



(a) (b) (c)

Fig. 2. Results of a simulation at four significant time steps. (a) Ground truth vs Estimates for step 0 , 3, 6, and 9. (b) Estimation error for step 0, 3, 6, and 9. (c) Measurement error for time steps 0, 3 , 6, and 9.

4.4 Resampling

The resampling step is performed after the measurement update, i.e., after sequentially using all images to perform partial measurement update. The resampling procedure follows the standard resampling procedure explained in the Section 3, where a pruning weight of $w^p = 0.001$ was used to remove particles with lower weights.

5. SIMULATIONS

We have tested the tracking system through numerical simulations in ROS/Gazebo. In particular, through standard Gazebo functionalities we have simulated a UAV surveying a lake environment. The UAV is equipped with a camera and follows a predetermined trajectory of about 15 minutes to survey the entire lake. The survey is performed on ten time steps, with each time step of the duration of a week.

ROS/Gazebo does not have the functionality to simulate a cyanobacteria bloom, or in general a subset evolving on the water surface. Therefore, we have created a set of (initially) six independent sub-targets (all included into the set B) evolving on the water surface for ten time steps. Each target is initially represented by a set of particles, each particle representing a $1m$ circle. Each sub-target evolves with a different forcing term, and zero-mean fixed covariance Gaussian noise is added to the particles at the end of each time step evolution. A reflection model is applied at the top boundary of the simulated lake, that is set for $y = 75m$. The combination of additive noise, different forcing terms, and the reflection model, makes the target shape and area change and evolve over time, as visible in Figure 2(a), were the simulated particles are represented in blue. The target set is reprojected online onto the images collected by the simulated UAV as yellow circles of $1m$ radius to simulate appropriate image measurements, therefore the covariance matrix $C_{2\times 2}$ in equation (15) is selected as the identity matrix.

The developed system takes as input the images with the reprojected target B as input to perform a color extraction algorithm and the filtering as detailed in Section 4. Note that, in order to simulate a missed detections, 20% of the measurements extracted following the methodology presented in Section 4.2 are removed before each measurement update. Accordingly, the parameter p_d in equation (14) is selected as $p_d = 0.8$.

Figure 2(a) reports the ground truth particles in blue and the estimated particles in red, showing a good accordance among the two sets. Figure 2(b) and 2(c) show the locations were the filter and the raw measurements respectively commit an estimation error. Together, the two figures provide a comparison between the estimates (Figure 2(b)) and the raw measurements (Figure 2(c)), showing that the estimation system greatly reduce the errors in the raw measurements. This is also confirmed by a numerical analysis of these data, that confirms that the PHD filter's estimates yielded over 600 percent fewer errors than the raw measurements from the RGB camera and color-extraction.

6. CONCLUSIONS

In this paper we have presented a novel instance of the multi-target tracking problem in which the tracked target is a subset of the state space. A possible solution to this problem has been identified in the application of the particle implementation of the PHD filter. The proposed solution has been tested in simulation and has shown promising results in reducing the error with respect to the direct use of the measurements.

On the other hand, the performance of PHD filters depend on the knowledge of the system dynamics and several parameters, including the system and measurement noise intensities, and the new target and survival probabilities. In the future we plan to improve the system model by applying hydrodynamic libraries to create the ground truth particles and to update the estimates in the time update. More realistic survival and new target models from historical data are also under investigation to improve the weight estimation. Moreover, we will be collecting real data with a UAV equipped with a multispectral camera to test our system on actual lake images. Other natural extensions that we will be working on include a multi-robot multi-sensor setup, and a 3D implementation of the filtering including also underwater measurements.

REFERENCES

Bostater Jr, C.R., Jones, J., Frystacky, H., Kovacs, M., and Jozsa, O. (2010). Image analysis for water surface and subsurface feature detection in shallow waters. In *Proc. SPIE 7825, Remote Sensing of the Ocean, Sea Ice, and Large Water Regions 2010*, volume 7825, 152 – 158. International Society for Optics and Photonics. doi:10.1117/12.870728.

Carmichael, W.W. (1994). The toxins of cyanobacteria. *Scientific American*, 270(1), 78–86. doi:10.1038/scientificamerican0194-78.

Falconer, I.R. (1999). An overview of problems caused by toxic blue–green algae (cyanobacteria) in drinking and recreational water. *Environmental Toxicology*, 14(1), 5–12. doi:10.1002/(SICI)1522-7278(199902)14:1⟨5::AID-TOX3⟩3.0.CO;2-0.

Junjie, W., Lingling, Z., Xiaohong, S., and Peijun, M. (2015). Distributed computation particle phd filter. doi:10.48550/arXiv.1503.03769.

Kislik, C., Dronova, I., and Kelly, M. (2018). Uavs in support of algal bloom research: A review of current applications and future opportunities. *Drones*, 2(4), 35. doi:10.3390/drones2040035.

Pashna, M., Yusof, R., Ismail, Z.H., Namerikawa, T., and Yazdani, S. (2020). Autonomous multi-robot tracking system for oil spills on sea surface based on hybrid fuzzy distribution and potential field approach. *Ocean Engineering*, 207, 107238. doi:10.1016/j.oceaneng.2020.107238.

Reynolds, C.S. and Walsby, A.E. (1975). Water-blooms. *Biological Reviews*, 50(4), 437–481. doi:10.1111/j.1469-185X.1975.tb01060.x.

RI-DEM (2016). Cyanobacteria (Blue-green Algae). URL `http://www.dem.ri.gov/programs/benviron/water/quality/surfwq/pdfs/bgalgae.pdf`. Accessed on 07.14.2022.

Richardson, L.L. (1996). Remote sensing of algal bloom dynamics. *BioScience*, 46(7), 492–501. doi:10.2307/1312927.

van Halderen, A., Harding, W., Wessels, J.C., Schneider, D., Heine, E., van der Merwe, J., and Fourie, J. (1995). Cyanobacterial (blue-green algae) poisoning of livestock in the western cape province of south africa. *Journal of the South African Veterinary Association*, 66(4), 260–264. doi:10.10520/AJA00382809_1591.

Vo, B.. and Ma, W.. (2006). The gaussian mixture probability hypothesis density filter. *IEEE Transactions on Signal Processing*, 54(11), 4091–4104. doi:10.1109/TSP.2006.881190.

Vo, B.N., Singh, S., and Doucet, A. (2003). Sequential monte carlo implementation of the phd filter for multi-target tracking. volume 2, 792– 799. doi:10.1109/ICIF.2003.177320.

Wu, D., Li, R., Zhang, F., and Liu, J. (2019). A review on drone-based harmful algae blooms monitoring. *Environmental monitoring and assessment*, 191(4), 1–11. doi:10.1007/s10661-019-7365-8.

Zajic, T. and Mahler, R.P.S. (2003). Particle-systems implementation of the PHD multitarget tracking filter. In I. Kadar (ed.), *Signal Processing, Sensor Fusion, and Target Recognition XII*, volume 5096, 291 – 299. SPIE. doi:10.1117/12.488533.